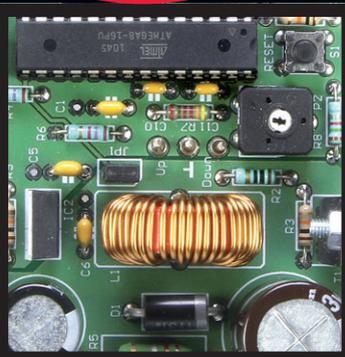
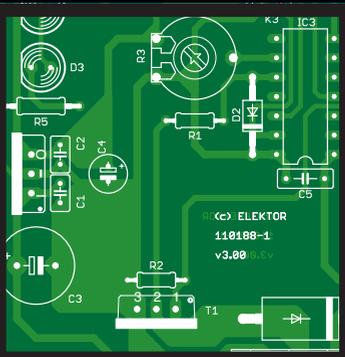


e lektor

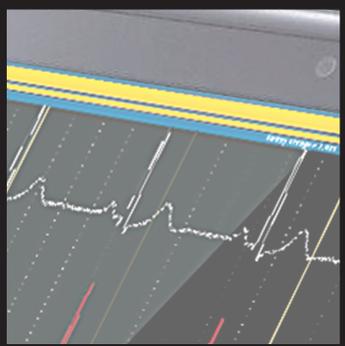
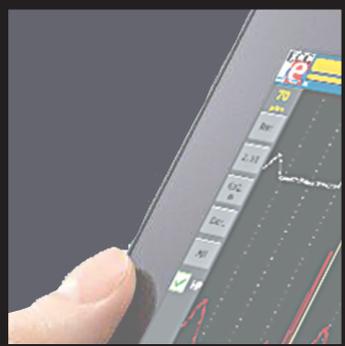
NUMÉRO DOUBLE | ÉTÉ 2013



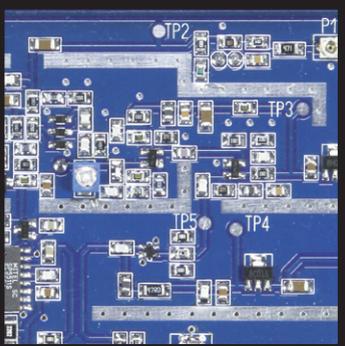
**AMBIANCE
LUMINEUSE
SUR COMMANDE
PAR LED RVB**



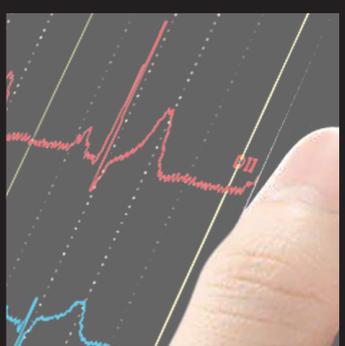
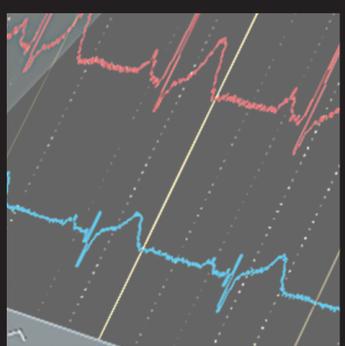
**ÉMETTEUR
FM 70 CM
À LARGE
SPECTRE**



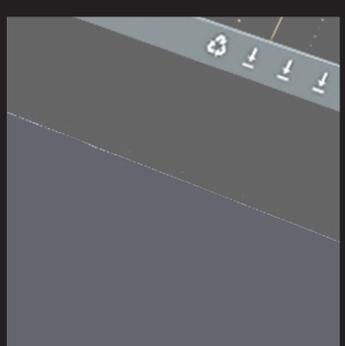
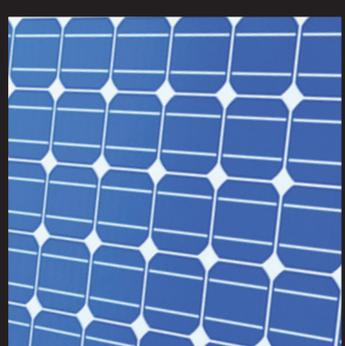
**ELEKTOR
CARDI♥-
SCOPE
ANDROID
BLUETOOTH**



**LIMITEUR
DE
COURANT
D'APPEL
X-TREME**



**132
PAGES DE
NOUVEAUX
PROJETS**



**TÉLÉCOMMANDE
PAR
SMARTPHONE**

(B) 16,50 € • (CH) 29,00 FS
(CAN) 20,99 \$ca • (And) 15,50 €
DOM surface 16,50 € • DOM avion 16,90 €
N Cal/S 2000 cfp

L 19624 -422- F: 15,50 €-RD

Le premier contrôleur d'alimentation analogique au monde avec optimisation numérique

Nouveau contrôleur d'alimentation analogique avec microcontrôleur intégré



Le nouveau contrôleur de gestion d'alimentation MCP1911 de Microchip vous permet d'associer la souplesse du traitement numérique et la connectivité I²C avec la vitesse, la performance et la résolution d'un contrôleur analogique dans vos applications de conversion d'énergie CC/CC.

Le MCP1911 est un nouveau contrôleur hybride à signaux mixtes qui associe la gestion d'alimentation analogique et numérique dans un seul circuit. En intégrant un contrôleur PWM analogique, un microcontrôleur PIC[®] Flash 8 bits, et des drivers MOSFET pour des applications de réduction de tension synchrones, le MCP1911 permet une conversion configurable et de rendement élevé.

Grâce à l'excellent comportement en transitoire de sa conversion analogique, le MCP1911 élimine le besoin d'un microcontrôleur rapide ou d'un convertisseur A/N haute vitesse, ce qui minimise la consommation et le coût de la solution.

Pour atteindre un rendement encore plus élevé, le MCP1911 peut être associé aux derniers MOSFET MCP87xxx haute vitesse de Microchip optimisés pour les alimentations. Ces MOSFET de 25V à niveau logique ont un faible Facteur de Mérite (FOM) et une résistance à l'état actif de 1,8 m Ω à 13 m Ω pour obtenir une conversion de puissance CC/CC de haut rendement.

DÉMARRAGE EN 3 ÉTAPES FACILES :

1. Évaluez le MCP1911 en utilisant la carte économique ADM00397
2. Associez le avec des MOSFET MCP87xxx optimisés pour les alimentations
3. Personnalisez la conversion CC/CC afin qu'elle corresponde à votre application

Pour plus d'informations, allez sur : www.microchip.com/get/eumcp1911



1



2



3



4



5

1 Andropod

Commandez vos montages avec un smartphone ou une tablette Android

Écran tactile à haute définition, méga puissance de calcul, connexion réseau et fonctions de téléphonie sans fil : les téléphones tactiles et les tablettes Android seraient des centrales de commande presque idéales pour nos projets d'électronique s'il était plus facile de s'y connecter. Voici Andropod, votre interface série TTL et RS485. Décollage vertical garanti ! L'offre de matériel Android (par différents fabricants) et de logiciel est énorme et le code source du système d'exploitation est libre. Le puissant kit de fonctions logicielles ou framework d'Android offre accès à presque toutes les fonctions matérielles et permet de programmer des applications élégantes et conviviales.

Réf 110405-91

2 Module réseau universel

Connectivité internet pour les nuls

Le raccordement de sa propre électronique par une liaison internet est une fonction impressionnante. Ce module réseau universel - composé d'un circuit compact, d'une bibliothèque de logiciels libres et d'un microcontrôleur utilisable immédiatement comme serveur web - s'adresse même aux débutants. Quant

aux électroniciens plus chevronnés, ils apprécieront à leur juste valeur les caractéristiques comme la communication SPI, PoE, et plus encore !

Réf 100552-91

3 LCR-Mètre 0,05 %

Le luxe de la précision accessible à tous

La remarquable précision de cet appareil et son étonnant confort d'utilisation sont le résultat d'une étude soignée. Il marche si bien derrière sa façade dépouillée, qu'on en oublierait presque la subtilité des techniques de mesure mises en œuvre. L'occasion rêvée, pour nos lecteurs passionnés par la mesure, de se faire plaisir. Si, comme nous, les prodiges que les techniques modernes mettent à notre portée vous émerveillent, venez palper le pouième de volt.

Module carte principale réf 110758-91

Module de l'extension afficheur et clavier réf 110758-92

Les deux modules réf 110758-93

4 Barostick

Clés USB, clés du succès ? Elles sont partout, elles sont notre album d'images, de vidéos, de musiques préférées, d'articles, de fichiers et même de tempé-

ratures. Et la pression atmosphérique, y aviez-vous pensé ? C'est fait : sur un baromètre sans mercure, avec un capteur Bosch, hectopascals et degrés Celsius rejoint Windows pour se faire tirer le portrait.

Réf 120481-91

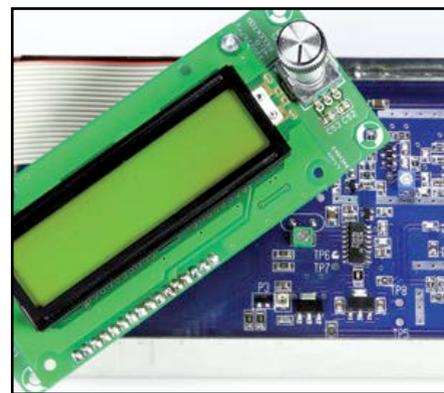
5 Accéléromètre DIPlômé

Accessoire d'adaptation pour puce d'accéléromètre (dont les surfaces de contact se trouvent sous le boîtier de type LGA) à une carte de prototypage ordinaire.

Réf 090535-91

Informations et gamme complète sur :

www.elektorpcbservice.com



● communauté

- 10 **elektor.labs célèbre Ste-Aise!**
Quand on aime, on ne compte pas. Enfin si... il a bien fallu compter pour constater qu'Elektor.LABS réunit plus de 5.000 membres enregistrés. Merci à tous !

● DesignSpark

- 50 **2^e jour – insérer un cartouche**
Comment insérer un cartouche dans un schéma à l'aide des bibliothèques.
- 12 **le jeu des différences : ARDUINO UNO contre GR SAKURA FULL**
Les Roses se mesurent aux Bleus.

● labs

- 11 **dessoudage de circuits intégrés CMS**
Ce n'est pas parce qu'un problème paraît délicat que sa solution ne pourra pas être simple. Et bon marché !

● projets

- 12 **télécommande par smartphone Android**
Se substitue aux télécommandes infrarouge classiques.
- 20 **interface universelle de mesure précise**
Précise, tout simplement.
- 22 **alimentation temporisée pour circuits de précision**
Précieux accessoire.

24 testeur de servo

Modélistes, ce testeur simple et indispensable fera votre bonheur !

26 ambiance lumineuse sur commande par LED RVB

Convient à toutes les LED RVB individuelles ou en rubans.

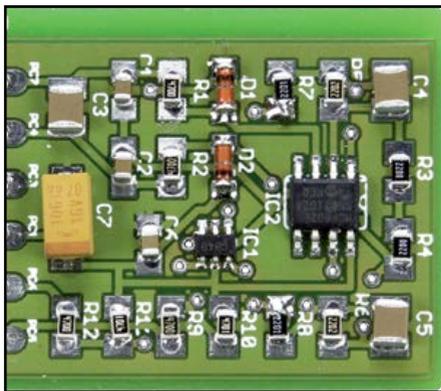
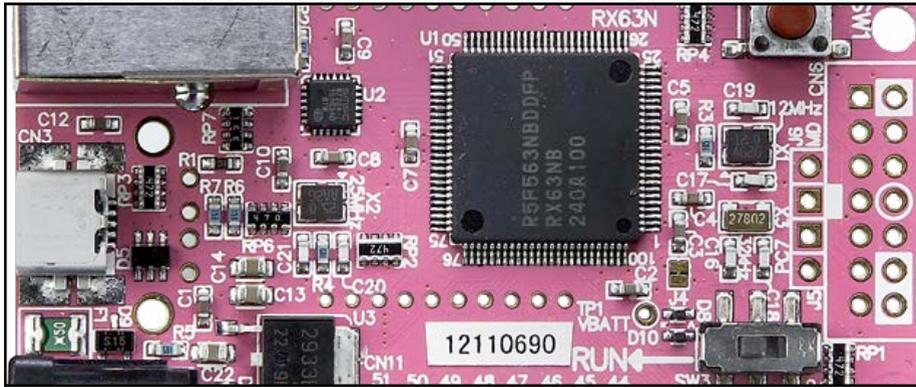
32 régulateur de pompe pour voilier ou caravane etc.

Supprime les inconvénients des circuits de distribution d'eau.

36 elektorcardi♥scope Android sans fil, sans bouton : Bluetooth & écran tactile 1^{ère} partie

Tablette ou téléphone Android comme terminal sans fil de visualisation des électrocardiogrammes.

46 niveau à bulle acoustique



100 thermo-hygromètre

Riches possibilités réunies en un seul circuit, dont certaines réservées aux instruments professionnels.

112 sauve qui peut ! version 2.0

Conserver les paramètres d'un appareil lors d'une coupure de courant inopinée.

114 retour sur des particularités du LCR-mètre 0,05 %

Quelques mises au point et des précisions.

● industrie

108 gravure de CI professionnels

Voyez comment *elektorPCBservice* prépare vos cartes à quatre couches!

● Tech the Future

106 neurosciences cognitives et musique

Mieux observer le cerveau pour mieux le comprendre

● magazine

8 savez-vous programmer des applications Android ?

Ce n'est pas difficile du tout...

120 les artisans de l'électronique

Les tribulations au Mexique d'un des pères fondateurs d'Arduino.

126 Rétronique

Konrad Zuse, ses Z1 à Z4 et ...

129 Hexadoku

Casse-tête pour électroniciens

130 bientôt dans Elektor

Avant-première

ATtiny45 à double emploi, autour du capteur d'inclinaison MMA7260.

48 1 bouton + 1 LED = 2 fils seulement

Une astuce encore plus raffinée!

52 oscillateur à large bande à pont de Wien

N'utilise pas de potentiomètre à double piste.

54 contrôleur de charge 4 A pour panneau solaire

Soit une puissance d'environ 75 W.

58 pilote *sumo* pour moteurs CC

Supporte jusqu'à 40 A avec toutes sortes de moteurs CC.

60 convertisseur élévateur MLI

42 V / 1 A à partir de 8 à 16 V.

64 limiteur de courant d'appel

Pour soulager les circuits.

68 chargeur de secours pour batterie de téléphone

Elektor USB *Power Pack*.

72 huit relais et plus encore

Pour les cartes à µC Linux et autres.

76 pilote de moteur pas à pas

Pour ElektorBus.

84 émetteur FM large spectre en bande 70 cm

Puissance : 130 mW.

91 veilleuse solaire

Avec accu Li-ion de secours.

92 commande de cartes par PC

Mainmise du PC sur l'électronique par SPI et UART interposés.

ELEKTOR / PUBLITRONIC SARL

c/o Regus Roissy CDG

1, rue de la Haye

BP 12910

FR - 95731 Roissy CDG Cedex

Tél. : (+33) 01.49.19.26.19

lundi, mardi et jeudi de 8h30 à 12h30

Fax : (+33) 01.49.19.22.37

www.elektor.fr

Banque ABN AMRO : Paris

IBAN : FR76 1873 9000 0100 2007 9702 603

BIC : ABNAFRPP

DROITS D'AUTEUR :

© 2013 Elektor International Media B.V.

Toute reproduction ou représentation intégrale ou partielle, par quelque procédé que ce soit, des pages publiées dans la présente publication, faite sans l'autorisation de l'éditeur est illicite et constitue une contrefaçon. Seules sont autorisées, d'une part, les reproductions strictement réservées à l'usage privé du copiste et non destinées à une utilisation collective, et, d'autre part, les analyses et courtes citations justifiées par le caractère scientifique ou d'information de l'oeuvre dans laquelle elles sont incorporées (Loi du 11 mars 1957 -art. 40 et 41 et Code Pénal art. 425).

Certains circuits, dispositifs, composants, etc. décrits dans cette revue peuvent bénéficier de droits propres aux brevets; la Société éditrice n'accepte aucune responsabilité du fait de l'absence de mention à ce sujet. Conformément à l'art. 30 de la Loi sur les Brevets, les circuits et schémas publiés dans Elektor ne peuvent être réalisés que dans des buts privés ou scientifiques et non commerciaux. L'utilisation des schémas n'implique aucune responsabilité de la part de la Société éditrice. La Société éditrice n'est pas tenue de renvoyer des articles qui lui parviennent sans demande de sa part et qu'elle n'accepte pas pour publication. Si la Société éditrice accepte pour publication un article qui lui est envoyé, elle est en droit de l'amender et/ou de le faire amender à ses frais; la Société éditrice est de même en droit de traduire et/ou de faire traduire un article et de l'utiliser pour ses autres éditions et activités, contre la rémunération en usage chez elle.

Elektor est édité par Elektor International Media B.V.
Siège social : Allee 1 - 6141 AV Limbricht, Pays-Bas

Imprimé aux Pays-Bas
par Senefelder Misset - Doetinchem

Distribué en France par M.L.P.
et en Belgique par A.M.P.

Hors-gabarit 2013

Si vous êtes nouveau lecteur, vous serez sans doute surpris par le format particulier de cette édition d'Elektor dite *hors-gabarit*, ou *numéro double* ou *d'été*. Avec ses 132 pages et son contenu enrichi, elle a l'ambition de surclasser l'offre habituelle. Elle ne ressemble pas non plus aux éditions estivales des années passées, ce qui surprendra sûrement nos lecteurs de la première heure. À la place de la ribambelle hétérogène de petits circuits, le labo d'Elektor a concocté



pour cet été un mélange équilibré de montages et de projets de toutes tailles. Nous conjuguons quantité et qualité pour proposer des circuits vraiment pratiques, c'est-à-dire avec pour chacun un (dessin de) circuit imprimé dûment testé. Une garantie de fiabilité d'autant plus cruciale que plusieurs de ces projets sont liés aux alimentations et aux circuits de puissance, aussi pour moteurs. On remarquera le **limiteur de courant d'appel**, le **chargeur de batterie de secours pour téléphone** ou le **convertisseur élévateur MLI**.

Si le circuit amiral de ce numéro double, le nouvel **elektorcardi♥scope Android**, n'a pas (encore) de circuit imprimé, c'est parce que celui-ci sera présenté le mois prochain. Oui, nous reviendrons en long et en large sur la réalisation de cette petite merveille imaginée et conçue par Marcel Cremmel, comme le fut le célèbre électrocardiographe d'Elektor à base de *GameBoy*. Elektor est d'autant plus heureux d'accueillir des projets de cette envergure qu'ils sont proposés par des auteurs français. Cocorico !

Deux autres morceaux de choix de cette édition sont **l'émetteur FM** et le **télécommande par smartphone Android**, l'une et l'autre avec circuits imprimés.

Cette profusion de sujets élaborés nous a contraints à remettre au numéro de septembre 2013 la parution du 3^e épisode de la série consacrée à Python. D'ici là, nous vous souhaitons un bel été, riche de bonnes lectures et de bonne électronique ! Ce n'est pas ce qui manque ici.

Denis Meyer

Notre équipe

Rédacteur en chef :	Denis Meyer (redaction@elektor.fr)
Directeur éditorial :	Wisse Hettinga
Rédaction internationale :	Harry Baggen, Jan Buiting, Thijs Beckers, Eduardo Corral, Jens Nickel, Clemens Valens
Laboratoire :	Thijs Beckers, Ton Giesberts, Luc Lemmens, Tim Uiterwijk, Jan Visser
Ont coopéré à ce numéro :	Jean-Paul Brodier, Robert Grignard, Hervé Moreau, Kévin Petit, NN
Service de la clientèle :	Jolanda van Kruchten
Graphistes :	Giel Dols, Jeanine Opreij, Mart Schroijen
Elektor <i>online</i> :	Daniëlle Mertens & Patrick Wielders

**France**

Denis Meyer
+31 46 4389435
d.meyer@elektor.fr

**United Kingdom**

Wisse Hettinga
+31 (0)46 4389428
w.hettinga@elektor.com

**USA**

Hugo Vanhaecke
+1 860-875-2199
h.vanhaecke@elektor.com

**Germany**

Ferdinand te Walvaart
+49 241 88 909-17
f.tewalvaart@elektor.de

**Netherlands**

Harry Baggen
+31 46 4389429
h.baggen@elektor.nl

**Spain**

Eduardo Corral
+34 91 101 93 95
e.corral@elektor.es

**Italy**

Maurizio del Corso
+39 2.66504755
m.delcorso@inware.it

**Sweden**

Wisse Hettinga
+31 46 4389428
w.hettinga@elektor.com

**Brazil**

João Martins
+55 11 4195 0363
joao.martins@editorialbolina.com

**Portugal**

João Martins
+351 21413-1600
joao.martins@editorialbolina.com

**India**

Sunil D. Malekar
+91 9833168815
ts@elektor.in

**Russia**

Nataliya Melnikova
+7 (965) 395 33 36
Elektor.Russia@gmail.com

**Turkey**

Zeynep Köksal
+90 532 277 48 26
zkoksal@beti.com.tr

**South Africa**

Johan Dijk
+31 6 1589 4245
j.dijk@elektor.com

**China**

Cees Baay
+86 21 6445 2811
CeesBaay@gmail.com

Notre réseau



VOICE COIL

CIRCUIT CELLAR
THE WORLD'S SOURCE FOR QUALITY ELECTRONIC ENGINEERING SOLUTIONS

audio X PRESS



vous connecte à



Nos annonceurs



Beta Layout

www.pcb-pool.com 9

HAMEG

www.hameg.fr 84

CadSoft

www.cadsoft.de 41

National Instruments

www.ni.com/labview-platform/f . . . 2

Eurocircuits

www.elektorpcbservice.com 81

Pour placer votre annonce dans le prochain numéro d'Elektor

veuillez contacter Mme Ilham Mohammadi par téléphone au (+31) 6.41.42.25.25
ou par courrier électronique : i.mohammadi@elektor.fr

Vos correspondants

Nous sommes à votre service pour toute question relative à votre commande ou votre abonnement
par téléphone au (+33) 01.49.19.26.19 lundi, mardi et jeudi de 8h30 à 12h30
ou par courriel : service@elektor.fr

savez-vous ... programmer des applis sous Android ?



Mariline Thiebaut (Elektor)

Ils ont beau être *smart*, nous rêvons de rendre nos téléphones encore plus intelligents, plus utiles ou plus séduisants.

Seulement pour cela il faut des applications. Alors écrivons-les nous-mêmes !

En fait, un court apprentissage suffit, que l'on soit familier ou non du langage Java. Brique par brique, au fil d'exemples simples et concrets, un nouveau livre publié par Elektor vous apprendra à construire des applications Android et à tirer profit des incroyables possibilités de ce système d'exploitation.

ISBN 978-2-86661-191 0

main » aux nouveaux outils. Il aborde des principes essentiels comme **l'affichage de texte, les champs de saisie, les boutons, les listes de choix, les fonctions de calcul...**

savez-vous... cuisiner une (Calcul)OhmLette ?

Les premiers chapitres du livre permettent au lecteur de découvrir les outils nécessaires pour programmer sous Android : concepts de base de la programmation sous Android, interface utilisateur Android et Java.

La première appli proposée calcule la valeur d'une grandeur électrique (courant, tension, résistance) à partir de la loi d'Ohm : *CalculOhmLette*. C'est un programme simple mais qui permet de « se faire la

savez-vous... déclencher une alarme ?

Deux applis s'intéressent à la gestion du temps : **chronomètre et minuteur**. C'est l'occasion d'apprendre comment afficher un message dans la barre d'état située en haut de l'écran ou comment allumer la LED de notification lorsque la durée réglée sur le minuteur est écoulée. L'auteur montre également comment enregistrer ces messages d'alarme dans un fichier ou les transmettre par SMS. Le lecteur dispose ainsi de fonctions qu'il pourra adapter à ses propres applis.

**savez-vous...
exploiter les capteurs ?**

Les tablettes et ordiphones Android disposent de différents capteurs : GPS, accéléromètre, magnétomètre, capteur de luminosité. L'auteur explique comment récupérer et exploiter les données fournies par ces capteurs à travers diverses applis. C'est aussi l'occasion de mettre en œuvre les fonctions de **dessin sur l'écran** (*Draw*) pour représenter graphiquement les données récupérées et d'apprendre à **créer des App Widgets** (composants graphiques).

**savez-vous...
produire/enregistrer du son ?**

Un appareil Android muet : c'est tout simplement impensable ! L'auteur explique bien sûr comment lire un fichier MP3 mais propose également un générateur de fréquences. Le tout est complété par **une appli d'enregistrement de son**.

**savez-vous...
communiquer par l'internet ?**

L'une des principales applications des téléphones tactiles est l'utilisation de l'internet. C'est pourquoi l'auteur consacre un chapitre entier aux applis pour le web. Il montre comment afficher sur son téléphone les images transmises par une webcam ou par Google Maps. La manipulation des vidéos est également abordée. Pour aller plus loin, il propose entre autres **une appli** en JavaScript et HTML **qui interroge plusieurs caméras IP** depuis un téléphone ainsi qu'**une appli qui interroge une base de données** via HTTP (base de données SQL et script PHP). Vous l'aurez compris, les applis de ce livre sont variées. Leur combinaison permet de créer toutes sortes d'applications.

**Question récurrente :
pour quelle version d'Android ?**

Les programmes proposés dans le livre sont compatibles avec toutes les versions d'Android. C'est le programmeur qui définit dans le fichier *Manifeste* avec quelles versions d'Android sera compatible son appli (paramètre « `uses-sdk android:minSdkVersion=` »). Tout ceci est expliqué en détails dans le livre. Par exemple, le premier programme du livre repose sur Android 4.0 mais c'est très simple de l'adapter pour le faire tourner sur 2.3 ou toute autre version.

Liens

Extraits, sommaire, code du livre : www.elektor.fr/android



L'auteur : Stefan Schwark
Après une formation en électronique et électrotechnique, Stephan Schwark, l'auteur, a travaillé plusieurs années comme concepteur de matériel et programmeur. Aujourd'hui il programme des systèmes de commande et de supervision de piles à combustible. Il a déjà publié (en allemand) un premier ouvrage sur la programmation des téléphones mobiles (*Java ME pour les mobiles : programmer, mesurer, afficher*). Dans son dernier numéro récemment paru, le magazine **Android MT** publie **un entretien avec Stefan Schwark** (voir ci-dessus).

célébrons Sainte-Aise !

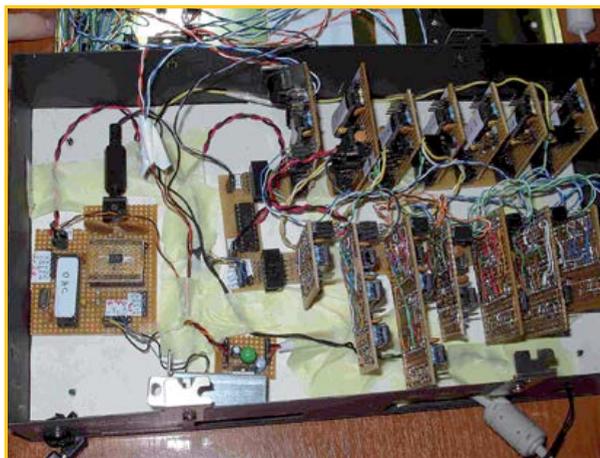
Clemens Valens
(Elektor.Labs)

Quand on aime, on ne compte pas. Enfin si... il a bien fallu compter pour constater qu'Elektor.LABS réunit à présent plus de 5.000 membres enregistrés. Merci à tous ! Nous célébrons les participants les plus zélés et invitons les autres à **faire le pas*...**



Le retour du Formant

Bien des projets postés sur .LABS se résument à un texte succinct, une photo ou un schéma. Tel n'est pas le cas de la contribution de l'OP *AChorevas*. Comme il n'était pas membre de la communauté d'Elektor, celui-ci nous a demandé de poster à sa place sur .LABS son projet **d-Formant**. Ce *d* vient de l'anglais **digital**, et *Formant* est le nom d'un légendaire synthétiseur de musique, modulaire et analogique, publié par Elektor dans les années 1970.



D'emblée le *d-Formant* semblait intéressant et son auteur méritait un accès gratuit au site .labs mais jamais nous ne nous serions attendus à une avalanche de 15 messages, avec description détaillée, échantillons sonores, schémas et code source.

Le *d-Formant* est un synthétiseur intégralement numérique. Les signaux analogiques du Formant original y sont remplacés par des signaux numériques de 16 bits ; aux transistors et amplis op de l'original se substituent ici des PIC24. La structure reste modulaire. On la recâble à la volée, ce qui offre toutes les possibilités d'interconnexion à chaud du Formant. Les diverses commandes de l'original peuvent être implantées sur le *d-Formant* quand bien même son interface est beaucoup plus simple et meilleur marché que celle de l'original : un afficheur LCD, quelques poussoirs et un codeur rotatif. Le *d-Formant* s'accommode de n'importe quel clavier MIDI standard et plus généralement de toute interface MIDI capable d'émettre des commandes *note on* et *note off*.

En attendant que ce projet débouche sur un article forcément passionnant publié dans le magazine, notre ami *AChorevas* a été récompensé par une carte de membre Elektor Gold. Félicitations !

www.elektor-labs.com/node/3124

(130098-1)

* Il n'y a que le premier pas qui coûte (proverbe aveyronnais)

www.elektor-labs.com

OP (pour Original Poster) désigne ici l'auteur d'un projet ou l'initiateur d'une discussion sur .labs. Les OP désireux de se voir publiés dans le magazine Elektor sont invités à relever régulièrement leur courrier à l'adresse avec laquelle ils se sont inscrits sur .labs, c'est notre seul moyen de communication avec eux.

dessoudage de circuits intégrés CMS

Ce n'est pas parce qu'un problème paraît complexe que sa solution ne pourra pas être simple. Et bon marché ! Notre collègue *Lucky Luc*, du labo d'Elektor, a trouvé sur la Toile une astuce efficace et élégante pour dessolder facilement les circuits intégrés CMS. Pour le fabriquer, il suffit d'une pince de bonne qualité, d'un petit morceau de fil électrique de cuivre de forte section, d'un gros fer à souder et d'un peu d'étain. Ah, oui, il faudrait aussi une pincette.

Pour les CMS à deux rangées de broches de part et d'autre du boîtier, la conformation à donner au fil de cuivre sera celle de la première photo. Soignez la rectitude de la portion de fil qui touche les broches à dessolder, de sorte que toutes soient chauffées également. Chauffez le fil et étamez-le là où doit avoir lieu le transfert de chaleur (deuxième photo). La présence d'étain favorise l'échauffement des broches au contact du cuivre. Puis appuyez le fil de cuivre contre les broches comme sur la troisième photo et faites-le chauf-

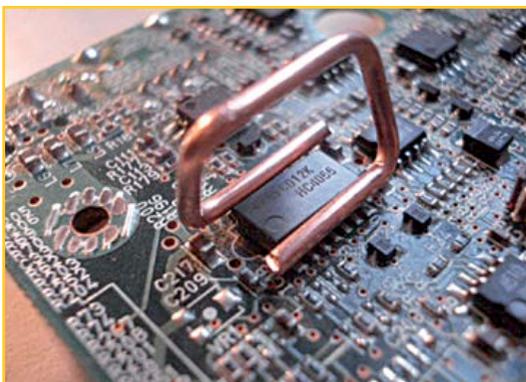
fer avec un fer puissant, en veillant sans cesse à ce que les broches de part et d'autre soient en contact avec le fil. Dès que la soudure est en fusion, retirez vivement le circuit intégré de la carte à l'aide d'une pincette.

Ce procédé convient aussi pour les circuits intégrés dont les broches sont placées sur quatre côtés. Il suffit d'adapter le coupage du fil de cuivre comme on le voit sur la quatrième illustration. Attention à la surchauffe, elle pourrait démolir le circuit intégré lui-même qui aurait alors le peu enviable statut DBR (*damaged beyond repair*) c'est-à-dire HS en jargon d'ici. Il faut aussi ménager les pistes du circuit imprimé pour qu'elles ne se décollent pas. C'est un coup de main à (ap)prendre. Avant de vous attaquer au circuit sur lequel vous devez intervenir, faites des essais sur un martyr.

(130099)

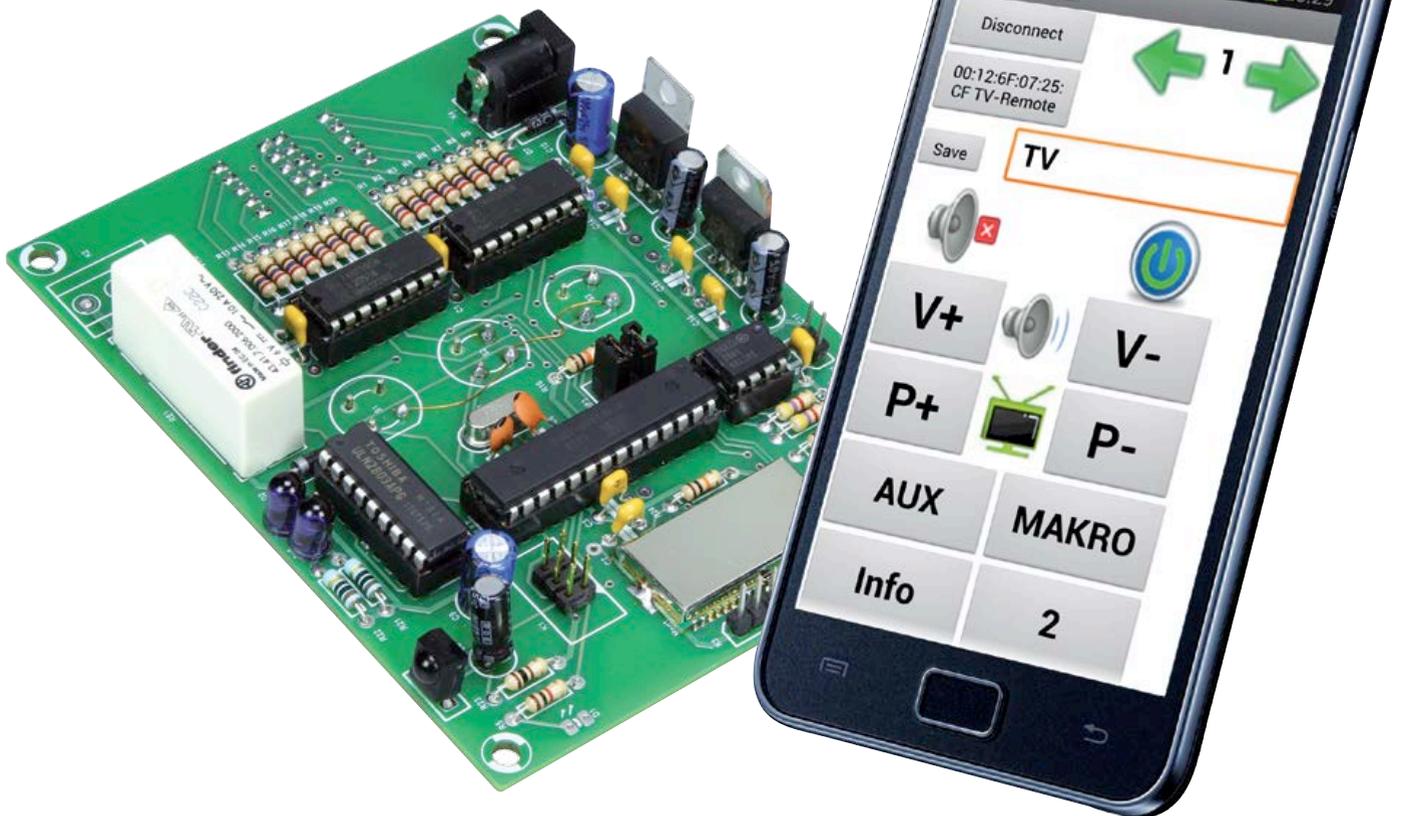
Vu sur <http://youtu.be/dCUSwADP6DE>

Thijs Beckers
(Elektor)



télécommande par *smartphone*

Émetteur et application pour appareils Android



Peter Zirngibl
(info@pezitec.com)

- Chérie, passe-moi la télécommande, s'il te plaît
- Quelle télécommande ?... Mais c'est *mon* téléphone !
- Ben oui, mais c'est *notre* télécommande...

Ou comment détourner une merveille multifonctions et en faire une télécommande pour le matériel audio-vidéo domestique ?

Avec un téléphone tactile, on peut naviguer sur l'internet, envoyer et recevoir des courriels et des textos, *chatter*, enregistrer et écouter du son, créer et regarder des photos et des vidéos, écouter la radio, regarder la télé, jouer et même – oui, oui – téléphoner. Des dizaines de milliers d'appli(cation)s en font le couteau suisse

qui fait le café et la paix dans le monde. Il est donc tout naturel qu'il existe aussi des applis qui permettent la télécommande par *smartphone* du matériel audio-vidéo moderne – à condition que ce matériel soit connectable au réseau informatique local. Les appareils qui ne possèdent qu'une télécommande infrarouge à l'ancienne, et pas de

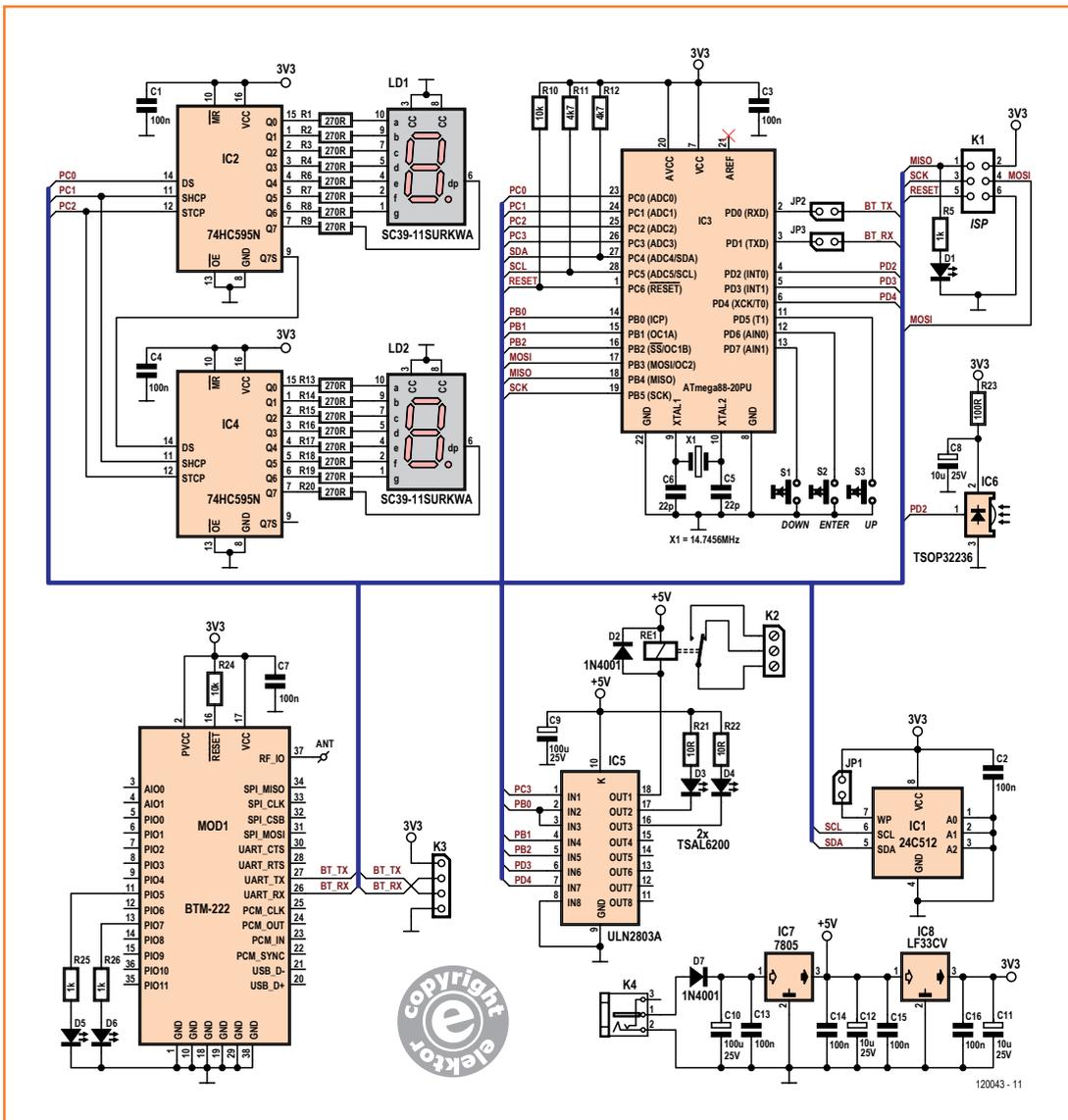


Figure 1. La dépense en matériel reste limitée grâce au module Bluetooth.

connexion à un réseau, ne peuvent pas être pilotés directement par téléphone tactile ; celui-ci a beau être *smart*, il n'a pas (encore) d'émetteur infrarouge.

Ce qui nous manque, c'est un adaptateur intelligent qui communique d'un côté par Bluetooth avec le *smartphone*, de l'autre par infrarouges avec l'appareil à commander. Intelligent signifie qu'un processus d'apprentissage préalable est nécessaire. Sinon comment l'adaptateur saurait-il de quel type de télécommande il s'agit, et quelle opération doit être déclenchée par telle commande ? On trouve certes dans le commerce des télécommandes universelles simples et bon

marché, pré-programmées pour des milliers d'appareils, mais elles ne connaissent que quelques commandes standard comme PLAY ou STOP. Les commandes ne sont pas modifiables et la fonction d'apprentissage n'est disponible que sur les modèles les plus chers.

Quelquefois, dans les modèles de luxe, il est possible de saisir des macro-commandes, des séquences de commandes individuelles fréquemment émises à la suite (par exemple : visionner un film d'un DVD = mise en marche de : lecteur DVD, télé, récepteur AV, réglage correct des canaux vidéo et audio). Cela fera partie des possibilités de notre adaptateur.

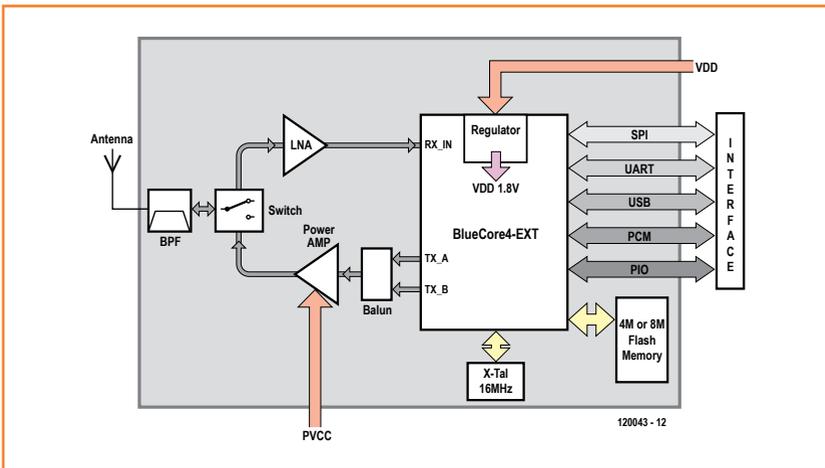
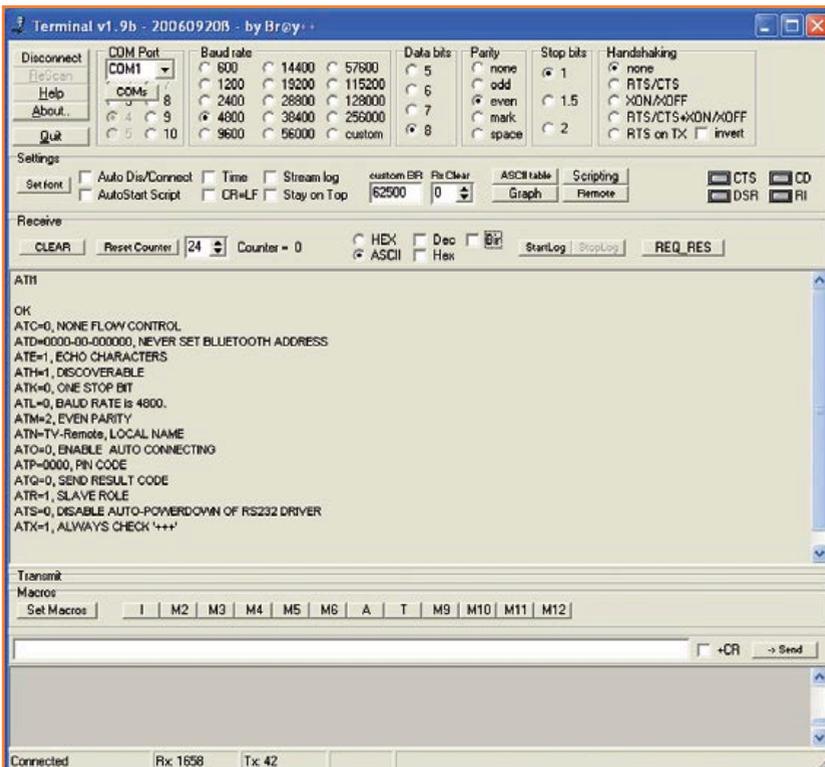


Figure 2.
Schéma synoptique du module Bluetooth BTM-222.

Figure 3.
Le module Bluetooth doit être configuré avec les paramètres indiqués au moyen d'un programme de terminal.

Les télécommandes universelles de luxe d'aujourd'hui se programment confortablement avec un PC. Pour ne pas compliquer la programmation, nous nous passerons de pré-programmation et de PC. L'adaptateur apprendra toutes les commandes nécessaires à partir de la télécommande d'origine. Cela signifie qu'on tient la diode émettrice de la télécommande devant le récepteur de l'adaptateur pour l'acquisition des commandes. Nous y reviendrons après un coup d'œil sur le matériel.



Microcontrôleur avec liaison radio

Le schéma (fig. 1) réunit un petit μC Atmel ATmega88 et quelques groupes de composants, à savoir :

- une EEPROM 24C512 de 512 Kbits avec interface I²C pour le stockage des codes programmés ;
- un module récepteur infrarouge TSOP3226 ;
- deux LED IR TSAL6200 avec un pilote de puissance ULN2803 ;
- un affichage à deux chiffres à 7 segments avec un registre à décalage HC595 pour l'affichage du code programmé et de l'état de la programmation ;
- trois touches UP, DOWN et ENTER utilisées pour la programmation ;
- une interface de programmation *in situ* (ISP) K1 ;
- un relais avec un contact inverseur, piloté également par l'ULN2803 et
- des régulateurs de tension +5 V et +3,3 V.

Rien d'exceptionnel. Le clou du circuit est le module Bluetooth BM222 de Rayson [1]. Il est petit, pratique, relativement simple à utiliser et avant tout abordable, avec son prix d'environ 10 €. En ces temps où les ordinateurs ont de moins en moins d'interfaces par câble, un tel module offre la possibilité de piloter des périphériques. Le module BTM-222 est un appareil de classe 1 avec une portée maximale de 100 m. Il offre (fig. 2) plusieurs interfaces sérieelles, parmi lesquelles l'UART, destinée à la communication avec les microcontrôleurs. Le débit garanti par le module à travers l'UART (et l'interface USB) est de 921 kbits/s. Si le module BTM-222 ne doit communiquer avec le μC que par l'UART, il peut fonctionner sans circuit externe. Le paramétrage sera classique : débit 19200 bauds, 8 bits de données, pas de bit de parité, 1 bit d'arrêt.

Il suffit pour cela que le microcontrôleur ouvre le canal UART correspondant. En cas de besoin, les paramètres et d'autres propriétés du module peuvent être adaptés au moyen des commandes dites « AT » [1]. La configuration est mémorisée dans une mémoire flash interne. Le noyau « bleu » du module est cadencé à 16 MHz par un oscillateur interne. Le signal de sortie est amené à l'antenne au niveau de +18 dBm par un réseau symétrique (balun) et un amplificateur de puissance. Pour la réception, le BTM-222 commute

l'antenne sur l'amplificateur à faible bruit (*Low Noise Amplifier*). Un filtre passe-bande améliore la qualité de réception. Le BTM-222 ne possède pas d'antenne externe, mais on peut la réaliser avec une petite longueur de piste imprimée ou (comme ici) un petit fil.

Le module comporte des sorties qui signalent l'état des données (LED D5 sur la broche 11) et de la liaison (LED D6 sur la broche 13). De plus, on peut (ce n'est pas fait ici) connecter une LED à la broche 14 comme témoin d'alimentation.

Mise en service du module BM-222

Commencez par assigner au module *Bluetooth* les paramètres corrects. Pour cela, retirez les cavaliers JP2 et JP3, de façon à couper la communication sérielle avec le µC. Connectez le BTM-222 au PC, par exemple avec le câble TTL232R de FTDI sur K3. La ligne VDD de ce câble véhicule une tension de 5 V qui ne doit pas parvenir au BTM-222 (sa tension d'alimentation – valeur nominale 3,3 V – ne doit jamais dépasser 3,6 V). On laisse donc simplement la broche 1 de K3 sans connexion et on alimente le module *Bluetooth* par la platine.

Lancez sur votre PC un programme de terminal comme *Hyperterminal* ou *Hterm* (**fig. 3**), choisissez le port COM (virtuel) auquel est relié le module et assignez à la liaison sérielle les paramètres d'usine listés plus haut. Vérifiez si le module réagit à la commande AT11. Cette commande doit renvoyer à l'écran les réglages du module. S'il n'y a aucune réponse, vérifiez l'alimentation 3,3 V, la configuration du port COM et le raccordement à K3 (inversion Tx/D/RxD ?). Si vous n'êtes pas sûr des paramètres actuels du module ou si quelque chose s'est mal passé lors de l'installation, vous pouvez recharger dans le module les réglages d'usine en maintenant pendant trois secondes au minimum PIO4 au niveau haut. Passez alors les réglages de l'UART à 4800 bauds (ATL = 0), parité paire (ATM = 2), pas de dialogue (ATC = 0). Accordez la configuration du terminal. Modifiez ensuite à votre guise le nom du module ou le code PIN de la liaison *Bluetooth*. Allumez maintenant le téléphone ou la tablette tactile Android, assurez-vous que le module BTM-222 a été trouvé et établissez la liaison avec le code PIN. Si tout fonctionne, arrêtez le circuit de télécommande, déconnectez le PC et placez les cavaliers JP2 et JP3.

L'application Remote

Une fois l'application *Remote_Control.APK* installée, le téléphone Android est converti en télécommande. Lancez l'application et procédez à l'appariement avec le BTM-222. Choisissez le module *Bluetooth* avec *SELECT DEVICE* et établissez une liaison avec *CONNECT*. La LED D6 est allumée en permanence pour montrer que le module est connecté à l'appareil Android. Le clignotement de D5 signale la réception de données provenant de l'appareil Android. L'adresse mémoire à exécuter est communiquée à l'adaptateur par un nombre (de 0 à 99) suivi d'un saut de ligne (/n). En mode exécution, les commandes reçues par le module *Bluetooth* peuvent être exploitées. Le principe est très simple : à la réception de « 2/n », l'adaptateur saute à l'adresse 2 de la mémoire et exécute la commande qui y est stockée (ce qui augmente le volume de l'appareil 1). Les commandes 73 et 74 présentent une particularité : aucun signal IR n'est émis, mais le relais Re1 est excité pendant 2 s (73) ou inversé (74). On peut raccorder un appareil à K2 et le mettre en marche ou l'arrêter.

La **figure 4** montre l'appli dans toute sa splendeur sur le *smartphone*. Les flèches du haut sélectionnent un appareil parmi cinq (1 à 5). On peut donner à chacun un nom dans la fenêtre de texte et l'enregistrer avec *SAVE*. Le nom est sauvegardé et s'affichera au prochain démarrage de l'appli. On peut aussi étiqueter les fonctions 1 à 10. Pour cela on appuie sur le bouton *EDIT*, on choisit la touche voulue et on saisit le texte. Une pression sur *SAVE* enregistre le texte. Les textes sont communs à tous les appareils.



Figure 4. L'aspect de l'application sur le smartphone.

Attention, enregistrement !

Le logiciel du μC se divise en deux modes : programmation et exécution. Le mode programmation enregistre les codes infrarouges d'une télécommande. Voici le processus :

Il faut d'abord effacer l'EEPROM sérielle. Pour cela, appuyer sur les trois touches simultanément et mettre sous tension. Quand le message « dE. » (*delete*) apparaît sur les afficheurs à sept segments, on peut relâcher les touches. L'opération d'effacement dure quelques minutes, puis le message « 00 » s'affiche.

Pour passer au mode de programmation et enregistrer les commandes, on maintient enfoncée la touche *ENTER* (port D6 à la masse) lors de la mise sous tension. Le message « Pr. » clignote trois fois sur les afficheurs à sept segments. Il faut maintenant choisir une mémoire entre 0 et 99 avec les touches *UP/DOWN* (ports D5 et D7)

et confirmer avec *ENTER*. L'afficheur indique par « — » que le circuit est prêt à enregistrer. Tenir la télécommande à une distance de 5 à 10 cm devant le récepteur infrarouge et appuyer une fois (une fois !) sur la fonction désirée. Pour éviter d'enregistrer des signaux parasites, ce processus doit être exécuté rapidement. L'affichage de « io. » confirme la programmation. Pour chaque appareil, il est possible d'enregistrer une macrocommande composée de six commandes au maximum. Les macros sont enregistrées comme les autres commandes.

Lors de l'enregistrement des commandes infrarouges, la durée des impulsions est mesurée et mémorisée temporairement comme une variable au format entier sans signe, puis stockée en EEPROM. Si on veut programmer une nouvelle commande, il faut maintenir la touche *ENTER* enfoncée pendant la confirmation (« io. » cli-

Liste des composants

Résistances

R1 à R4, R6 à R9, R13 à R20 = 270 Ω
 R5, R25, R26 = 1 k Ω
 R10, R24 = 10 k Ω
 R11, R12 = 4,7k Ω
 R21, R22 = 10 Ω
 R23 = 100 Ω

Condensateurs

C1, C2, C3, C4, C7, C13, C14, C15, C16 = 100 nF
 C5, C6 = 22 pF céramique
 C8, C11, C12 = 10 $\mu\text{F}/25\text{ V}$ radial
 C9, C10 = 100 $\mu\text{F}/25\text{ V}$ radial

Semi-conducteurs

D1, D5, D6 = LED rouge, 3 mm, haut rendement
 D2, D7 = 1N4001
 D3, D4 = LED infrarouge 940 nm TSAL6200 (Vishay)
 LD1, LD2 = afficheur LED 7 segments 10 mm (Kingbright SC39-11SURKWA)
 IC1 = CAT24C512LI-G (On Semiconductor)
 IC2, IC4 = 74HC595N
 IC3 = ATmega88-20PU (Atmel), programmé : 120043-41 [2]
 IC5 = ULN2803APG
 IC6 = récepteur infrarouge 36 kHz TSOP32236 (Vishay)
 IC7 = 7805
 IC8 = LF33CV (ST)
 Mod1 = module Bluetooth BTM-222 (Rayson)

Divers

X1 = quartz 14,7456 MHz
 Re1 = relais 6V 1 RT, Finder 43.41.7.006.2000
 S1 à S3 = touche tactile D6 ronde

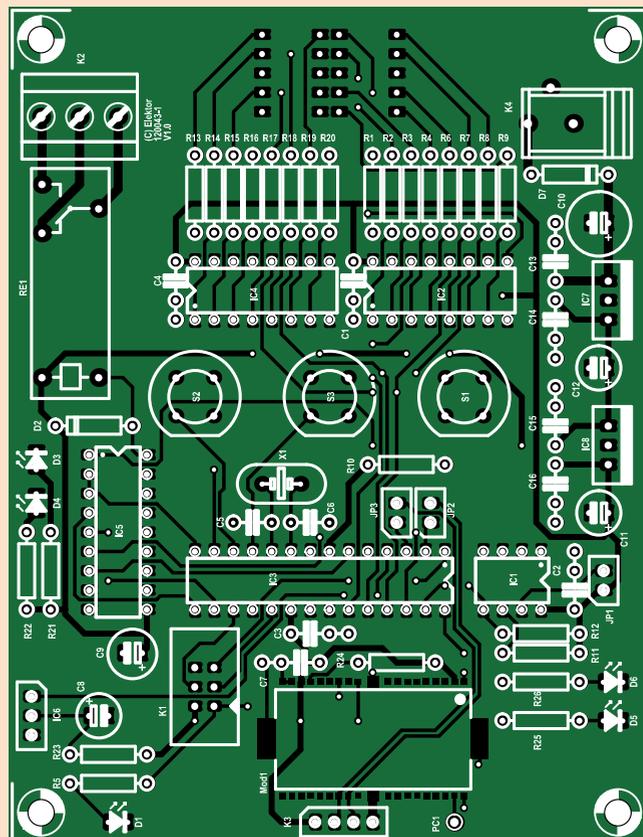


Figure 5.
 Le circuit imprimé. Tous les organes de commande sont montés du côté soudures.

gnotant), sinon le μC est remis à zéro après la programmation.

Si aucune touche n'est pressée pendant la remise à zéro, le programme passe automatiquement en mode exécution. Dans ce mode, on choisit une des commandes enregistrées avec les touches *UP/DOWN* et on l'émet par la touche *ENTER* via la LED infrarouge. On peut vérifier de cette façon si la commande est exécutée correctement.

Circuit imprimé

La platine de l'adaptateur de télécommande est disponible chez Elektor [2], naturellement, de même que le μC programmé. Vous trouverez par le même lien l'application compilée et son code source. Un coup d'œil au dessin de la **figure 5** calme un peu les inquiétudes : certes, le module *Bluetooth* se monte en surface, mais heureusement ses broches sont au pas de 1,25 mm, encore

compatible avec le fer à souder. Cela devrait permettre de le souder sur le circuit imprimé, avec le soin nécessaire, mais sans gros problème. Reste une petite difficulté pour l'orientation du module : la broche 1 est repérée par un petit point rond dans le métal du blindage (du côté de l'antenne). Ce point doit être orienté vers PC1 sur la platine. Les gros îlots au milieu des quatre côtés sont au potentiel de la masse et reliés à la piste de masse qui entoure le module.

Le module une fois bien soudé, on peut installer les autres composants, tous traversants. Les circuits intégrés seront montés dans des supports. Les deux afficheurs, les trois touches et les diodes D5-D6 se montent en dessous de la platine. Ainsi, toutes les « interfaces homme-machine » (HID) se trouvent d'un côté et la platine s'installe facilement dans un coffret. Un simple fil tient lieu d'antenne. Le calcul lui assigne pour 2,4 GHz une

JP1 à JP3 = barrette droite 2 points, avec cavalier
 K1 = barrette double rangée 2x3 points
 K2 = bornier 3 points, pas 5 mm
 K3 = connecteur SIL 4 points

K4 = embase pour l'alimentation 2,1 mm
 circuit imprimé 120043-1 [2]

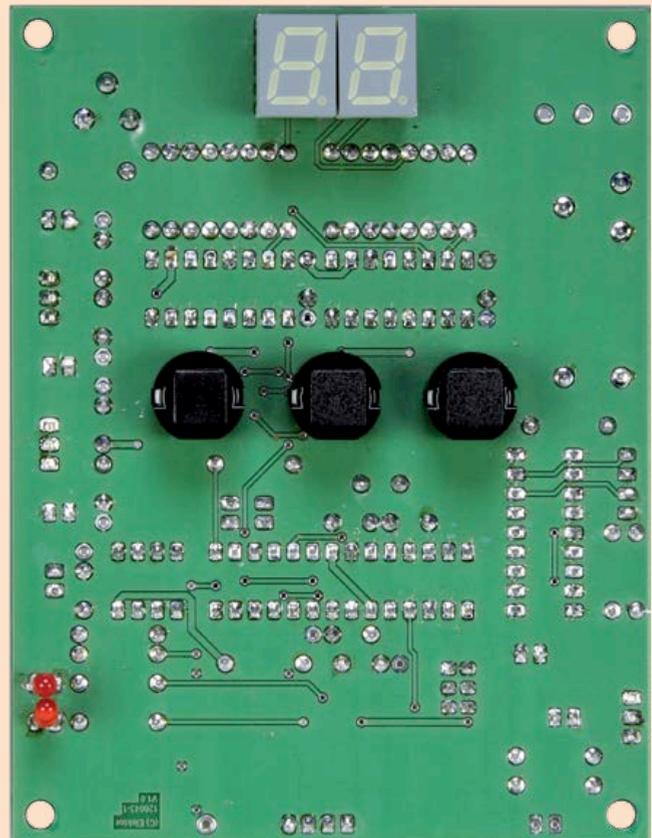
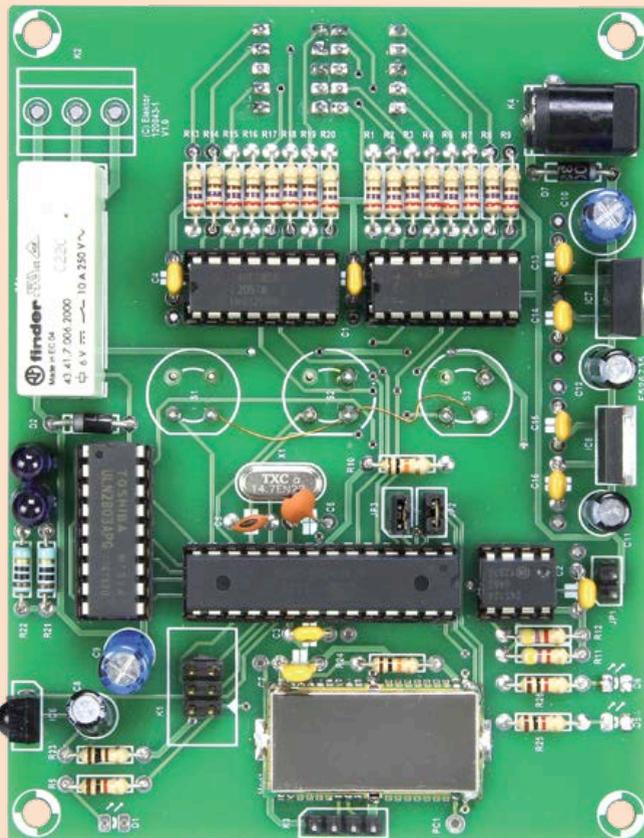


Tableau 1. Correspondance entre mémoires et fonctions

fonctions	appareil				
	1	2	3	4	5
MUTE	0	8	16	24	32
ON/OFF	1	9	17	25	33
VOL+	2	10	17	26	34
VOL-	3	11	19	27	35
PRG+	4	12	20	28	36
PRG-	5	13	21	29	37
AUX	6	14	22	30	38
MACRO	7, 75-79	15, 80-84	23, 85-89	31,90-94	39, 95-99
fonction 1	40	50	60		
fonction 2	41	51	61		
fonction 3	42	52	62		
fonction 4	43	53	63		
fonction 5	44	54	64		
fonction 6	45	55	65		70
fonction 7	46	56	66		71
fonction 8	47	57	67		72
fonction 9	48	58	68		73 ¹⁾
fonction 10	49	59	69		74 ²⁾

¹⁾ relais activé pendant 2 secondes ²⁾ basculement du relais

longueur de 3,1 mm ($\lambda/4$), mais cette valeur n'est pas critique, d'autant moins que cette application n'a pas besoin d'une grande portée. L'essentiel est que l'antenne dépasse un peu si le coffret est métallique.

(120043 – version française : Jean-Paul Brodier)

Liens

- [1] www.mikrocontroller.net/wikifiles/f/fc/BTM-222_DataSheet.pdf
- [2] www.elektor.fr/120043
- [3] <http://appinventor.mit.edu>

Modification de l'application

L'application a été développée avec *App-Inventor* du MIT [3]. Si vous voulez modifier ou adapter quelque chose, il vous faudra ouvrir un compte Gmail et vous inscrire en suivant le lien [3]. Téléchargez *Remote_Control.ZIP* sous *My Projects* dans *App-Inventor*, sous *More Options* → *Upload Source*. L'application peut être modifiée, compilée en ligne, puis le fichier résultant *Remote_Control.APK* peut être téléchargé sur le PC ou installé dans l'appareil Android.





L'ORIGINAL DEPUIS 1994

PCB-POOL[®]

Beta LAYOUT

Le nouveau standard

Finition ENIG au lieu de l'étamage chimique sur votre carte

De l'or !

Sans supplément !

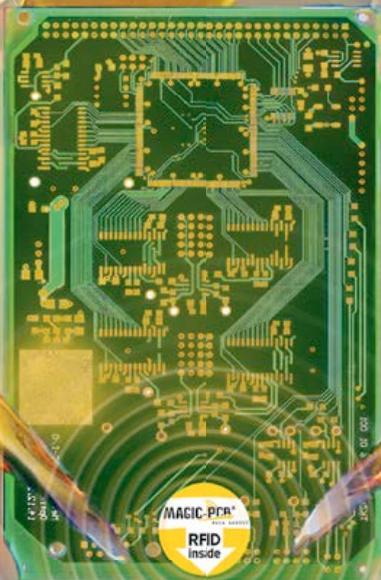
La meilleure finition : nickel-or chimique

Multicouche

Jusqu'à 50%

de réduction de prix

pour les cartes
multicouche 2-5



www.pcb-pool.com

Beta

LAYOUT

create : electronics

interface universelle de mesure précise la précision, tout simplement

Michel Defrance
(Chartres)

La plupart des microcontrôleurs possèdent un convertisseur numérique analogique intégré, mais que faire lorsque sa précision n'est pas satisfaisante ? Ne cherchez plus : la solution est sous vos yeux.

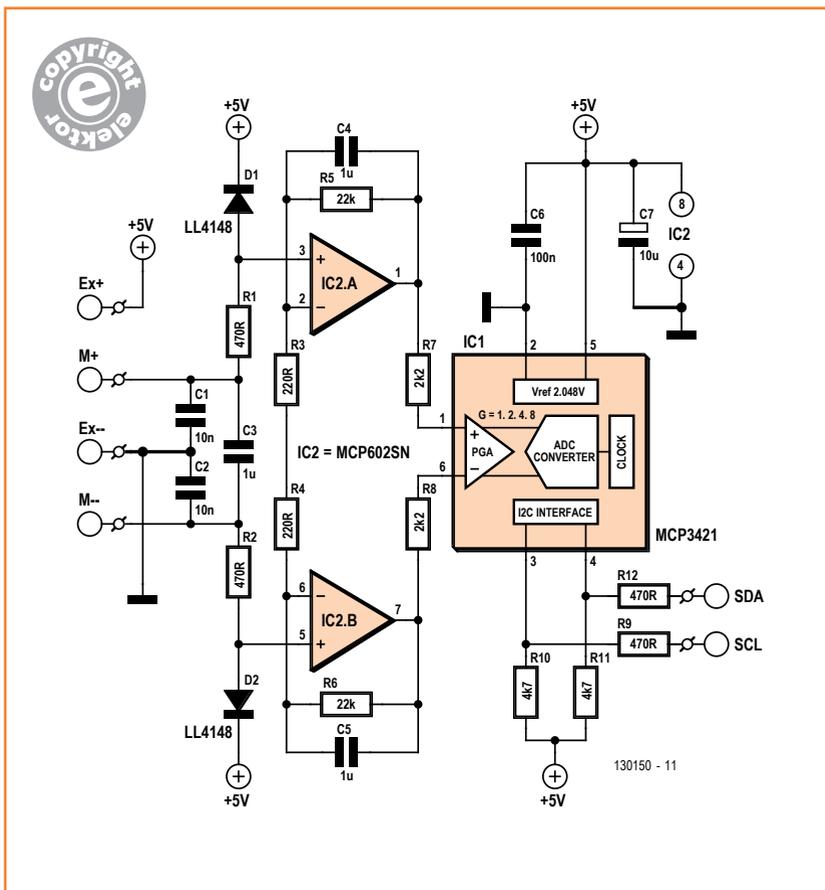


Figure 1.
Schéma de l'interface de mesure.

Tableau 1. Plage de la tension d'entrée en fonction du gain du MCP3421.

Gain du MCP3421	Étendue de la tension d'entrée
1	-20 mV à +20 mV
2	-10 mV à +10 mV
4	-5 mV à +5 mV
8	-2,5 mV à +2,5 mV

Très souvent les électroniciens ont besoin de mesurer de faibles tensions avec une grande précision, p. ex. la tension issue d'un capteur de pression, de température ou encore la tension de sortie d'un pont de Wheatstone (souvent de l'ordre du millivolt). Les μC sont pratiques et assez faciles à programmer, mais tout le monde sait qu'ils ne brillent pas par la précision de leurs convertisseurs (souvent à 8/10/12 bits). En plus, pour mesurer des tensions faibles, celles-ci doivent être conditionnées, souvent à l'aide d'amplis-op.

Confronté une fois de plus à cette problématique, j'ai décidé de trouver une solution à la fois satisfaisante et réutilisable. L'inspiration étant là, il ne restait qu'à passer par la case transpiration. Le cahier des charges est clair ; il faut une interface facile à relier à n'importe quel μC , facile à assembler, avec des composants courants et bon marché. Dans le catalogue de *Microchip*, deux circuits ont retenu mon attention :

- l'ampli à hautes performances MCP602
- le CAN à 18 bits programmable MCP3421 avec interface I²C, référence de 2,048 V intégrée et amplificateur programmable

Amplificateur d'instrumentation

Lorsque l'on souhaite conditionner de faibles tensions pour une numérisation, on utilise souvent un montage appelé amplificateur d'instrumentation, réalisé avec trois amplis-op. Ici, j'utilise les deux amplis-op MCP602 IC2A et IC2B en configuration différentielle pour attaquer l'amplificateur à gain programmable contenu dans le MCP3421. La tension en sortie de ce premier étage sera proportionnelle à la différence de tension entre les entrées de mesure M+ et M- du schéma (fig. 1).

Le gain du premier étage est donné par :

$$G1_+ = 1 + R5/R3$$

$$G1_- = 1 + R6/R4$$

Comme nous souhaitons que l'amplification du potentiel à l'entrée M+ soit la même que celle de l'entrée M-, nous choisirons $R5 = R6$ et $R3 = R4$. Avec les valeurs du schéma, nous aurons : $G1 = 1 + 100 = 101$

Veillez à choisir des résistances avec une tolérance de 1 % pour R3 à R6, faute de quoi vous risquez une forte dissymétrie dans l'étage d'entrée. Passons au deuxième étage construit autour de l'amplificateur/CAN IC1. Utilisé en mode symétrique, il accepte des tensions de $-2,048$ V à $+2,048$ V entre ses broches 1 et 6. Comme le gain G2 de l'ampli qu'il contient est programmable par logiciel, il sera donc possible de choisir différentes plages pour la tension d'entrée du premier étage. Le gain total du montage $G = G1 \times G2$ variera donc entre 101 pour $G2 = 1$ et 808 pour $G2 = 8$. Le **tableau 1** donne une liste de valeurs possibles. Si ces plages ne vous conviennent pas, modifiez les valeurs de R3 à R6.

Alimentation

L'interface est alimentée sous 5 V. En présence de signaux faibles, le bruit de l'alimentation peut devenir gênant et ce montage n'échappe pas à la règle. Peu importe l'alimentation que vous choisirez, elle devra être précise et faire le moins de bruit possible. En effet, avec une précision de conversion de 18 bits, le moindre bruit d'alimentation viendra parasiter vos mesures. Je recommande également l'utilisation d'une alimentation à démarrage progressif afin de réduire la dérive due aux variations des caractéristiques des composants. Il serait possible d'utiliser une temporisation logicielle à cet effet, mais cela ne bénéficierait pas à l'ensemble des composants de la carte. Je propose sur [1] une telle alimentation autour d'un régulateur à faible chute MIC2941 de *Micrel*.

Fabrication et utilisation

L'assemblage du circuit imprimé (**fig. 2**) de 35 x 25 mm ne devrait pas poser de problème aux lecteurs familiers des CMS. Si vous dessinez votre propre circuit imprimé, pour obtenir des performances optimales respectez les conseils donnés dans la feuille de caractéristiques du MCP3421.

Caractéristiques

- Conversion 18 bits
- Interface I²C
- Gain programmable par logiciel

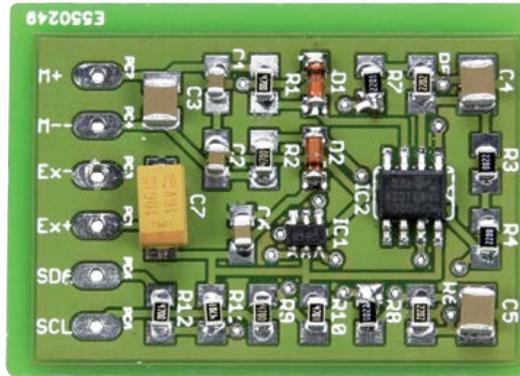


Figure 2. Vue du circuit imprimé de l'auteur.

Celle-ci est précieuse aussi pour l'utilisation du montage. Attention à l'adresse I²C du MCP3421 : celle-ci dépend de la référence exacte du composant que vous obtiendrez. Ceci est aussi précisé dans la feuille de caractéristiques.

J'ai réalisé, à titre d'exemple, une application articulée autour d'un μ C PIC18F452 (ou un PIC16F876A) qui affiche sur LCD la tension lue depuis le MCP3421 via le bus I²C. Vous la trouverez ailleurs dans ce numéro. Elle mesure la faible tension (quelques millivolts) issue d'une jauge de contrainte placée dans un pont de Wheatstone. Le programme du PIC, en PICBASIC, se charge de l'acquisition de la tension depuis le MCP3421, convertit en tension la valeur lue puis affiche sur un LCD la valeur brute issue de l'ADC et valeur signée de la tension correspondante. Ce programme ne devrait pas être trop dur à porter p. ex. sur un Arduino.

Ce circuit peu onéreux (moins de 15 €) pourra être réutilisé dans de nombreux montages différents. Vous n'aurez plus maintenant de raison de pester contre le CAN de votre microcontrôleur favori.

(130150)

Liens

[1] www.elektor-labs.com/node/3053

Liste des composants

Résistances :

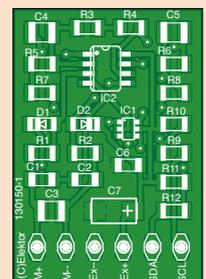
- R1, R2, R9, R12 = 470 Ω 5 %
 R3, R4 = 220 Ω 5 %
 R5, R6 = 22 k Ω 1 %
 R7, R8 = 2,2 k Ω 5 %
 R10, R11 = 4,7k Ω 5 %

Condensateurs :

- C1, C2 = 10 nF 50 V 10 %
 C3, C4, C5 = 1 μ F 50 V 10 %
 C6 = 100 nF 50 V 10 %
 C7 = 10 μ F 16 V 10 % électrolytique

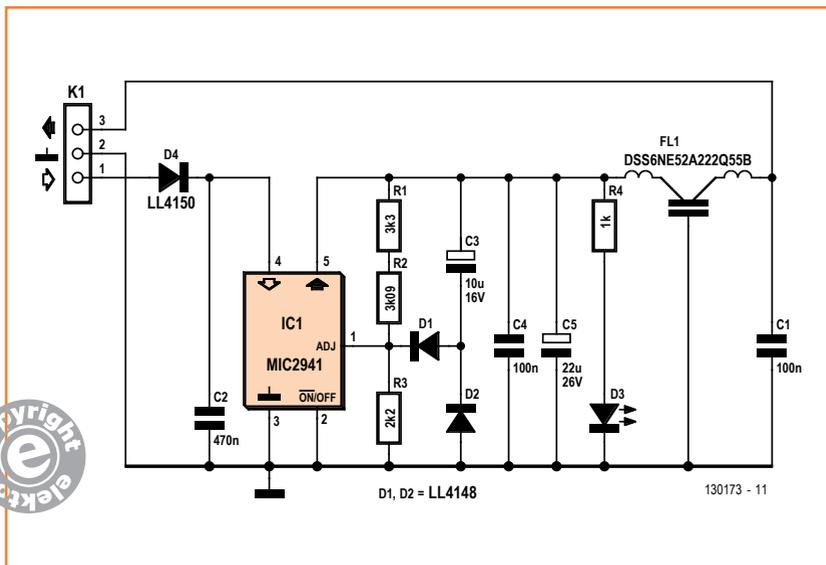
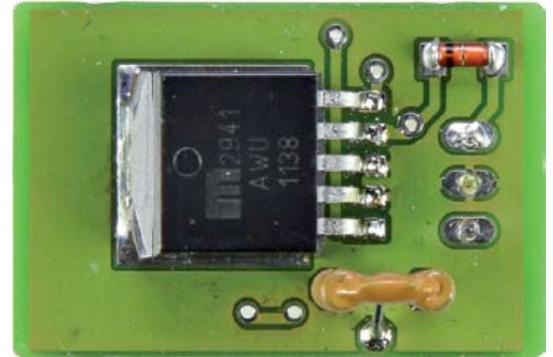
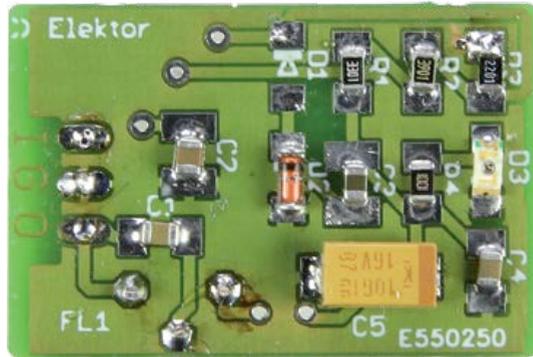
Semiconducteurs :

- D1, D2 = LL4148
 IC1 = MCP3421A1T
 IC2 = MCP602SN



alimentation temporisée pour circuits de précision

Michel Defrance
(Chartres)



Pour alimenter des circuits de mesure de précision et des convertisseurs analogique-numérique, on ne peut pas se contenter d'un régulateur 7805, trop bruyant et imprévisible au démarrage. Prenons l'exemple de l'interface universelle de mesure précise [2] publiée ailleurs dans ce numéro. Elle fait appel à un CAN MCP3421 ; pour pouvoir tirer pleinement profit des 18 bits de résolution de ce convertisseur, il faut l'alimenter à partir d'une source non seulement parfaitement stable et sans bruit, mais avec une tension à montée lente. Ceci permet une stabilisation optimale du circuit de mesure, aussi bien en tension qu'en température. Ce retard à l'allumage pourrait bien sûr être obtenu avec une temporisation logicielle, laquelle ne pourrait toutefois bénéficier qu'à une partie des composants, pas à tous.

Le circuit proposé ici remplit tous les critères évoqués. Il peut remplacer avantageusement un 7805 ordinaire avec lequel le circuit imprimé est compatible broche à broche. Cela implique l'utilisation de composants montés en surface afin d'obtenir le plus faible encombrement possible. La tension de sortie du régulateur à faible tension de déchet MIC2941 (*Micrel*) est fixée à l'aide d'un diviseur de tension comme sur un LM317. Pour le reste, le circuit est aussi simple qu'efficace. La tension d'alimentation est dosée par $(R1+R2)/R3$. Les valeurs indiquées donnent 5 V. La diode D4 protège le circuit contre une malencontreuse inversion de polarité. Le filtre EMI à la sortie, de type DSS6NE52A222Q55B, est un composant à trois broches de *Murata*, faite de deux selfs avec,

Liste de composants

Résistances (CMS 1206) :

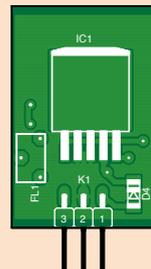
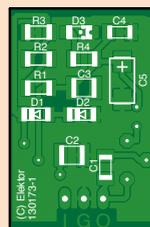
- R1 = 3k3
- R2 = 3k09
- R3 = 2k2
- R4 = 1k Ω

Condensateurs :

- C1, C4 = 100 nF (CMS 1206)
- C2 = 470 nF (CMS 1210)
- C3 = 10 μ F/16 V (CMS 1210)
- C5 = 22 μ F/10 V (CMS 2312)

Semi-conducteurs :

- D1, D2 = LL4148
- D3 = LED à faible courant (CMS 1206)
- D4 = LL4150



IC1 = MIC2941AWU TR (TO-263)

Divers :

- Filtre EMI DSS6NE52A222Q55B (*Murata*)
- circuit imprimé 130173-1 [1]

entre les deux, un condensateur à la masse. La montée lente de la tension est obtenue à l'aide de C3. Lors de la mise sous tension, ce condensateur maintient le point commun à R2 et R3 à un potentiel quasi nul, avant de se charger en 20 ms environ à travers R3. Pendant ce temps, la tension augmente progressivement. À l'extinction, les diodes D1 et D2 étouffent l'apparition éventuelle d'une tension négative à l'entrée de régulation.

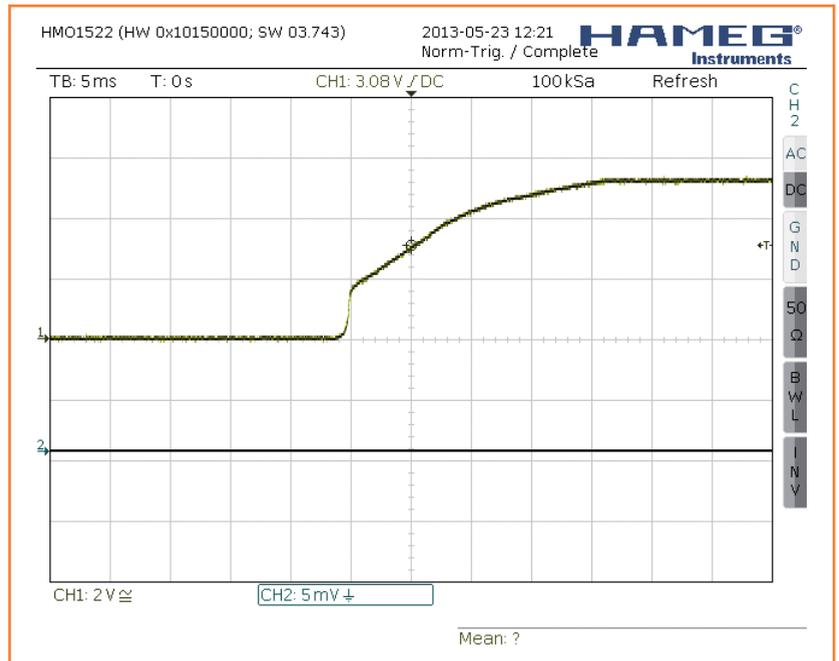
Bien refroidi, ce circuit est capable de débiter au moins 1 A, sinon il ne faut pas en escompter plus de quelques dizaines de milliampères pour une tension d'entrée de 12 V.

Un dessin de circuit imprimé est disponible [1].
(130173)

Liens

[1] www.elektor.fr/130173

[2] www.elektor.fr/130150



Publicité



ISBN 978-2-86661-186-6 - 352 pages - 42,50 €

Kit n°1 : sirène - réf. 119016-71 - 22,50 €

Kit n°2 : chenillard & thermomètre - réf. : 119016-72 - 22,50 €

**nouvelle édition revue et augmentée du livre
l'électronique pour les débutants**

deux kits d'initiation disponibles

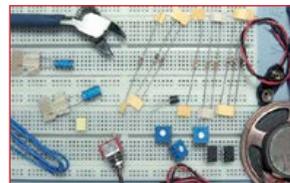
Voici le cadeau idéal pour partager votre passion de l'électronique avec vos enfants, petits-enfants, neveux... et autres geeks

Fin pédagogue, Rémy Mallard écrit pour les débutants dans un style inédit, et répond d'abord aux questions prosaïques du néophyte : quel fer à souder acheter ? Un multimètre à 5 € peut-il suffire ? Et bien d'autres interrogations trop souvent laissées en suspens.

L'auteur démystifie l'électronique en n'utilisant que ce qu'il vous faut de théorie pour aborder la pratique : identifier les composants et leur rôle, les récupérer, les tester et les ranger ; lire un schéma ; choisir ses outils ; mettre en boîte ses montages...

Les deux kits disponibles séparément permettent de réaliser, sur une plaque d'expérimentation sans soudure, quelques-uns des montages simples et ludiques présentés dans le livre.

nouveau kit !



Offre spéciale : livre + deux kits = 81,50 € au lieu de 87,50 €

**Informations complémentaires et commande :
www.elektor.fr/debut**

testeur de servomoteurs simple et indispensable

Bernhard Kaiser & Michael Gaus
(Allemagne)

À vue d'œil, rien ne distingue un servomoteur en panne d'un servomoteur en bon état de marche. Modélistes, ce petit testeur fera donc votre bonheur !

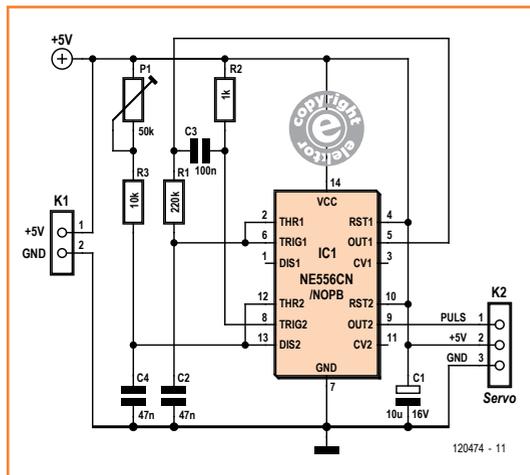
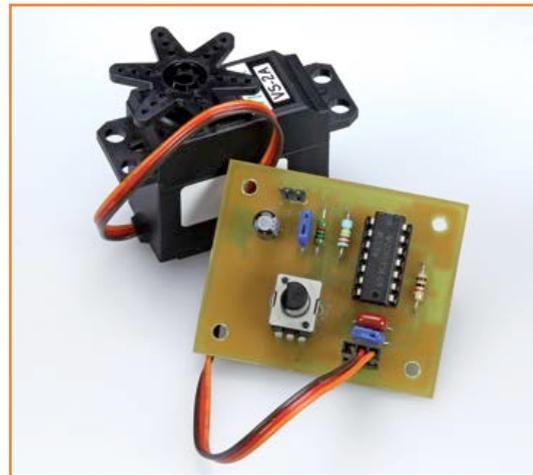


Figure 1.
Un double temporisateur/ compteur au cœur d'un circuit simple.



En modélisme, les servomoteurs sont appréciés pour leur faible encombrement, leur légèreté, leur bas prix, et aussi parce qu'ils sont faciles à commander. Les servomoteurs pour modèles réduits sont reliés directement à un récepteur radio. Ils comprennent habituellement trois fils : une alimentation (+5 V), une masse (GND), et un signal de commande modulé en largeur d'impulsion fourni par le récepteur radio. C'est ce signal qui détermine la position du bras : une impulsion positive de 1 ms de largeur l'emmène sur la butée de gauche, une largeur de 2 ms le positionne contre la butée de droite et, fort logiquement, le bras est centré avec une durée d'impulsion de 1,5 ms. La durée de l'impulsion détermine donc

l'angle que prend le bras du servomoteur. Les impulsions sont émises toutes les 20 ms environ, ce qui équivaut à un signal de 50 Hz. La précision de cette fréquence est toutefois secondaire.

La panne ou le dysfonctionnement d'un modèle peuvent avoir pour origine un défaut de la télécommande ou du récepteur radio, mais également du moteur. Le circuit présenté ici offre un moyen simple et rapide de vérifier le bon fonctionnement d'un servomoteur et de le mettre ou non en cause. Le générateur d'impulsions (**fig. 1**) qui est au cœur du testeur fait partie des ingrédients de base de la cuisine électronique.



Figure 2.
Ce circuit imprimé facilitera votre montage.

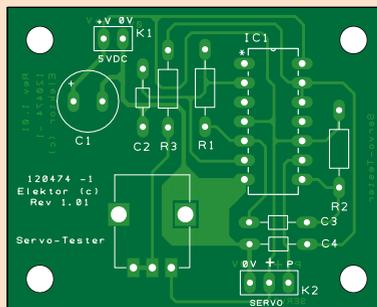
Liste des composants

Résistances :

R1 = 220 kΩ
R2 = 1 kΩ
R3 = 10 kΩ
P1 = pot. lin. 50 kΩ

Condensateurs :

C1 = 10 µF, 16 V, au pas de 7,5 mm



C2, C4 = 47 nF
C3 = 100 nF

Semi-conducteur :
IC1 = NE556CN

Divers :

K1 = embase à 2 contacts au pas de 2,54 mm
K2 = embase à 3 contacts au pas de 2,54 mm
circuit imprimé 120474-1

Temporisateur/compteur double

Le générateur d'impulsions est construit avec un NE556 (version double du NE555). Le temporisateur Timer1 de ce NE556 fixe la fréquence du signal de commande par l'intermédiaire de la résistance R1 et du condensateur C2. Le signal de sortie sur la broche 5 possède un rapport cyclique à peu près symétrique. Le front négatif déclenche le second temporisateur via le condensateur C3. Ce *timer* produit alors une unique impulsion positive sur la broche 9 de sortie. La largeur d'impulsion dépend des valeurs de C4 et de R3, ainsi que du potentiomètre P1, qui permet donc de la modifier.

Lors des essais, les valeurs des composants du circuit ont permis d'obtenir pour la largeur d'impulsion des durées comprises entre 0,5 et 2,6 ms, soit de quoi couvrir l'habituel intervalle de 1,0 à 2,0 ms. Ne réglez toutefois pas le potentiomètre au minimum ou au maximum, vous risqueriez d'envoyer l'axe du servomoteur contre la butée mécanique et de réduire sa durée de vie ; pla-

cez donc le curseur en milieu de course avant de mettre le circuit sous tension. Lors des essais, nous avons mesuré un écart de 18 ms entre chaque impulsion.

Les servomoteurs pour modèles réduits fonctionnent en général avec une tension comprise entre 4,8 et 6 V. La tension d'alimentation est ici comprise entre 5 et 6 V, et est fournie par quatre piles/accumulateurs AA. Vous pouvez également utiliser quatre accus NiMH de 1,2 V.

Pour faciliter le montage du circuit, nous avons conçu une petite carte (**fig. 2**) que vous pouvez vous procurer auprès d'Elektor [1]. L'implantation des composants ne pose aucune difficulté. Les fichiers de projet DesignSpark peuvent être téléchargés depuis [1].

(120474 – version française : Hervé Moreau)

Lien

[1] www.elektor.fr/120474

Publicité

OBTÉNEZ LA DERNIÈRE VERSION !

EAGLE

V6

Nouvelles fonctionnalités de la version 6.4:

- La simulation de schémas EAGLE dans LTspice IV
- L'affichage et la fonction de recherche des attributs pour le dialogue ADD- et REPLACE-
- L'importation de fichiers de design depuis P-CAD, Altium et Protel dans ACCEL-ASCII
- Un guide utilisateur et le pré-réglage (astuces sur les outils, raccourcis) optimisés



www.cadsoft.de



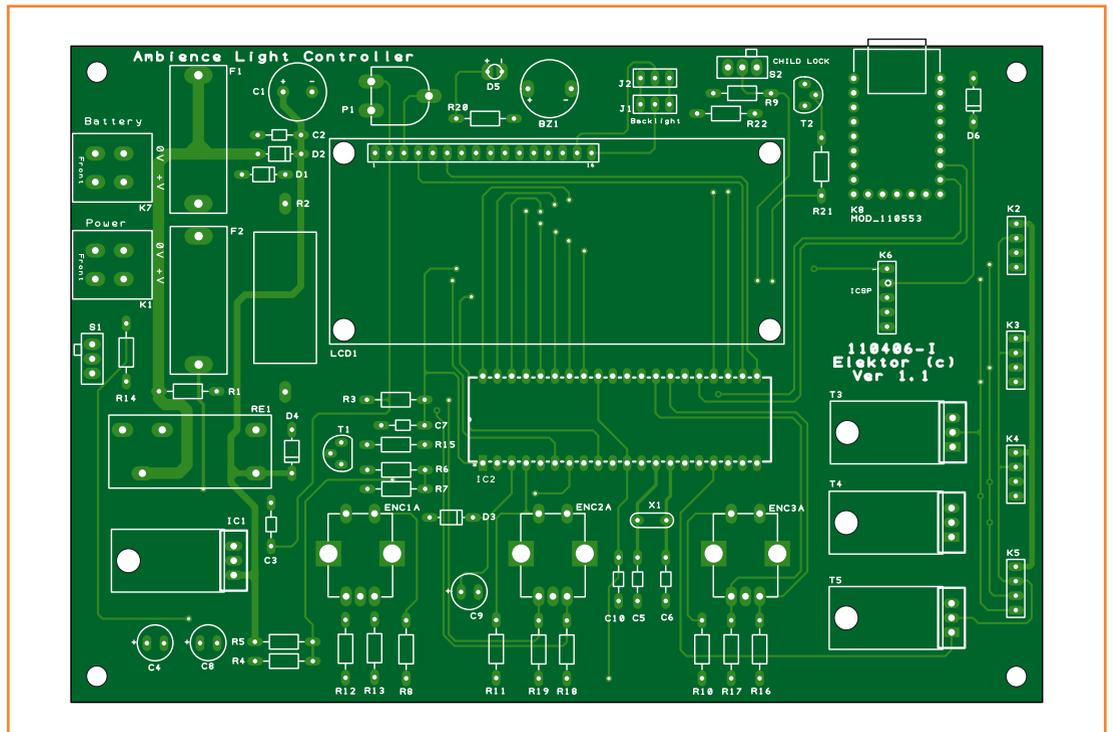


Figure 2.
Le circuit imprimé porte aussi tous les composants de service et le LCD.

K2 à K5. Sans refroidissement supplémentaire, chaque FET peut fournir quelques ampères. Le vibreur acoustique Bz1 possède son oscillateur interne. Dès qu'il est alimenté, il fait du bruit.

Ce sont trois codeurs rotatifs à bouton-poussoir incorporé qui permettent l'introduction des instructions de fonctionnement de l'appareil. Sur le μC , RA0 et RA1 servent d'entrées au conver-

Liste des composants

Résistances :

R1 = 47 Ω /0,5 W
R2 = 1 Ω /5 W
R3, R5, R6, R8 à R14, R16 à R19 = 10 k Ω
R4, R7 = 3k9
R15, R20, R21 = 1 k Ω
R22 = 47 Ω
P1 = 10 k Ω aj. horiz.

Condensateurs :

C1 = 470 μF /16 V radial
C2, C3, C10 = 100 nF
C4 = 100 μF /16 V radial
C5, C6 = 15 pF
C7 = 470 nF
C8, C9 = 1 μF /16 V radial

Semi-conducteurs :

D1, D2 = 1N5400
D3, D4, D6 = 1N4148
T1, T2 = BC547B
T3, T4, T5 = IRL540 (International Rectifier, Farnell 8651078)
IC1 = LM7805
IC2 = PIC16F887 (programmé : EPS 110406-41)
D5 = LED rouge, 3 mm

Divers :

X1 = quartz 20 MHz
F1 = fusible 2 AT avec support encartable
F2 = fusible 3,15 AT avec support encartable
BZ1 = vibreur CC (oscillateur incorporé)
RE1 = relais 12 V contact aiguille, courant commuté min. 2 A (p.ex. Finder 40.31.7.012.0000, Farnell 1169158)
MOD1 = passerelle USB-FT232R BOB Elektor 110553-91 [1]
S1, S2 = commutateur à glissière, commande latérale, encartable (p.ex. C&K OS102011MA1QN1, Farnell 1201431)
S3, S4, S5 = encodeur rotatif à poussoir incorporé (p.ex. Alps EC12E2424407, Farnell 1520813)
K1, K7 = embase à 2 picots, au pas de 5 mm
K2, K3, K4, K5 = embase à 4 picots au pas de 2,54 mm
K6 = embase à 5 picots, au pas de 2,54 mm
LCD1 = LCD 2x16 caractères, avec éclairage d'arrière plan (p.ex. Elektor 120061-71))
J1, J2 = embase à 3 picots avec cavalier, au pas de 2,54 mm
support DIL à 40 contacts pour IC2
circuit imprimé 110406-1 [2]

tisseur A/N (CAN). RA0 mesure la tension de batterie et RA1 détecte le branchement d'un chargeur et vérifie si sa tension est au moins de l'ordre de 14 V. Les diviseurs R5/R4 et R6/R7 ramènent les tensions dans la gamme du contrôleur.

Le schéma montre également un module USB-FT232R BOB d'Elektor [1] pour établir une liaison avec le PC. On peut ainsi utiliser un programme de terminal pour envoyer des commandes à distance au circuit. La notice d'utilisation est gratuite [2]. Un classique 7805, IC1, stabilise la tension d'alimentation avec quelques condensateurs et la diode D2 protège de l'inversion de polarité. La valeur du fusible F1 qui précède le régulateur dépend de la charge. Pour 2 m de ruban à LED, un fusible de 2 A suffit, mais le mieux est de le dimensionner en mesurant la consommation quand toutes les LED brillent à pleine intensité. L'accumulateur gélifié se branche sur K7, tandis que l'adaptateur secteur vient sur K1, il doit délivrer environ 15 V et au moins 2 A. Le transistor T1 commande RE1 qui court-circuite R1 quand on recharge l'accu. Le type de relais n'est pas critique, il faut qu'il commute sous 12 V et que les contacts supportent plus de 2 A. Davantage d'information dans l'encadré *Mise en service*.

Le circuit lui-même ne consomme que 25 mA en service, mais avec l'éclairage de l'écran, il prend environ 50 mA.

Le circuit imprimé

La **figure 2** reproduit le circuit imprimé développé par le labo d'Elektor. On n'y voit que des composants à fils, c'est plus facile. Tous les composants sont sur la même face. Les broches du régulateur et des MOSFET seront pliées à la pince à becs plats de manière à ce que ces pièces soient couchées à même la platine. Pour un usage normal, ces composants ne requièrent aucun radiateur. Une fois programmé, il est recommandé d'insérer le μC dans un support. Si vous désirez utiliser une liaison série avec le PC, il faut aussi placer une passerelle USB-FT232R BOB. Vous pouvez la monter directement sur le circuit imprimé ou sur deux rangées de 9 broches SIL.

Le logiciel

Le code pour ce circuit a été écrit en ANSI C à l'aide de MPLAB et compilé avec une version complète de Hi-Tech C en mode d'évaluation (valable 45 jours). La version *lite* du compilateur ne suffit pas dans ce cas-ci parce qu'elle ne dispose pas d'une place suffisante pour l'optimisation.

L'exécutable est alors trop volumineux pour la mémoire de 8 Ko du μC . On peut télécharger les codes source et hexadécimal ainsi que le tracé des pistes du site d'Elektor [2] ou y commander une version programmée du contrôleur.

La partie la plus importante du logiciel est l'ISR : *Interrupt Service Routine*. Avec la fonction *stopwatch* de MPLAB, cette routine est optimisée pour une exécution aussi rapide que possible. L'ISR est divisée en parties qui peuvent s'exécuter en 100 μs , 5 ms, 100 ms ou 1 s. L'ISR est appelée toutes les 100 μs , après quoi l'utilisation de compteurs veille à ce que les autres intervalles cités puissent exécuter quelque chose.

Pour épargner de la mémoire, j'ai de prime abord utilisé un compteur pour déterminer les instants précis pour 1 ms et 100 ms. Lors de chaque appel d'interruption, toutes les 100 μs , un module de calcul opérait une division pour évaluer le reste. Quand ce résultat était nul, 1 ms s'était écoulée. Pendant le débogage avec *stopwatch*, il est apparu que ce module de calcul exigeait tellement de temps qu'il valait mieux utiliser un autre compteur pour déterminer les intervalles de 1 ms. Voici comment fonctionne le réglage automatique de l'intensité lumineuse. La fréquence normale de la MLI est de 100 Hz à 11 V. Si la tension d'accu dépasse 11 V, les LED devraient briller davantage pour un même rapport cyclique. L'adaptation de ce rapport à la tension de batterie signifie qu'il faut calculer pour chaque couleur un nouveau rapport cyclique. Une solution beaucoup plus simple consiste à maintenir constant le temps d'allumage, mais de réduire la fréquence de la MLI quand la tension d'accumulateur est plus haute. Il ne faut plus qu'un seul calcul pour obtenir le même résultat. On sait que la différence entre une fréquence de MLI de 100 Hz et une de 90 Hz n'est pas visible.

Un autre morceau de logiciel à donner la migraine au programmeur, c'est celui qui préside au mélange des couleurs fondamentales.

Pourtant, le calcul en lui-même est très simple : prendre la valeur MLI actuelle, calculer la différence avec la valeur MLI de l'étape suivante et diviser le résultat par la durée de la transition. Augmenter ou diminuer ensuite chaque valeur MLI à chaque étape pendant la transition. Dans ce cas-ci, le temps de transition est exprimé en pas de 100 ms.

Le résultat de cette division peut être un nombre avec beaucoup de chiffres derrière la virgule, il faut donc utiliser une variable à virgule flottante

Listing 1

```
fade_step_red = current_red_value - next_red_value;
fade_step_green = current_green_value - next_green_value;
fade_step_blue = current_blue_value - next_blue_value;
fade_step_red = fade_step_red * 100;
fade_step_green = fade_step_green * 100;
fade_step_blue = fade_step_blue * 100;
fade_step_red = fade_step_red / fade_time;
fade_step_green = fade_step_green / fade_time;
fade_step_blue = fade_step_blue / fade_time;
```

Listing 2

```
tmp_red_value = fade_tmr * fade_step_red;
tmp_green_value = fade_tmr * fade_step_green;
tmp_blue_value = fade_tmr * fade_step_blue;
tmp_red_value = tmp_red_value / 100;
tmp_green_value = tmp_green_value / 100;
tmp_blue_value = tmp_blue_value / 100;
red_value = next_red_value + tmp_red_value;
green_value = next_green_value + tmp_green_value;
blue_value = next_blue_value + tmp_blue_value;
```

(*float*) pour calculer et enregistrer le nombre. Les PIC16F sont à 8 bits et le compilateur peine avec les « grandes » variables *float*. D'où il découle des soucis de chronologie et des erreurs dans le code compilé. La solution trouvée est relativement simple : multiplications et divisions d'entiers vont plus vite et prennent bien moins de place. Aussi, multiplions la différence de valeur MLI de chaque couleur par 100 avant de diviser par le temps de transition, voyez donc ce qu'il advient du code (listage 1). On calcule la valeur MLI actuelle à chaque pas de transition, toutes les 100 ms. Finalement, on divise par 100. Après arrondi sur 2 décimales, le résultat est un entier, donc sans chiffres derrière la virgule (listage 2). Voilà comment on arrive à mélanger entre elles les couleurs sans trop de difficultés en termes de performance avec un µC à 8 bits.

Mise en service

Dès sa mise sous tension, le contrôleur se manifeste sur l'écran LCD par son nom et son numéro de version ; une seconde plus tard, il présente le menu. Le maniement du circuit s'explique de lui-même, son mode d'emploi en anglais est gratuit [2].

Le codeur rotatif ENC1 permet de naviguer dans le menu déroulant. D'abord les 3 menus Run Program <x> (x va de 1 à 3) pour lancer les programmes enregistrés. Sélectionnez l'un d'eux et appuyez sur ENTER, c'est le bouton-poussoir de l'ENC2. On retourne au menu par BACK, c'est le poussoir de l'ENC3.

L'option du menu *Charge Battery* assure une charge surveillée de l'accu. Branchez un chargeur de 15 V et choisissez cette option. Le relais s'enclenche, la résistance R1 est court-circuitée et l'accu reçoit un courant de charge suffisant. Les rubans à LED sont mis hors circuit parce que la tension d'entrée est trop élevée pour eux, elle risque de les endommager. Quand la batterie atteint 13,8 V, le relais chute et il ne circule plus qu'un faible courant pour une charge au goutte-à-goutte, mais les rubans de LED colorées s'éclairent de nouveau.

Battery Charge indique la charge restante en paliers de 10 %. La valeur résulte de la mesure de tension de l'accu : 0 % pour 11 V, 100 % pour 13,8 V.

La LED D5 témoigne de l'état de charge de l'accumulateur. S'il est plein (13,2 V), la LED reste allumée. Quand il commence à se vider, elle clignote à la fréquence de 1 Hz. Le rapport cyclique varie en fonction de la charge restante. Quand il est presque vide, la LED clignote brièvement. Quand il est quasiment à plat, les rubans de LED sont débranchés et le vibreur émet des bips. L'interrupteur S2 fonctionne à la manière d'un code parental (*Child Lock*). S'il est fermé, les options du menu Edit Program <x> et *Edit <color>* ne sont pas accessibles.

Pour régler l'une des couleurs, sélectionnez *Edit <color>*, appuyez sur ENTER et réglez les couleurs rouge, verte et bleue entre 0 et 100 % par pas de 1 % au moyen des trois codeurs. Poussez sur ENTER pour enregistrer le nouveau réglage ou sur BACK pour revenir au menu sans enregistrer. Pour rédiger un programme, on sélectionne *Edit Program <x>* et on appuie sur ENTER. On choisit alors la couleur avec l'encodeur rotatif ENC1, le temps d'allumage *Hold* avec ENC2 et le temps de transition *Fade* avec ENC3.

(110406 – version française : Robert Grignard)

Liens

[1] www.elektor.fr/110553

[2] www.elektor.fr/110406



DESIGNSPARK PCB

NOUVEAU

L'ÉVOLUTION SE POURSUIT

DESIGNSPARK PCB v5

Découvrez les nouvelles fonctionnalités
www.designspark.com/pcb

RESSOURCES
UNIQUES PAR



régulateur de pompe pour voilier ou caravane etc.



Paul Cordonnier
(Belgique)

Le mot *pompe*, avec son radical latin *pupp-* qui signifie téter ou sucer, est entré dans la langue française par le vocabulaire de la marine. C'est aussi par la même voie qu'il entre dans ce numéro d'été : je passe en effet pas mal de temps en mer, où j'ai souvent été confronté aux inconvénients des pompes de distribution d'eau sur les voiliers.

En principe, une installation d'eau se compose d'un réseau de distribution mis sous pression par une pompe de 5 à 10 A, alimentée sous 12 V, qui fonctionne généralement en tout ou rien, et d'un ballon d'expansion. Ce dernier supprime les saccades du débit et les coups de bélier. Il est source de confort, mais aussi de soucis : encombrement, maintenance, etc. De nouveaux modèles de pompes permettent de se passer de ballon. C'est un petit circuit électronique qui se charge alors de faire tourner la pompe en fonction du débit. Plus de ballon, la pompe fait moins de bruit, sa consommation chute et le débit se module en douceur. Tout baigne ! Sauf

le prix. Ces nouvelles pompes, avec leur circuit de régulation encapsulé, reviennent pratiquement au double de celles qu'elles remplacent.

Je ne suis pas électronicien de formation, mais de conviction : j'ai donc cherché et trouvé une solution simple et meilleur marché, que je suis heureux de partager ici.

Je commence par attirer l'attention du lecteur sur S1 du schéma de la **figure 1**, dessiné comme un interrupteur ordinaire alors qu'il s'agit d'un pressostat (ou manostat), commandé comme son nom l'indique par la pression d'un fluide quand elle franchit un seuil. C'est donc lui qui met le circuit sous tension quand la pression dans le réseau de distribution d'eau est insuffisante et le coupe quand elle a remonté.

La commande de la pompe (existante) M1 est assurée par IC1 au moyen d'un signal de 1 kHz dont il module la largeur d'impulsion (PWM) sur



Liste des composants

Résistances :

(0,25 W, 6,3 mm)
 R1 = 1 k
 R2 = 10 k
 R3 = 2k5 aj.
 R4, R5 = 470 Ω

Condensateurs :

C1, C2 = 100 nF, 2,5 mm
 C3 = 470 μF, 25 V, radial, 5 mm
 C4 = 33 μF, 10 V, radial, 2 mm
 C5 = 100 nF, 5 mm

Semi-conducteurs :

D1 = 1N5408
 D2 = 1N4001
 D3, D4 = LED 5 mm, rouge
 T1 = RFP70N06, N-MOSFET
 70 A, 60 V
 IC1 = PIC16F684-I/P
 IC2 = 7805

Divers :

K1 = bornier à vis ou à ressort, 3 points, pas de 5,08mm
 K2 = embase mâle 1x5, 2,54 mm

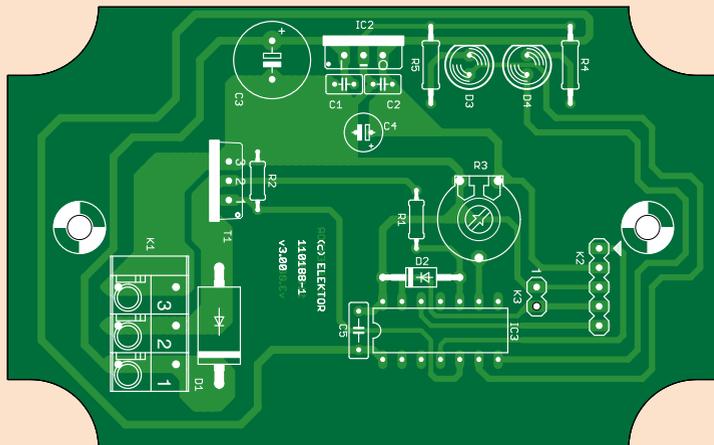


Figure 3. La configuration particulière du circuit imprimé s'explique par le choix d'un boîtier étanche.

K3 = n.c.
 support DIL à 14 broches

modèle de la pompe utilisée :
 Jabsco PAR-MAX 1.9 7,2 L/min 12 V 3 A

Le programme de régulation en PIC Basic Pro est simplissime puisque le PIC n'a rien d'autre à faire. Le code source est téléchargeable sur le site [1]. Le PIC programmé est disponible auprès d'Elektor de même qu'un circuit imprimé.

L'insertion du régulateur dans la ligne d'alimentation négative d'une pompe dans une installation existante ne devrait poser aucun problème, mais son installation demande une attention particu-

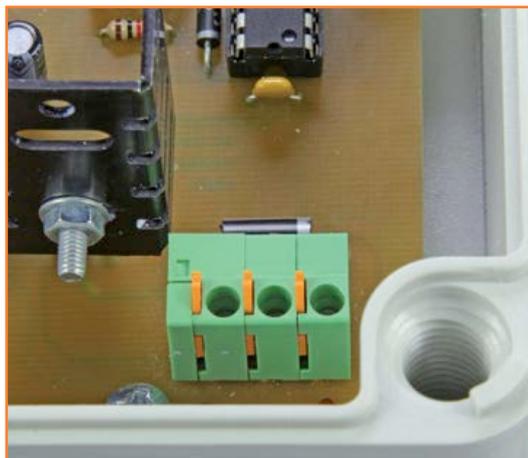
lière, puisqu'il sera placé dans un milieu potentiellement humide. C'est ce qui explique le design particulier du circuit imprimé, étudié pour un boîtier précis (le modèle PICCOLO ABS B 65 T de *Fibox*) choisi pour son indice de protection relatif à l'étanchéité (IP 66/67) ; son contenu est à l'abri de fortes projections d'eau et même jusqu'à 1 m de submersion. La pompe que j'ai utilisée est un modèle 7,2 L/min 12 V 3 A.

Pour K1, les marins préféreront un bornier à ressort plutôt qu'un modèle à vis qui risque de se desserrer en présence de vibrations. Bon vent !

(110188)

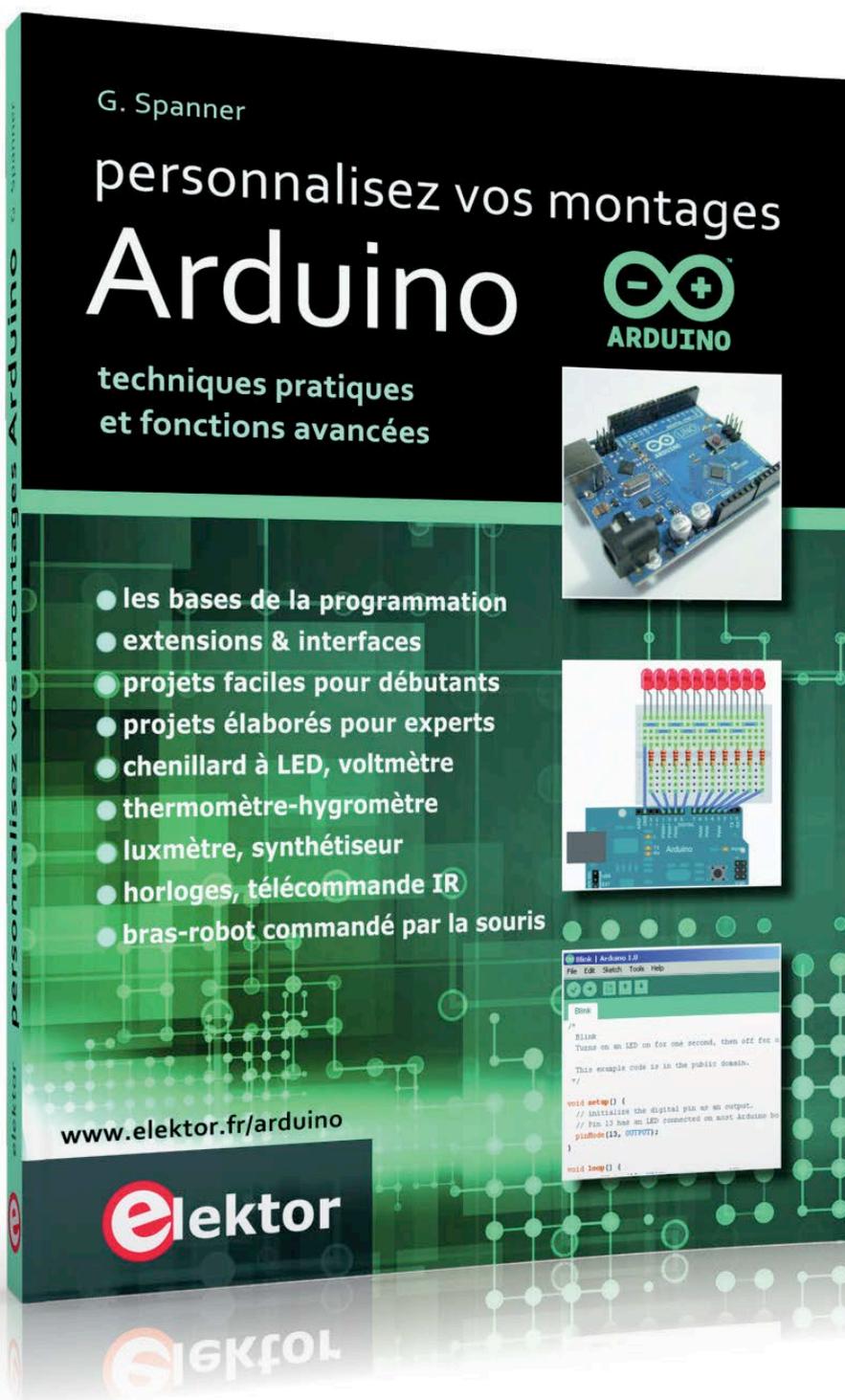
Liens

[1] www.elektor.fr/110188



Personnalisez vos montages Arduino

techniques pratiques et fonctions avancées



NOUVEAU

L'objectif de ce livre est de vous emmener à pas guidés vers la maîtrise d'Arduino.

Les projets, regroupés par thème, accompagnés de bases théoriques, sont des applications concrètes : chenillard à LED, voltmètre, thermomètre numérique, horloges sous différentes formes, ou encore bras de robot commandé par la souris.

Vous apprendrez ainsi à exploiter des techniques essentielles comme la conversion analogique-numérique, la modulation de largeur d'impulsion, ou encore les interruptions.

Après avoir mené à bien tous ces projets vous maîtriserez les fondamentaux de la technique des microcontrôleurs.

ISBN 978-2-86661-191-0

272 pages

34,50 €

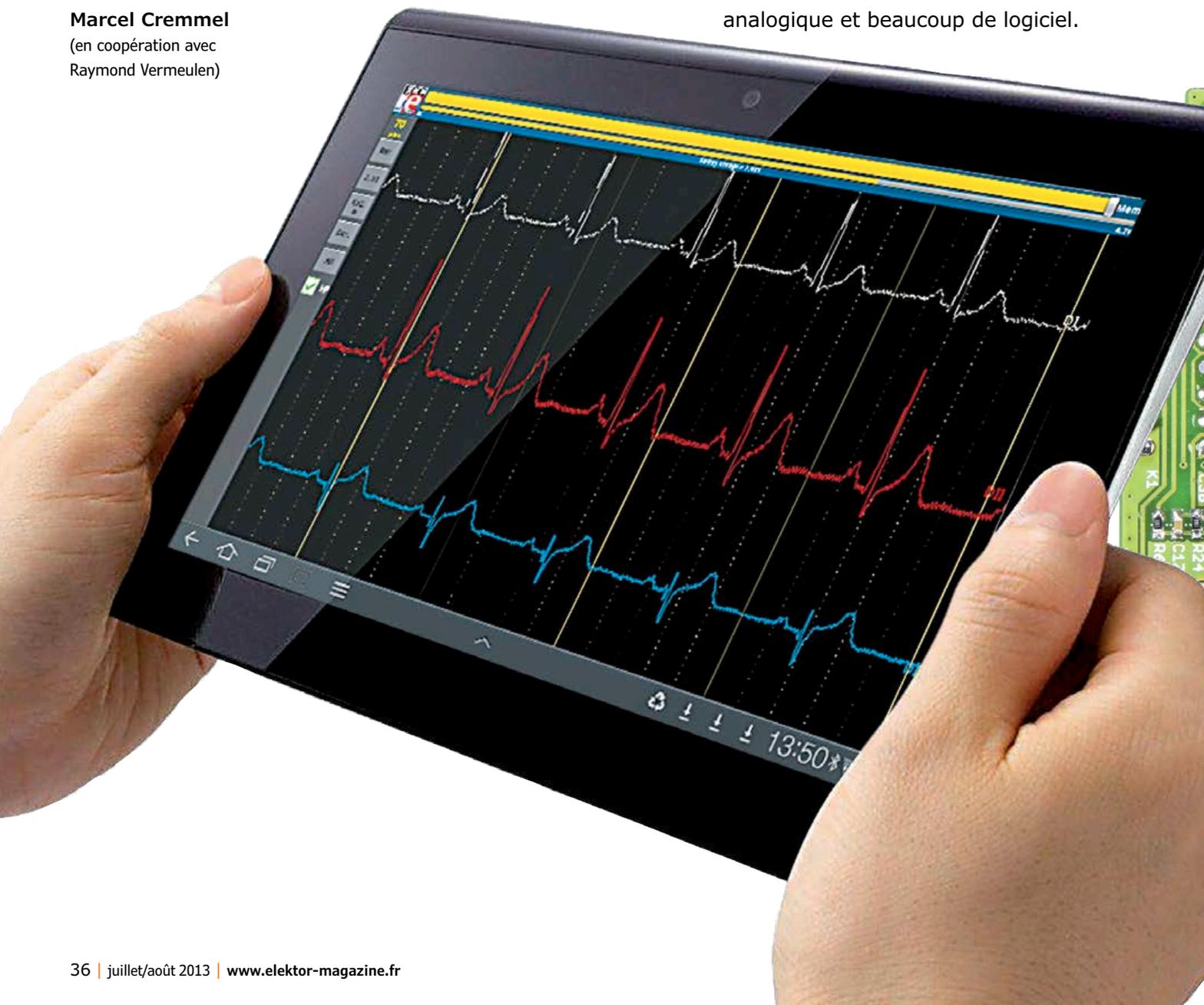
elektor cardi♥scope Android

1^{ère} partie

sans fil, sans bouton : Bluetooth & écran tactile

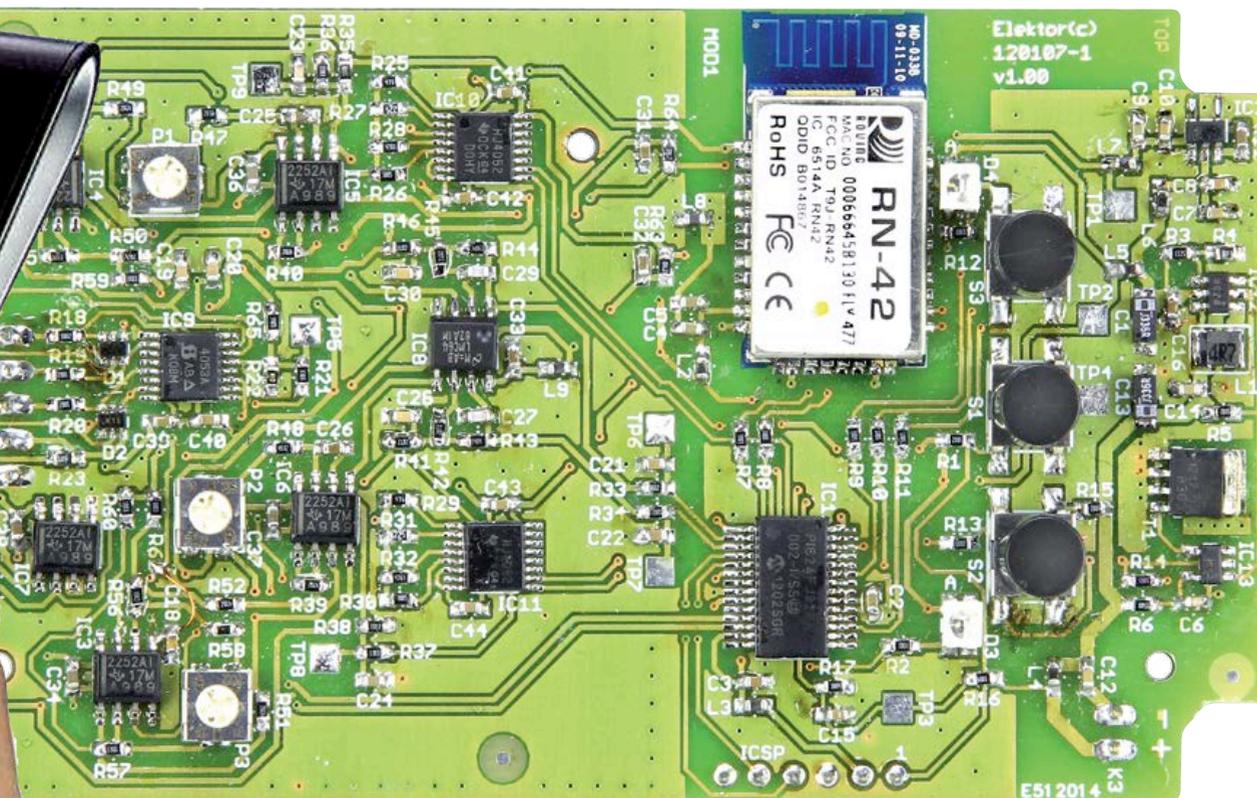
Tout est dans le titre de cet article qui décrit un électrocardioscope à construire soi-même, utilisant une tablette ou un téléphone Android comme terminal sans fil de visualisation des électrocardiogrammes. La réalisation consiste à combiner habilement une petite interface à PIC pour commander un étage d'entrée analogique et beaucoup de logiciel.

Marcel Cremmel
(en coopération avec
Raymond Vermeulen)



Caractéristiques :

- Interface pour téléphones ou tablettes tactiles Android équipés de Bluetooth
- Affichage déroulant simultané ou individuel des trois dérivations standard (DI, DII et DIII) et des trois dérivations augmentées (aVR, aVL et aVF)
- Adaptation automatique à la résolution de l'écran
- Mesure et affichage du rythme cardiaque
- Indication sonore du battement cardiaque
- Vitesses de défilement : 250/125/62,5 et 31,25 pixels par seconde
- Gain à l'affichage : x1 x1,2 x1,5 x2, x3 et x10
- Sensibilité pleine échelle : 3,2 mV. Conversion sur 10 bits.
- Fréquence d'échantillonnage : 2000 Hz
- Réjection du mode commun : > 100 dB
- Tension de contact maximum : +/- 150 mV
- Constante de temps d'alignement auto adaptative
- Bande passante : 0,005 Hz à 170 Hz
- Mémoire de trace dynamique : 10 minutes
- Enregistrement des ECG dans la mémoire flash (10 min)
- Injection périodique d'un signal de calibration 1 mV/2 Hz
- Alimentation par 2 piles 1,5 V ou batteries 1,2 V
- Affichage permanent de la tension de la pile
- Consommation moyenne : 50 mA (en veille : < 4 µA)
- Autonomie : 15 h (piles "High Energy" 1 Ah)
- Faible coût



En 2006, quand Elektor publiait un électrocardioscope à base de console de jeux *GameBoy*, le petit robot Android n'existait pas encore ou alors tout au plus au stade fœtal. Sept ans après, les applications qu'il anime se comptent en millions de millions ; à l'heure

où est publié cet article, plus de 900 millions d'appareils Android sont en circulation. Chaque jour, de nouveaux horizons s'ouvrent pour le petit bonhomme vert. Elektor est heureux de contribuer à cette saga qui ne fait que commencer, avec une application instructive, passionnante,

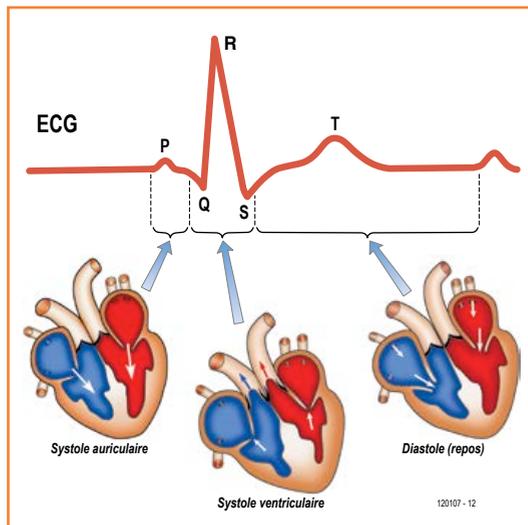


Figure 1.
Correspondance de l'activité électrique prélevée et des phases d'un cycle cardiaque.

et potentiellement utile à tout le monde : faites vous-même vos électrocardiogrammes sur votre tablette ou votre téléphone tactile !

Cet accessoire d'exploration physiologique se compose d'une seule (toute petite) carte de 5,5 x 10 cm, qui réunit les sections analogique et numérique du circuit. Notre interface ECG est disponible sous forme de module prêt à l'emploi, qu'il suffit de compléter avec quatre électrodes et une application Android pour téléphone ou tablette ; entre ce terminal et l'interface, pas de fil, mais une communication *Bluetooth* ! Comme il se doit sur un appareil moderne, celui ne comporte que très peu de points de réglage (3 ajustables en tout et pour tout) et d'organes de commande (3 poussoirs). Autant dire que c'est le logiciel qui fait tout le travail.

Nous décrivons dans au moins deux articles :

- l'électronique : notre domaine de prédilection,
- le programme exécuté par le microcontrôleur PIC24 écrit en C sous MPLAB (IDE et compilateur gratuits chez *Microchip*),
- l'application Android écrite en Java avec le SDK de *Google*,
- la réalisation de l'interface à loger dans un petit boîtier aux dimensions d'un téléphone tactile.

La partie informatique est assez lourde : nous ne détaillerons donc pas le code source, mais, afin de vous encourager à vous l'approprier, le personnaliser ou même l'améliorer, nous décrivons les fonctions logicielles. Pour cela, vous retrouve-

rez dans le code source les identificateurs utilisés dans les schémas. Les projets MPLAB et Android complets sont disponibles sur le site d'Elektor.

Imagerie médicale et électronique

L'électrocardioscope d'Elektor met en œuvre le procédé inventé par Willem Einthoven (cf. *Elektor* oct. 2006, p.32 [1]) qui consiste à explorer la fonction cardiaque au moyen de l'image des phénomènes électriques produits au cours de la révolution cardiaque. Le cœur est un muscle autonome, le seul à ne pas être piloté par le cerveau. Le nœud sinusal, situé dans l'oreillette droite, déclenche les flux nerveux qui commandent les muscles du cœur. Ceux-ci se contractent (dépolariation) et se relaxent (polarisation) pour pomper le sang. La contraction est causée par un changement de polarité électrique des membranes cellulaires. Pendant les phases de relaxation, les charges électriques retrouvent leur équilibre avant une nouvelle excitation. Les potentiels résultants sont transmis jusqu'à la surface de la peau, où ils peuvent être prélevés par des électrodes cutanées.

Le placement approprié des électrodes permet à un cardiologue de déduire le comportement mécanique du cœur (et ses défaillances !) en analysant son activité électrique [2].

La **figure 1** met en correspondance l'activité électrique prélevée et les phases d'un cycle cardiaque :

- *Onde P* : contraction des oreillettes : le sang provenant des veines est poussé dans les ventricules.
- *Complexe QRS* : contraction des ventricules : le sang des ventricules est poussé dans les artères.

Ces deux ondes provoquent le *touduoum* sonore du cœur.

- *Onde T* : repolarisation des ventricules : le muscle des ventricules revient au repos.

La position des quatre électrodes, une sur chaque poignet et cheville est illustrée par la **figure 2**, qui décrit aussi les dérivations affichées par l'appareil. Les dérivations des membres sont placées à la face interne des poignets et des chevilles. Elles explorent le champ cardiaque dans un plan frontal (**fig. 2a** et **2b**). Ces trois dérivations forment le triangle équilatéral d'Einthoven, le montage des polarités étant tel que $D3 = D2 - D1$. La translation des axes DI, DII et DIII au centre

du triangle permet de construire un système de coordonnées (dit double triaxe de Bailey), utile au calcul de l'axe du vecteur d'activation dans le plan frontal (fig. 2c). Les six dériviations représentent l'activité cardiaque selon ces six axes. Leur analyse permet d'établir le diagnostic. Les trois électrodes actives LA (Left Arm), RA (Right Arm) et LL (Left Leg) prélèvent les dériviations DI et DII, la quatrième, RL, sert à fixer la tension de repos des trois autres.

L'électronique de l'interface amplifie les forces électromotrices prélevées par les dériviations I et II pour produire les signaux DI et DII nécessaires à l'élaboration des électrocardiogrammes. Ces tensions sont converties en grandeurs numériques avant d'être transmises par *Bluetooth* au terminal Android. L'application calcule les dériviations manquantes (DIII, aVR, aVL et aVF) à partir de DI et DII (formules sur la fig. 2b) et les affiche. Les copies d'écran qui illustrent cet article ne laissent subsister aucun doute : les signaux sont excellents, le bruit faible, tout comme le résidu de 50 Hz.

L'électronique analogique

La partie analogique de notre ECG est chargée d'amplifier les deux *très faibles* tensions prélevées entre les électrodes actives LA et RA d'une part et LL et RA d'autre part. Il s'agit des dériviations DI et DII (fig. 2). Pour obtenir une dynamique suffisante après la conversion A/N sur 10 bits, il faut une amplification de 1000 (soit 60 dB). Celle-ci est associée à deux autres fonctions indispensables : la compensation des tensions continues de contact des électrodes (qui peut dépasser 100 mV, soit 100 fois l'amplitude des signaux utiles) et la réjection de la fréquence du secteur (50 Hz). Nous y reviendrons.

Le corps humain et les fils de liaison aux électrodes sont considérablement influencés par les fortes tensions ou différences de potentiel par rapport à la terre, qui règnent sur le câblage électrique du local. La capacité mutuelle, pourtant très faible, provoque l'apparition sur la peau d'une tension par rapport à la terre relativement élevée (souvent supérieure à 1 V) malgré sa fréquence de 50 Hz.

Il semble a priori difficile d'isoler le signal utile, car son amplitude est 1000 fois plus faible que celle du signal parasite ! De plus, la fréquence de 50 Hz est incluse dans le spectre utile ; le filtrage ne convient pas.

Compte tenu de la longueur d'onde du 50 Hz (6000 km !), on admet que, du fait de la conduc-

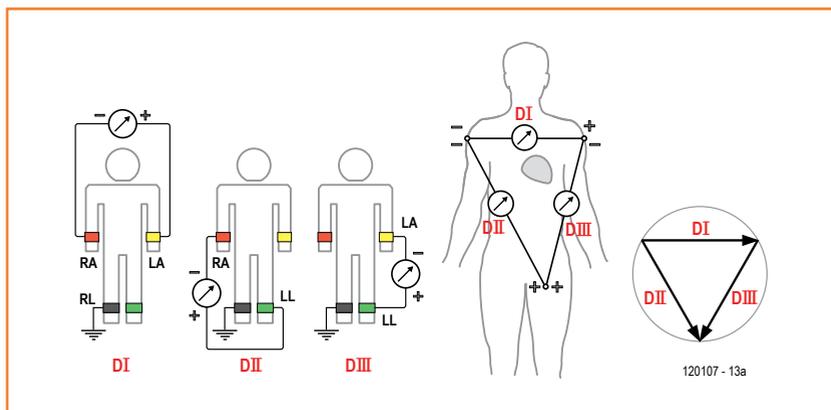


Figure 2a. Dérivations bipolaires standard

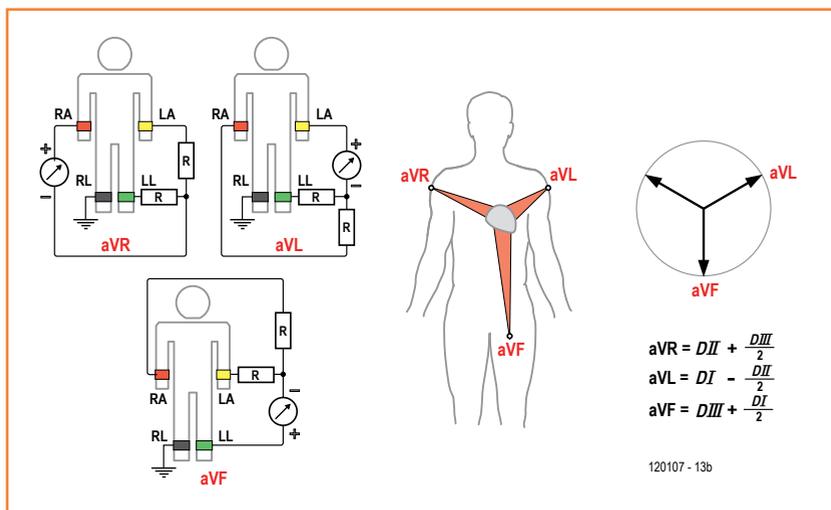


Figure 2b. Dérivations unipolaires standard ou dériviations augmentées.

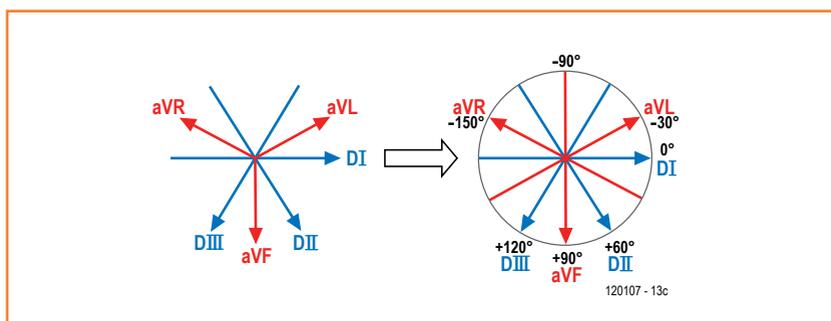


Figure 2c. Système de coordonnées utile au calcul des dériviations.

tivité de la peau, chacun des points de l'épiderme est au même potentiel que les autres. Il s'agit donc, vis-à-vis des électrodes, d'une tension de **mode commun**.

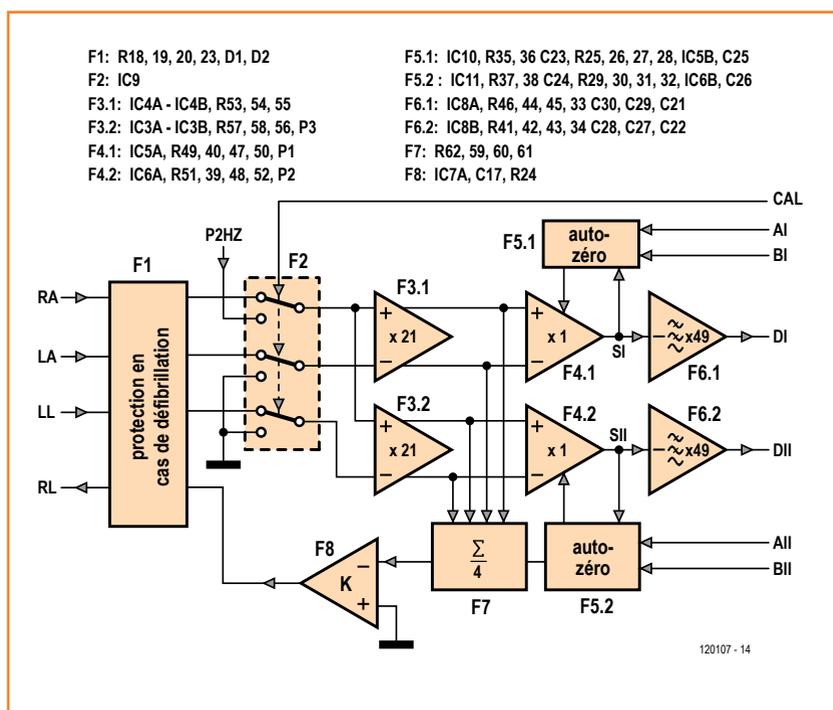


Figure 3. Schéma fonctionnel de la partie analogique

Dans ce cas, la solution devient évidente : un **amplificateur différentiel d'instrumentation** dont le taux de réjection du mode commun (TRMC) soit suffisant :

$$TRMC \geq \left[\frac{S_p}{S_{ECG}} \right]_{dB} + \left[\frac{S}{N} \right]_{dB}$$

où S_p est l'amplitude du parasite : 1 V,
 S_{ECG} l'amplitude ECG : 1 mV,
 S/N le rapport signal sur bruit : 40 dB nécessaires,
 soit : **TRMC $\geq 60+40 = 100$ dB**

L'amplificateur doit en plus être caractérisé par une très forte impédance d'entrée (> 10 M Ω) et une faible tension de décalage. La tentation est grande de faire appel pour cela à un circuit intégré spécialisé (p. ex. ADS1294 de TI). Il est possible aussi, sans aucune concession sur la qualité, de réaliser l'amplificateur différentiel à partir d'amplificateurs opérationnels (AOP) classiques TLC2252, un modèle rail à rail (pour la dynamique), à faible consommation et à faible bruit. Sa fréquence de coupure et sa vitesse de balayage (*slew rate*) sont modestes, mais largement suffisantes pour un signal ECG. La concession majeure (mais acceptable) de ce choix est la résistance ajustable pour optimiser le TRMC dont il n'est pas possible de faire l'économie.

Fonctions de la partie analogique

Avant d'aborder le schéma détaillé de l'interface ECG, examinons-en la structure (**fig. 3**) et les fonctions. Les quatre électrodes sont connectées sur les entrées RA (bras droit), LA (bras gauche), LL (pied gauche) et RL (pied droit). Comme l'électrocardiogramme est parfois utilisé conjointement avec un défibrillateur, il doit donc être protégé contre les fortes tensions produites par ce type d'appareil : c'est la tâche de la fonction **F1**. Dans des conditions normales, les tensions de sortie de cette fonction sont égales à celles d'entrée. La protection agit seulement si les tensions prélevées dépassent le niveau des alimentations, soit $\pm 3,3$ V.

La fonction de multiplexage F2 permet de remplacer les signaux RA, LA et LL par un signal de calibration P2HZ, d'amplitude 1 mV et de fréquence 2 Hz. Le multiplexeur est activé sur demande une fois par minute pendant 10 s pour observer le signal de calibration sur l'écran. Pour établir le diagnostic, le signal de calibration donne une référence en amplitude pour les ECG relevés. L'amplificateur d'instrumentation est constitué des fonctions F3 et F4. La fonction **F6** est un filtre passe-bas du 2^e ordre, de fréquence caractéristique 170 Hz et de facteur d'amortissement $m = 0,71$ (*Butterworth*). Il permet d'atténuer toutes les composantes hors du spectre de fréquences utiles et fait office de filtre anti repliement pour le CAN (**C**onvertisseur **A**nalogique/**N**umérique) qui suit.

L'amplification est obtenue comme suit : $AD3 = 21$, $AD4 = 1$ et $A6 = 49$. L'indice des amplifications correspond à l'indice des fonctions : $AD3$ est l'amplification différentielle de F3 (F3.1 ou F3.2) etc. Conformément aux objectifs, l'amplification totale est de 1029. Celui des deux premiers étages est relativement faible pour augmenter l'efficacité de la fonction d'auto-zéro. Les autres fonctions (F5, F7 et F8) assistent l'amplificateur d'instrumentation pour assurer son bon fonctionnement. En effet, les amplificateurs opérationnels sont alimentés entre $-3,3$ V et $+3,3$ V. La tension de repos idéale sur chacune de leurs trois bornes est de 0 V.

L'électrode RL fixe la tension moyenne (point de repos) des électrodes actives au moyen des fonctions F7 et F8. On montre que la tension de sortie de F7 est égale à la moyenne des tensions RA et

LA et LL. Elle est comparée par F8 à la consigne 0 V et la tension d'erreur amplifiée produit la tension de l'électrode RL. Comme aucun courant ne circule dans les électrodes, les tensions RA, LA et LL sont égales à celle de RL (à quelques mV près). On réalise ainsi un véritable asservissement à travers la peau : la moyenne des tensions des électrodes actives est maintenue égale à la consigne de 0 V. C'est le but recherché : la tension de repos des amplis d'entrée est conforme, sans dégrader leur impédance d'entrée naturelle.

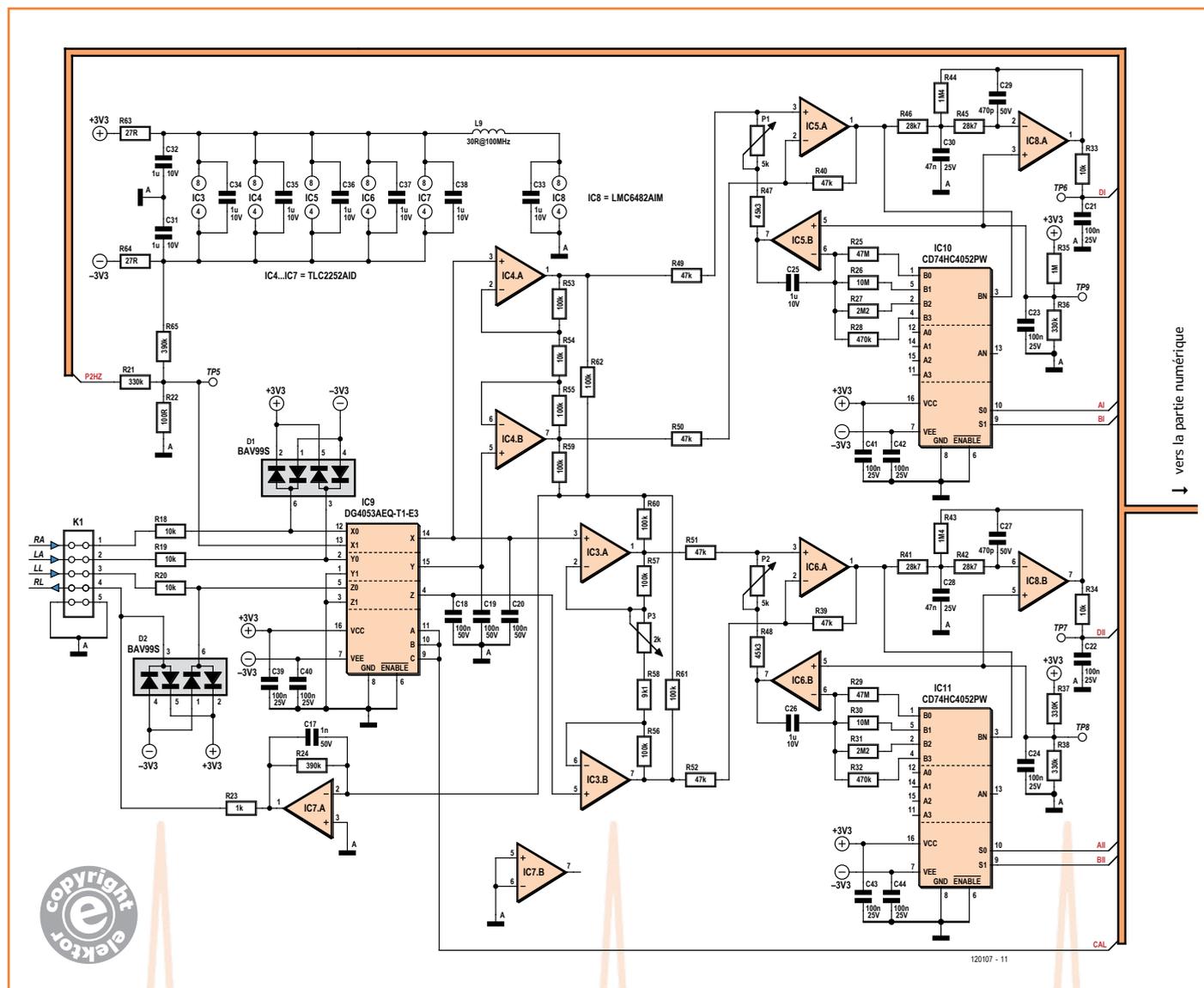
Par ailleurs, un phénomène naturel, mais très gênant, apparaît quand on pose les électrodes : une FEM de contact naît entre la peau et le métal de l'électrode. Cette micro pile est faible

(quelques dizaines de mV) mais n'est pas éliminée par l'amplificateur d'instrumentation. Au contraire, elle est amplifiée !

Les fonctions F7 et F8 réduisent en partie ce phénomène, mais les décalages entre les sorties + et - de F3.1 et F3.2 peuvent atteindre plusieurs volts. Ces décalages sont compensés par les fonctions F5.1 et F5.2 pour éviter les saturations des étages F4.1, F4.2, F6.1 et F6.2.

F5.1 et F5.2 comparent les valeurs moyennes des signaux SI et SII avec une consigne fixe. La tension d'erreur est intégrée avec une constante de temps sélectionnée avec AI et BI (et respectivement AII et BII pour la voie II) pour produire la tension continue de décalage appliquée sur F4.1 et F4.2. Ce décalage se retrouve sur SI et

Figure 4a. Schéma du circuit analogique.



$$T3 = R26 \cdot C25 = 10 \text{ s}$$

$$T4 = R25 \cdot C25 = 47 \text{ s}$$

La plus rapide est sélectionnée à l'application des électrodes de façon à arriver rapidement au point de repos. La constante de temps est ensuite augmentée au fur et à mesure du rapprochement du point de repos attendu, pour finalement atteindre une valeur de 47 s qui n'altère pas les signaux ECG.

Les tensions de repos diffèrent entre les voies DI et DII pour s'adapter aux formes attendues pour chaque dérivation et ainsi éviter une saturation du CAN en aval, tout en gardant une dynamique suffisante.

Le dernier étage de chaque voie (IC8 a et b) constitue un filtre passe-bas du 2^e ordre. Sa fréquence de coupure est de 170 Hz et son gain dans la bande passante est de 34 dB. Il s'agit d'un filtre anti repliement pour le CAN, dont la fréquence d'échantillonnage est de 2000 Hz. Il est suivi d'un passe-bas du 1^{er} ordre constitué de R33 et C21 dont la fréquence de coupure est 160 Hz. L'atténuation des deux filtres à la fréquence de Shannon (1000 Hz) est de 15 dB environ.

Avez-vous remarqué qu'IC8 est alimenté entre +3V3 et 0 V et non entre +3V3 et -3V3 comme les autres AOP ? Ce n'est pas une erreur : on protège ainsi l'entrée analogique du PIC24 qui n'accepte pas de tension négative.

Du fait du gain très élevé de l'amplificateur, la partie analogique est très sensible aux parasites de toute nature, en particulier ceux produits par la partie numérique et l'alimentation à découpage. Le typon a été dessiné avec soin pour bien séparer ces trois parties. En complément, les réseaux R63-C32 et R64-C31 atténuent l'ondulation résiduelle sur les tensions d'alimentation de la partie analogique.

Pour la partie numérique du circuit, le schéma est limpide (**fig. 4b**), nous nous passerons de synoptique.

PIC24FJ32GA002

Le type choisi parmi les variantes du PIC24 de *Microchip* est un modèle de début de gamme, en boîtier SOIC à 28 broches : le PIC24FJ32GA002 (**fig. 4b**). Son horloge interne de 8 MHz (aucun composant externe requis) donne une puissance de 4 MIPS et une précision de 2 %, suffisantes pour cette application. À cette fréquence, sa



Figure 4c.
Photo du module *Bluetooth* de *Roving Networks* utilisé.

consommation de 5,4 mA environ est raisonnable pour une alimentation par piles.

Module Bluetooth RN-42

Le RN-42 (**fig. 4c**) de *Roving Networks* est un module hybride OEM *Bluetooth* de classe 2, pré-qualifié, peu encombrant à faible consommation. L'antenne intégrée permet une portée d'une vingtaine de mètres. Il supporte, entre autres, le protocole RFCOMM-SPP, utilisé dans cette application, à des débits jusqu'à 240 kb/s. Ici on atteint 16000 b/s.

Le profil *Serial Port Profil* (SPP) permet d'établir facilement une sorte de liaison série asynchrone duplex sans fil. Les signaux RX et TX du coupleur UART du μC sont simplement reliés à leurs homologues du module *Bluetooth*. Ainsi, les trames produites sur TX au format série asynchrone sont transférées de façon transparente vers le terminal connecté. Inversement, les messages émis depuis le terminal Android se retrouvent au même format sur RX.

Quelques signaux supplémentaires sont utilisés pour commander le module :

- le signal STATUS indique l'état de la liaison *Bluetooth* : établie (1) ou rompue (0)
- l'entrée RESET permet d'initialiser le module en cas de besoin
- l'entrée PIO7 (SETBAUD) permet de sélectionner le débit de transmission :
 - 1 → 9600 bauds,
 - 0 → 115,2 kbauds
- les résistances de 1 k Ω en série sur ces signaux respectent les recommandations de protection des E/S du fabricant
- le clignotement de la LED D4 indique l'état du module :
 - 10 Hz : configuration en cours
 - 2 Hz : phase d'initialisation
 - 1 Hz : le module est identifiable
 - allumée : connexion établie.

Alimentation

Pour un circuit aussi simple, la structure de l'alimentation (**fig. 4b**) constituée des circuits IC13, IC12 et IC2 est assez complexe, mais elle est garante d'une tension symétrique stabilisée ($\pm 3,3$ V) jusqu'à la dernière goutte d'énergie disponible dans les deux piles AA de 1,5 V ou la paire de batteries de 1,2 V. *Pas question d'alimenter un tel appareil par le secteur !*

Le transistor MOS T1, associé à l'élevateur IC13,

d'IC13 à un niveau H. Sa sortie GATE fait conduire T1 avec une tension égale à trois fois sa tension d'alimentation grâce à un tripleur interne. Cela permet de réduire fortement la résistance R_{ON} de T1. Le - de la pile est connecté au GND.

La sortie GDR de IC13 passe aussi à 1 et valide IC12. Celui-ci produit alors le potentiel de +3V3 à partir de la tension de la pile. L'inverseur IC2 étant maintenant alimenté, il délivre à son tour la tension -3V3. L'interface est sous tension.

Le μ C commence alors son traitement et doit rapidement relayer l'action de l'utilisateur sur S2 en plaçant le signal PowerOff à l'état H.

La mise hors tension est commandée depuis le terminal Android quand l'utilisateur quitte l'application : le signal PowerOff est placé à l'état L. IC13 passe alors en veille (*shutdown*) et place sa sortie GDR à 0 pour inhiber IC12 et ainsi couper l'alimentation.

Les convertisseurs à découpage utilisés permettent d'obtenir un rendement global proche de 90 % et fonctionneront (mais pas longtemps...) avec des piles déchargées jusqu'à 0,8 V.



protège l'électronique en cas d'inversion de la polarité des piles : R6 limite le courant consommé par IC13, T1 est bloqué ainsi que sa diode interne. Le - de la pile n'est pas connecté à GND et les circuits sont protégés, notamment IC12.

Quand la polarité est bonne, IC13 est correctement alimenté. Il est commandé par le signal PowerOff du μ C. Si l'interface est hors tension, le PIC24 n'est pas alimenté et ne peut appliquer un niveau H sur PowerOff. L'entrée SHDN est donc au niveau bas, notamment à cause de R14. La sortie GATE d'IC13 est alors au niveau bas et T1 est bloqué. Par contre, sa diode de roue libre conduit (le courant consommé circule de GND vers le - de la pile) et IC12 est alimenté. Dans ces conditions, la sortie GRD de IC13 est à 0, ce qui inhibe le convertisseur à découpage IC12 qui, par conséquent, ne produit pas la tension d'alimentation 3V3. L'inverseur IC2 qui délivre le -3V3 en aval ne fonctionne donc pas. L'interface est bien hors tension. La consommation se résume à celles d'IC13 et IC12 en veille, soit moins de 4 μ A.

Appuyons sur S2 pour faire passer l'entrée SHDN

Dans le prochain numéro d'Elektor, nous présenterons le logiciel, l'application Android et la carte ainsi que la mise en œuvre de l'appareil. Vous verrez, c'est étonnamment simple.

(120107)

Liens

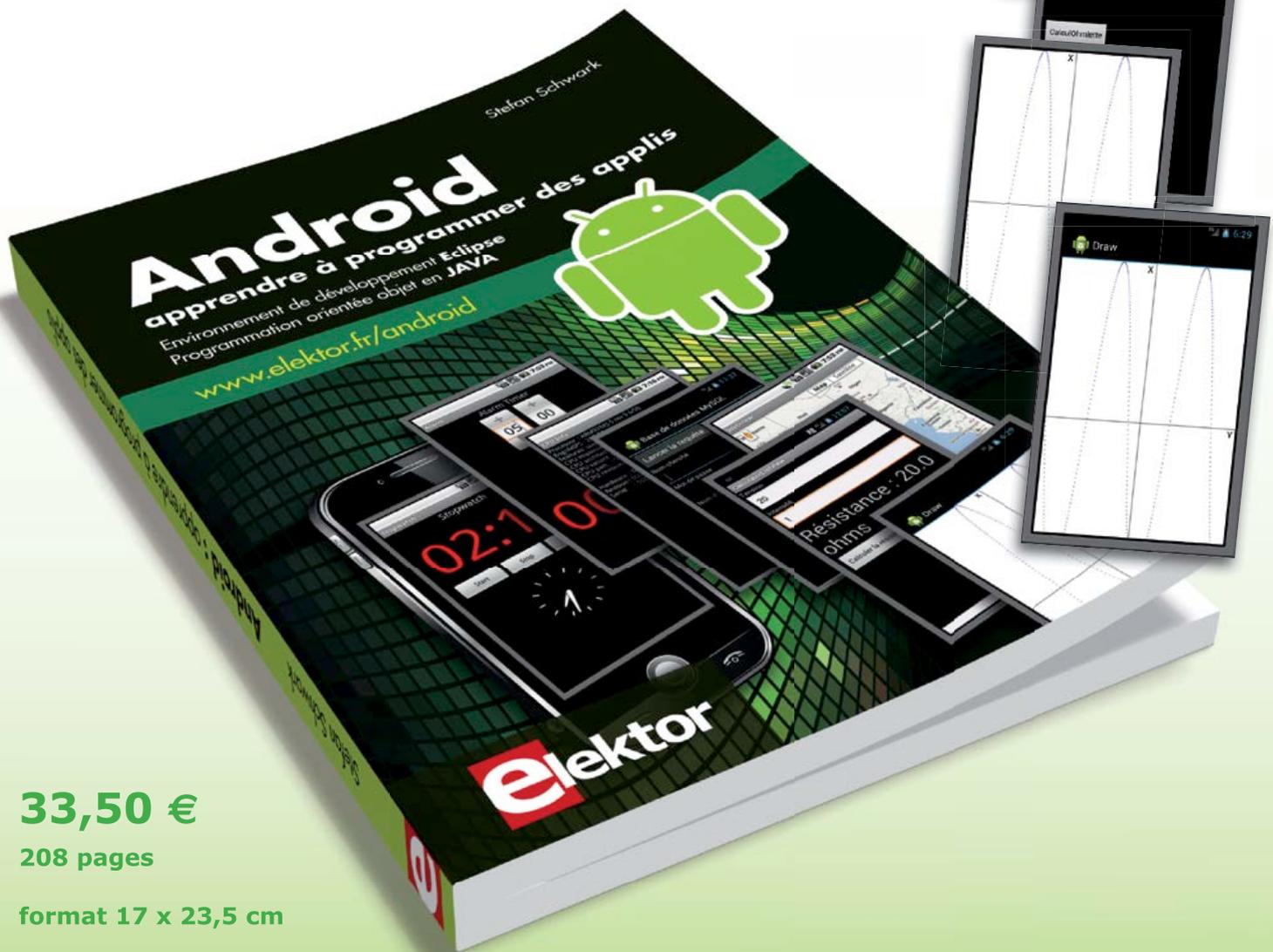
- [1] article GBECG Elektor oct. 2006 p. 32 www.elektor.fr/050280
- [2] Cours de cardiologie <http://goo.gl/mSr20>
- [3] www.elektor.fr/120107
- [4] Android | Apprendre à programmer des apps de Stephan Schwark www.elektor.fr/android
- [5] Le Site du Zéro www.siteduzero.com/informatique/tutoriels/apprenez-a-programmer-en-java ou <http://goo.gl/OVZQY>
- [6] site de l'auteur <http://electronique.marcel.free.fr/>



Télécharger des applications existantes n'est pas le seul moyen de **rendre votre téléphone tactile encore plus intelligent !**

Voici un nouveau livre, publié par Elektor, pour apprendre à construire des applications sous Android et à tirer le meilleur de ce système d'exploitation.

C'est un ouvrage d'initiation, avec des exemples simples, variés et concrets, qui montre de façon progressive comment la combinaison de briques de code permet de créer toutes sortes d'applis dans l'environnement de développement Eclipse :
calculatrice simple, interrogation des capteurs, exploitation des données GPS, communication par l'internet etc.



33,50 €

208 pages

format 17 x 23,5 cm

ISBN 978-2-86661-187-3

elektor

www.elektor.fr/android

niveau à bulle acoustique

ATtiny45 à double emploi

Jörg Trautmann
(Allemagne)

Ce projet a été inspiré par l'inclinomètre à USB du numéro d'été de 2010. L'idée était de construire autour du capteur d'inclinaison MMA7260 un circuit offrant autant d'usages que possible. Le voici aussi bien niveau à bulle acoustique qu'alarme de surveillance.

Mettre à l'horizontale un objet comme une grande table de jardin n'est pas toujours facile. Les points de référence sont rares à l'extérieur, et le maniement d'un niveau à bulle est malaisé. Notre dispositif vient à la rescousse, et grâce à son alarme peut même surveiller un objet : collez-le sous votre table de jardin, et à la moindre tentative de maraude les pieds du voleur aux grandes poches se mettront à bouger bien plus vite que ceux de la table.

Fonctionnement

Le circuit (**fig. 1**) exploite un microcontrôleur ATtiny45 et un accéléromètre à 3 axes MMA7260QT. Le MMA7260 a déjà été utilisé à deux reprises, en 2007 et dans le numéro d'été de 2010 pour un inclinomètre équipé d'un afficheur LCD. Ce petit CI, assemblé sur une carte

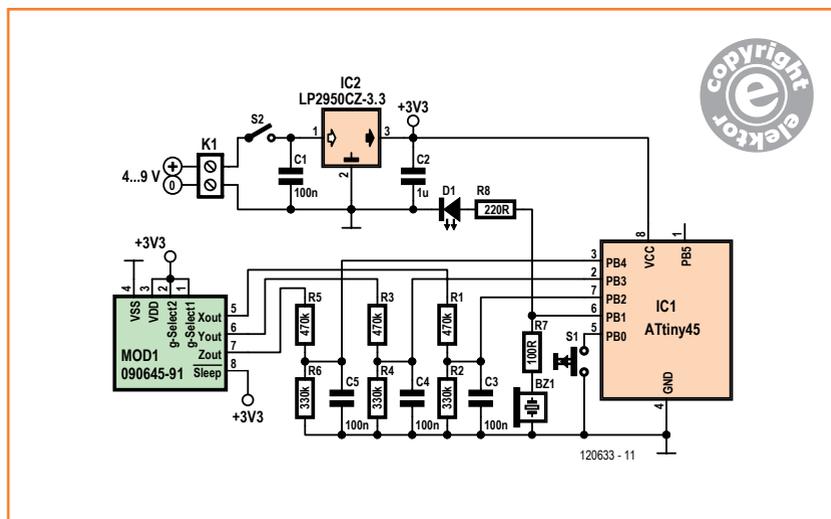
non moins petite (**fig. 2**), possède trois sorties analogiques dont la tension est proportionnelle à l'accélération subie par l'axe associé : à une accélération de +1 g correspondent 2,45 V. Le μC ATtiny d'Atmel [2] dispose de plusieurs convertisseurs A/N, dont trois sont utilisés pour mesurer l'accélération ou les changements d'inclinaison. La tension de référence interne du μC est fixée à 1,1 V, donc les tensions à mesurer doivent passer par un diviseur de tension. D'après les paramètres connus, la valeur des résistances R1, R3 et R5 doit être de 470 k Ω , et de 330 k Ω pour R2, R4 et R6. Avec ce pont, la tension de sortie maximale de 2,45 V est abaissée jusqu'à environ 1 V, ce qui garantit une résolution de mesure maximale.

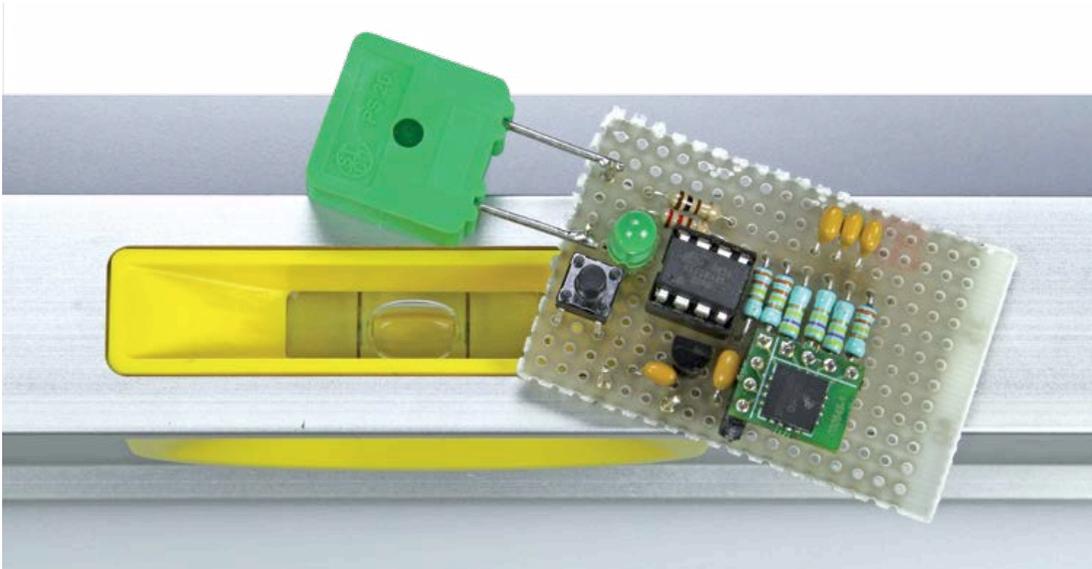
Le μC est programmé pour que toute variation des paramètres X, Y et Z influe sur la fréquence des trois générateurs de son.

Les générateurs restent muets lorsque le circuit est horizontal. Dès que l'inclinaison dépasse ± 2 degrés, ils émettent un son qui change à mesure que l'écart avec l'horizontale augmente. Le bouton S1 sert à la fois à calibrer le circuit et à choisir son mode de fonctionnement. Le mode alarme s'enclenche en appuyant sur S1 durant plus de 5 s.

Le montage est alimenté par l'intermédiaire d'un régulateur à faible tension de déchet (*Low Dropout*), le LP2950CZ3.3. Il produit une tension optimale de 3,3 V pour le μC et l'inclinomètre, et grâce à lui une pile de 9 V durera très longtemps. Lors des essais, nous avons établi que la tension d'alimentation minimale était de 3,6 V, le cou-

Figure 1.
Le circuit est si petit qu'il peut être assemblé sur un morceau de plaque d'essai.





rant maximal de 4,56 mA, et le courant moyen d'environ 3,06 mA lorsque la LED clignote et que l'alarme retentit.

Assemblage et mise en service

Le montage peut être réalisé facilement sur une plaque d'expérimentation, p. ex. la plaque d'essai ELEX-1. Lorsque le circuit est mis sous tension pour la première fois, la LED rouge doit rester allumée et le haut-parleur rester muet. Si ce n'est pas le cas, revérifiez votre assemblage ! Pour calibrer le circuit, placez-le sur une surface que vous savez plane (le sol l'est... a priori), et appuyez sur le bouton S1 pendant à peu près 1 s. La LED s'éteint lorsque vous le relâchez, ce qui indique que la valeur correspondant à la position horizontale a été enregistrée. Fin du calibrage, le dispositif doit maintenant être silencieux. Inclinez-le : vous devriez voir la LED clignoter, et entendre un, deux, puis trois bips, dont la fréquence varie en fonction de l'inclinaison. Le pépiement cesse et la LED s'éteint lorsque le dispositif retrouve l'horizontale. Placez le dispositif dans un boîtier plat : vous verrez que même les yeux bandés on apprend vite à le positionner correctement.

Si vous voulez utiliser le circuit pour la surveillance d'objets, fixez-le d'abord sur l'objet en question puis appuyez sur la touche de calibrage (l'objet ne doit donc pas nécessairement être en position horizontale). Enfoncez une nouvelle fois le bouton S1, mais cette fois-ci pendant plusieurs secondes, jusqu'à ce que la LED clignote régulièrement. Une fois le bouton relâché, le système est « armé ». Une alarme de sirène de cambriolage retentit lorsque l'objet est incliné d'environ

20 °. Appuyez brièvement sur S1 pour l'arrêter : le système est de nouveau en alerte. Pour le remettre en mode « niveau à bulle », mettez-le hors tension un bref instant. Les dernières valeurs de calibrage enregistrées restent en mémoire.

Le programme

Vous pouvez télécharger le programme depuis la page associée à ce projet [3]. Dans le code, écrit en BASCOM AVR, la broche PB1 est configurée en sortie pour le pilotage du résonateur piézo-électrique, et PB0 est définie en tant qu'entrée avec une résistance de forçage au niveau haut. Les convertisseurs ADC0, ADC1 et ADC2 ont été paramétrés pour fonctionner avec une référence de tension interne de 1,1 V. Les valeurs de calibrage sont sauvegardées dans l'EEPROM lorsque le poussoir S1 est fermé (PB0 au niveau bas).

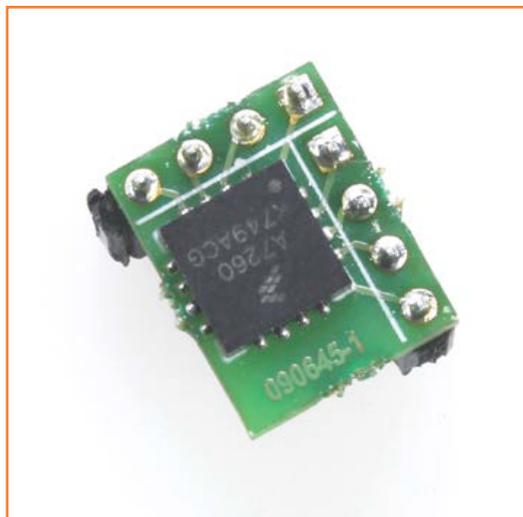


Figure 2.
Le capteur sur sa micro-carte adaptatrice.

Remarque

Freescall ne fabrique plus le MMA7260QT qu'on trouve encore, mais de plus en plus difficilement. En avril 2013, Elektor en possédait encore un stock de 67 pièces (090645-91) que vous pouvez commander à l'unité. Les premiers arrivés seront les premiers servis ! Le programme (BASC0M-AVR) peut toutefois être facilement adapté pour être utilisé avec un autre capteur [4].

Ces valeurs sont relues et servent de référence après chaque mise sous tension du circuit. La logique pour l'évaluation de l'état de PB0 est programmée de façon à ce que le calibrage ne soit possible que si le programme n'est pas en mode alarme. Sinon, appuyer sur S1 remet le programme en mode alarme. Un seul bouton remplit donc deux fonctions.

Un cycle de mesure comprend sept mesures sur 3 canaux effectuées dans un intervalle de 210 ms, puis suivies du calcul de la valeur moyenne. Il en

résulte une précision de la mesure et une stabilité satisfaisante. Si vous souhaitez modifier la sensibilité d'un mode, adaptez en conséquence la valeur de la variable *Trigger_value*.

Si la perspective d'implanter vous-même ce programme ne vous emballa pas, vous pouvez vous procurer un μC déjà programmé auprès d'Elektor.

(120633 – version française : Hervé Moreau)

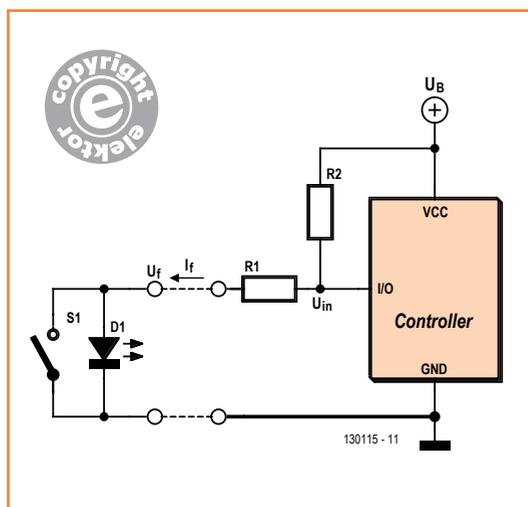
Liens

- [1] www.elektor.fr/070829
- [2] www.atmel.com/devices/ATTINY45.aspx
- [3] www.elektor.fr/120633
- [4] Capteur d'inclinaison Low-g : www.freescale.com/webapp/sps/site/taxonomy.jsp?nodeId=01126911184209#2

1 bouton + 1 LED = 2 fils seulement

Version 3.0

John Hind
(Royaume-Uni)



Deux fils suffisent pour relier un poussoir et sa LED témoin à un microcontrôleur. Klaus Thieslers nous l'avait montré dans les deux astuces « **1 bouton + 1 LED = 2 fils seulement** » dont la 1^{ère} version [1] et la version améliorée [2] utilisaient deux transistors et plusieurs autres composants. Trop selon John, qui a simplifié le montage, en ne recourant cette fois-ci qu'à deux résistances et à une seule broche GPIO. Qui fera mieux ?

Moins de composants signifie plus de travail pour le μC . Le μC doit pouvoir commuter une broche d'E/S entre les modes entrée et sortie. Cette condition préalable est la plupart du temps satisfaite. Broche configurée en sortie, le μC pourra

Tableau 1.

État	Br. E/S	Niveau	Poussoir	LED	I_f	U_{in}
1	entrée	haut	ouvert	off	$(U_B - U_f) / (R1 + R2)$	$U_f + I_f * R1$
2	entrée	bas	fermé	off	$U_B / (R1 + R2)$	$I_f * R1$
3	sortie	haut	ouvert	on	$(U_B - U_f) / R1$	U_B
4	sortie	haut	fermé	off	$U_B / R1$	U_B

Tableau 2.

Couleur LED	U_f	I_f	@ 5 V: R1 R2	@ 3,3 V: R1 R2	@ 2,1 V: R1 R2
rouge	1,7 V	10 mA	330 Ω 470 k Ω	160 Ω 220 k Ω	39 Ω 56 k Ω
orange, jaune	2,1 V	10 mA	300 Ω 430 k Ω	120 Ω 180 k Ω	-
verte	2,2 V	10 mA	270 Ω 390 k Ω	110 Ω 160 k Ω	-
bleue, blanche	3,6 V	20 mA	68 Ω 200 k Ω	-	-

allumer la LED avec un niveau haut, et avec la broche définie en entrée il pourra scruter l'état du poussoir lorsque la LED est éteinte. Les deux premières lignes du **tableau 1** montrent l'état de la broche configurée en entrée : selon l'état du poussoir, la tension U_{in} est interprétée comme un niveau bas ou haut – si R1 et R2 sont choisies correctement en fonction de la tension d'alimentation U_B , car la plupart des μC ont un seuil de commutation haut de l'ordre de la moitié de la tension U_B .

Ces lignes sont claires, mais les deux autres ? Comment les états 3 et 4 du poussoir peuvent-ils être scrutés lorsque la broche d'E/S est configurée en sortie ? Simple : la broche est configurée en entrée plusieurs fois par seconde et prend les états 1 et 2 durant un temps si court que l'œil humain ne voit rien. Si le poussoir est vu comme « enfoncé », la broche garde l'état 2 jusqu'à ce qu'il soit relâché, car l'état 4 serait de toute façon identique (la LED reste éteinte), si l'on excepte la circulation d'un courant inutile. La broche reprend toutefois aussitôt l'état 3 afin que la LED puisse continuer à briller.

En plus du traitement anti-rebond de base, nous pourrions même nous permettre le luxe d'une commande de la luminosité de la LED en commutant simplement les états 1 et 3 de façon rapide. Le PIC16F883 [3] utilisé est équipé, comme les contrôleurs AVR, de résistances internes de rappel au niveau haut qu'il est possible d'activer. En théorie, et avant tout pour une tension U_B basse, elles pourraient remplacer R2, ce qui donnerait un

montage avec une seule résistance. La plage de valeurs de ces résistances *pull-up* s'étend hélas de 10 à 50 k Ω , valeurs avec lesquelles l'éclairage de la LED à l'état 1 peut devenir perceptible. Le micrologiciel n'active donc les *pull-up* que le temps de scruter l'état du poussoir.

R2 doit dans tous les cas être dimensionnée de façon à ce que le seuil de commutation de l'entrée soit dépassé sans ambiguïté, car avec des petits courants la tension directe U_f chute. Pour remédier à cette situation, on branchera p. ex. une diode au silicium en série avec la LED dans le cas d'une LED rouge et d'une alimentation de 5 V. Le dimensionnement des résistances en fonction de la tension d'alimentation et de la couleur de la LED est indiqué dans le **tableau 2**. Pour les intensités absentes, livrez-vous à un peu de calcul !

(130115 – version française : Hervé Moreau)

Liens

- [1] 1 bouton + 1 LED = 2 fils seulement, Elektor mai 2012, p. 35 ; www.elektor.fr/110572
- [2] 1 bouton + 1 LED = 2 fils seulement, version améliorée, Elektor janvier/février 2013, p. 47 ; www.elektor.fr/120071
- [3] Micrologiciel : www.elektor.fr/130115



2^e jour – insérer un cartouche

Ce que l'on conçoit bien s'énonce clairement

Neil Gruending
(Canada)

Dans mon article « 1^{er} jour » [4], nous avons vu comment configurer DesignSpark PCB à l'aide de fichiers de styles et de modèles. Voyons ici comment insérer un cartouche dans un schéma à l'aide des bibliothèques.

Nous allons ici créer un symbole de documentation qui servira au cartouche d'un schéma. Vérifions d'abord les chemins d'accès aux bibliothèques. Allez dans *File/Libraries*, puis dans l'onglet *Folders* (**fig. 1**). Dans la boîte *Folders and Search Order*, assurez-vous que vos fichiers de bibliothèque soient les premiers de la liste (C:\Users\Neil Gruending\Documents\dspcb2lib\library dans mon cas). Au besoin, remontez-les en tête de liste à l'aide des boutons *Up/Down*. Lors d'une mise à jour de DesignSpark, le répertoire par défaut des bibliothèques peut être écrasé : n'y sauvegardez donc aucun fichier nouveau ou modifié.

Les chemins d'accès étant corrects, créons une nouvelle bibliothèque de type *Schematic Symbols* qui servira à la création de notre cartouche. Cliquez sur le bouton *New Lib...* de l'onglet *Schematic Symbols*, puis sur le bouton *New Item...* pour ouvrir une page vierge de type *Schematic Symbol* (vous trouverez plus d'informations dans le tutoriel [1]).

Création du cartouche

J'ajoute toujours un cartouche à mes schémas. Un cartouche donne un aspect professionnel et permet de documenter le projet comme il se doit. DesignSpark diffère ici des autres logiciels, puisque les cartouches sont stockés dans une bibliothèque de composants de schéma, et non pas dans un fichier de style ou de modèle ; DesignSpark ignore donc tous les éléments de dessin contenus dans ces fichiers. La bibliothèque *Schema* contient des modèles de cartouches de différentes dimensions, p. ex. A4 ou Letter. J'use volontiers du format Ledger (11 in. x 17 in.), et préfère les zones de documentation dont la surface n'empiète que peu sur le dessin.

Dans le précédent article, j'avais montré comment utiliser des polices TrueType, plus agréables à l'œil selon moi. Leur emploi serait toutefois trop incertain ici, car DesignSpark les décale légèrement vers le bas lors de la conversion des schémas au format PDF. Ce travers est sans importance pour des éléments comme les identifiants, mais, dans un cartouche, l'œil remarquerait immédiatement

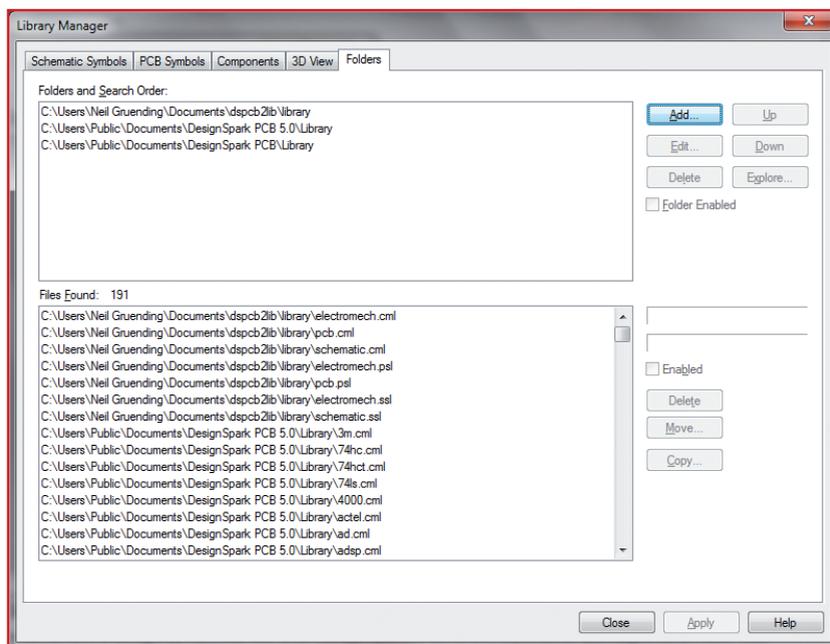


Figure 1.
Les chemins d'accès aux bibliothèques.

Configuration des bibliothèques

DesignSpark se sert de bibliothèques pour organiser l'ensemble des informations relatives à un schéma. On trouve ainsi une bibliothèque pour les symboles de schéma (*Schematic Symbol*), et une pour les empreintes (*PCB Symbol*). Leur combinaison forme une bibliothèque de composants (*Component library*) que vous pouvez utiliser pour placer dans votre projet des composants et des symboles de documentation.

La seule différence entre un symbole de documentation et un composant normal est que le premier contient soit un symbole de schéma, soit une empreinte, mais pas les deux. Le tutoriel [1] explique plus en détail le fonctionnement de ce système de bibliothèques, et vous familiarisera avec sa terminologie.

un mauvais alignement du texte. J'ai donc utilisé la police par défaut (**fig. 2**).

Donnez des noms parlants à vos styles de texte, vous les identifierez plus rapidement.

La figure 3 montre les miens : le style *Frame* est celui des chiffres et des lettres qui définissent les coordonnées du dessin, *Title* celui des titres des différents champs du cartouche, et *Title-small* celui du contenu des champs. DesignSpark ne gère pas les variables de projet, et les chaînes de texte du cartouche doivent donc être ajoutées à la main, ce qui explique que tous les champs du cartouche soient blancs dans le composant de schéma. Il est du reste inutile d'insérer ces styles de texte dans le fichier de style, ils seront copiés dans le schéma lorsque vous ajouterez votre cartouche.

Sauvegardez vos styles dans la bibliothèque *Schematic Symbol* créée précédemment, nous en avons bien sûr besoin pour créer notre symbole de schéma. Ouvrez ensuite le gestionnaire de bibliothèques (*File/Librairies...*), puis l'onglet *Components*. Sélectionnez le bouton *New Lib...*, puis le bouton *New Item...* pour ouvrir la fenêtre *New Component*. Donnez un nom au composant (zone *Component*), et sélectionnez votre symbole de cartouche. Décochez la case à cocher *PCB Symbol* pour que votre composant soit du type souhaité, à savoir un symbole de schéma. Sauvegardez vos changements.

Création d'un nouveau projet

DesignSpark réunit dans une même source appelée « projet » toutes les informations relatives à une conception, p. ex. les schémas ou les fichiers de routage. Utiliser un projet, c'est lier un ensemble de feuilles de schéma à une conception PCB. Ces feuilles de schéma liées représentent alors un seul vaste projet, dans lequel les informations sur les équipotentielles (les *nets*) sont partagées entre chaque feuille et où chaque composant possède un nom unique.

Créer un nouveau projet DesignSpark est simple : cliquez sur *File/New*, dans la fenêtre *New Design* choisissez *Project*, puis validez avec *OK*. Le programme vous demande le répertoire de sauvegarde du projet, puis crée un projet vierge.

Vous pouvez y ajouter des fichiers existants depuis le menu *Project/Add Files to Project...* L'ajout de nouveaux éléments s'effectue depuis la fenêtre *New Design*, mais avant de cliquer sur *OK* assurez-vous d'avoir coché la case *Add to Open Project*.

Conclusion

Vous savez créer des projets et des modèles de schéma, vous êtes donc prêt à utiliser les composants de la bibliothèque DesignSpark pour dessiner. Vous pouvez également créer vos propres bibliothèques, et affecter à vos composants des attributs qui faciliteront leur nomenclature. Le cartouche que j'ai conçu pour cet article (**fig. 4**) peut être téléchargé sur le site Bitbucket depuis la page associée à mon projet dsppcb2lib [3].

(130181 – version française : Hervé Moreau)

Liens

- [1] www.designspark.com/tutorial/components-library-structure-library-manager
- [2] www.designspark.com/tutorial/components-creation-with-symbol-footprint-wizards
- [3] <https://bitbucket.org/neilg/dspcb2lib>
- [4] www.elektor.fr/130172

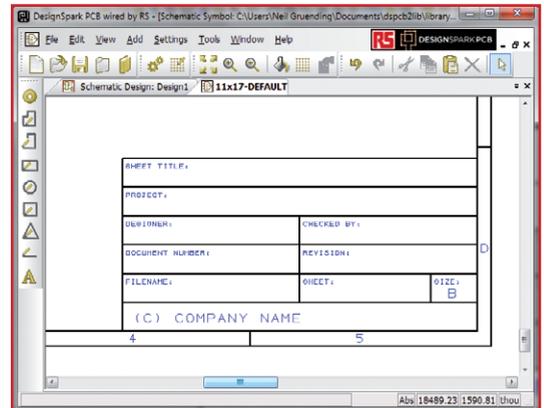


Figure 2. Structure et polices du cartouche.

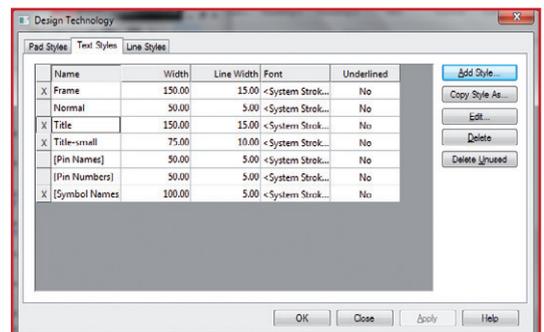


Figure 3. Les styles de texte pour le cartouche.

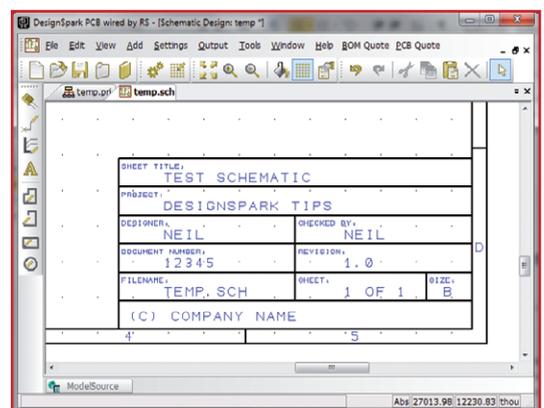


Figure 4. Le cartouche fini.

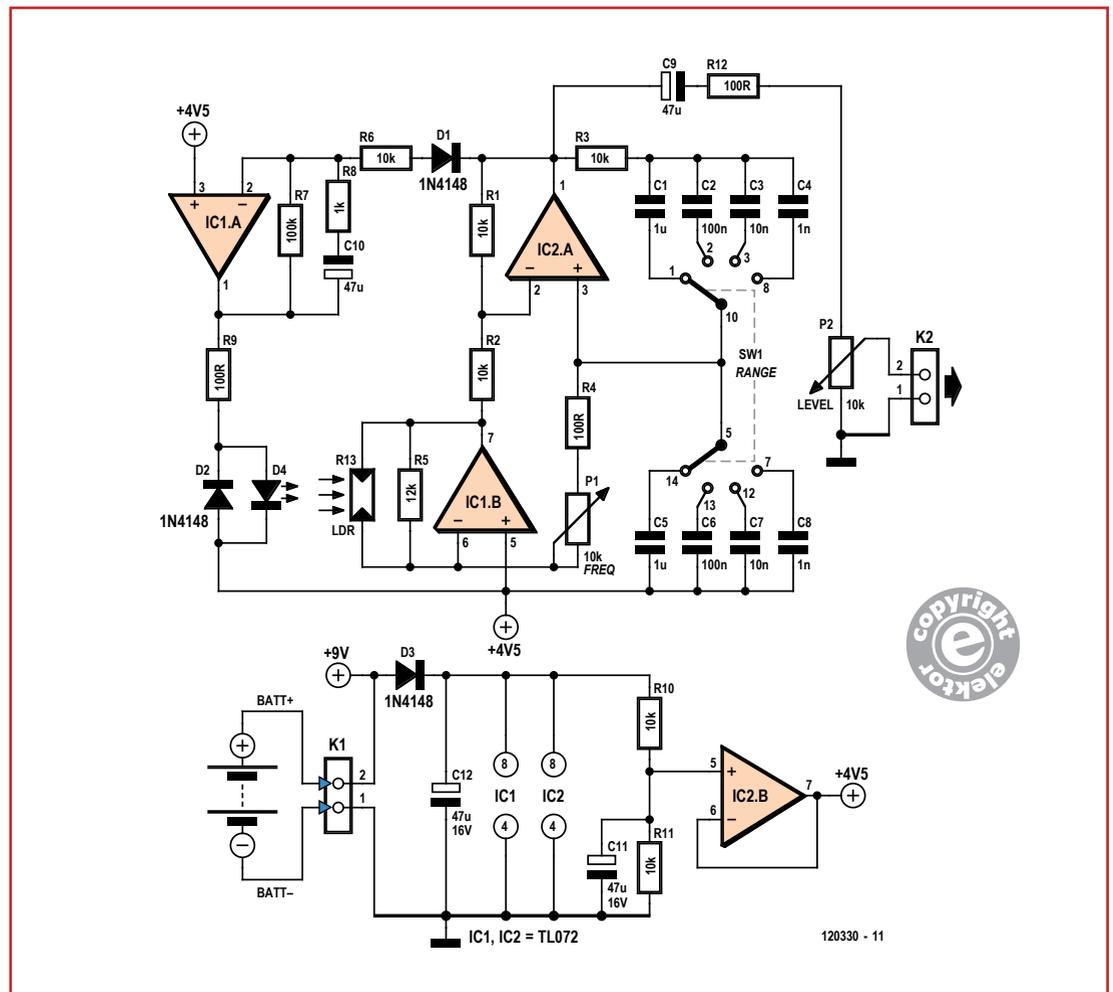


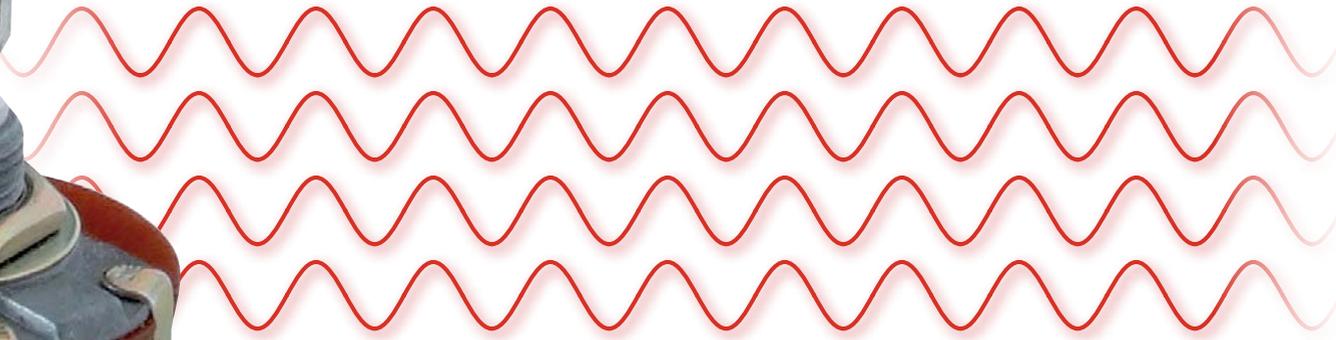
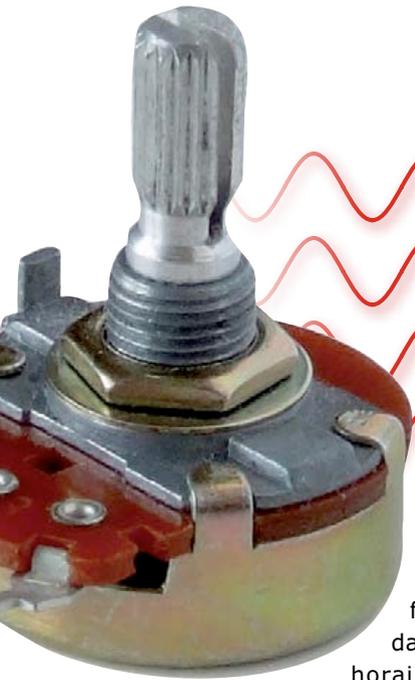
oscillateur à large bande à pont de Wien avec potentiomètre à simple piste

Merlin Blencowe
(Royaume-Uni)

Cet oscillateur produit une onde sinusoïdale d'amplitude constante à faible distorsion d'environ 15 Hz à 150 kHz, avec quatre ampli-op alimentés par une pile (9 V). Contrairement à la plupart des oscillateurs à pont de Wien, il n'utilise pas de potentiomètre à double piste. L'ampli-op IC2b crée une masse artificielle permettant de se passer d'une alimentation symétrique. IC2a est l'amplificateur principal. La plage de fréquences de sortie est répartie en quatre décades sélectionnées par le commutateur rotatif SW1.

Seul l'un des bras du réseau de Wien est réglable, mais la rétroaction positive qui se produit alors est compensée par IC1b en élevant le potentiel de référence de R2 ; cela change suffisamment la rétroaction négative pour entretenir l'oscillation. Un changement *linéaire* de la résistance du potentiomètre de réglage entraîne grosso modo un changement *logarithmique* de la fréquence. Pour obtenir une variation linéaire, plus conventionnelle, un potentiomètre à échelle logarithmique est utilisé pour





P1 afin qu'en le faisant tourner dans le sens antihoraire la fréquence augmente. Si vous arrivez à dénicher un, vous pourrez utiliser un potentiomètre anti-log dans l'autre sens. L'intégrateur IC1a surveille l'amplitude du signal de sortie et pilote la LED D2 qui devra être montée face à la photorésistance (LDR) et isolée de la lumière ambiante (p.ex. avec un morceau de gaine thermorétractable). IC1a pourra ainsi commander le gain d'IC2a afin que l'oscillation soit maintenue avec une distorsion minimale. L'amplitude maximale en sortie du générateur

est d'environ $2 V_{c-c}$ lorsque la LED et la LDR sont l'une contre l'autre. La distorsion est inférieure à 0,5 % dans la gamme inférieure et négligeable dans les autres gammes. N'importe quelle LDR devrait convenir si sa résistance dans l'obscurité est supérieure à 100 k Ω . Si vous n'en trouvez pas, essayez d'augmenter R5 jusqu'à ce que l'oscillation démarre. Les deux prototypes assemblés par l'auteur avec des doubles et quadruples ampli-ont bien fonctionné.

Le schéma et les fichiers CAO DesignSpark de la carte sont disponibles sur www.elektor.fr/120330.

(120330 - version française : Kévin PETIT)

Liste des composants

Résistances

R1,R2,R3,R6,R10,R11 = 10 k Ω
 R7 = 100 k Ω
 R4,R9,R12 = 100 Ω
 R5 = 12k Ω
 P2 = pot. lin. 10 k Ω
 P1 = pot. log. 10 k Ω
 R13 = LDR, R(obscurité) >100 k Ω , p.ex. Excelitas
 Tech type VT90N1 (Newark/Farnell 2568243)

Condensateurs

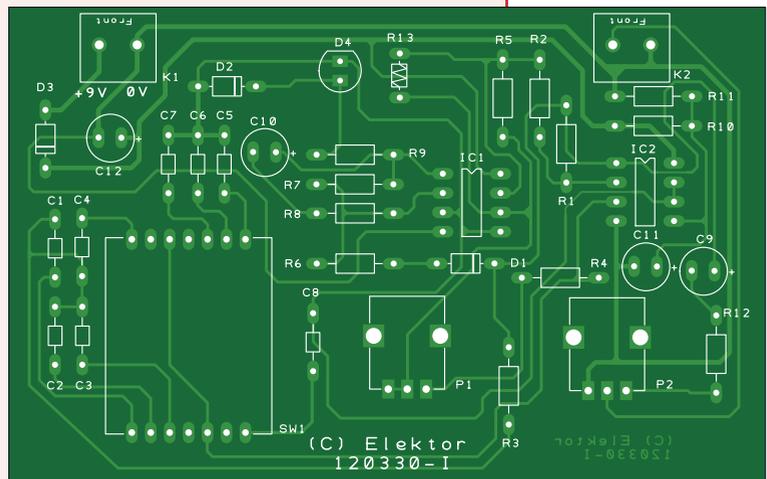
C1,C5 = 1 μ F, à électrolyte solide
 C2,C6 = 100 nF
 C3,C7 = 10 nF
 C4,C8 = 1 nF
 C9-C12 = 47 μ F 16V, électrolytiques radiaux

Semiconducteurs

D1,D2,D3 = 1N4148
 D4 = LED rouge 5 mm
 IC1,IC2 = TL072ACP

Divers

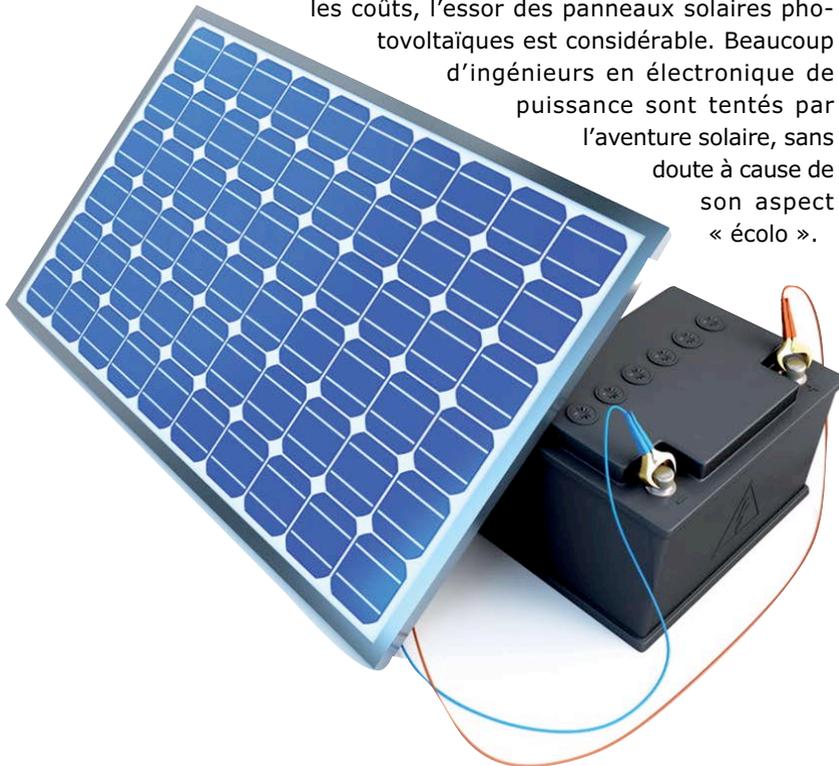
SW1 = commutateur rotatif 2 circuits 4 positions, C&K
 RTAP42S04WFLSS



K1,K2 = borniers pour CI au pas de 5 mm
 circuit imprimé 120330-1

panneau solaire : contrôleur de charge 4 A

Qu'il soit dû aux inquiétudes liées au réchauffement climatique ou à la volonté de réduire les coûts, l'essor des panneaux solaires photovoltaïques est considérable. Beaucoup d'ingénieurs en électronique de puissance sont tentés par l'aventure solaire, sans doute à cause de son aspect « écolo ».



T. A. Babu (Inde)

Le circuit dont il est question ici accepte jusqu'à 4 A depuis un panneau solaire, soit une puissance d'environ 75 W.

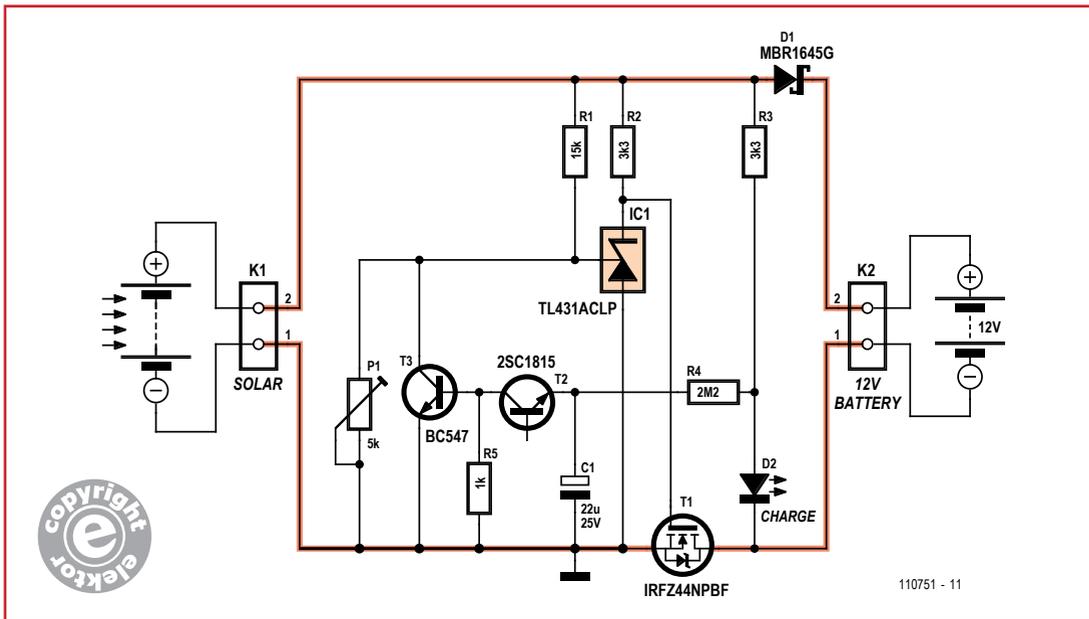
Le courant circulant du panneau solaire vers la batterie est commandé par T1, un MOSFET à canal N. Comme la $R_{DS(on)}$ de ce dernier n'est que de 24 m Ω , on peut se passer de dissipateur. La diode Schottky D1 non seulement empêche la batterie de se décharger dans le panneau la nuit, mais la protège des inversions de polarité. Sur le schéma, les fils en rouge indiquent le chemin pris par les courants de forte intensité.

Le contrôleur de charge est entièrement alimenté par le panneau solaire et ne tire jamais de courant de la batterie. La nuit, il se met tout simplement en veille. Dès que le panneau solaire fournit une tension et un courant suffisants, la charge commence.

La tension aux bornes de la batterie, divisée par R1 et P1, déclenche ou inhibe la charge via le contrôleur. Le cœur de ce dernier est IC1, une référence de tension de type TL431ACZ dotée d'un amplificateur d'erreur à collecteur ouvert. Le montage compare constamment la tension issue du diviseur R1/P1 avec la tension de référence interne au TL431. Tant que le niveau réglé par P1 est inférieur à la référence, IC1 autorise la conduction du MOSFET. À mesure que la batterie se charge, la tension à ses bornes augmente. Lorsqu'elle dépasse le seuil à partir duquel la batterie est considérée comme chargée, le potentiel en sortie d'IC1 chute à un peu moins de 2 V, le MOSFET se bloque, la LED D2 s'éteint, et plus aucun courant ne circule dans la batterie.

IC1 n'a plus d'image de la tension à ses bornes et le MOSFET restera donc bloqué, même si la tension de batterie baisse. Comme les batteries plomb-acide nécessitent une charge d'entretien, il m'a fallu ajouter l'oscillateur très simple que vous voyez sur le schéma. Celui-ci exploite la résistance négative des transistors, découverte par Leo Esaki alors qu'il étudiait l'effet tunnel dans les solides, et qui lui a valu le prix Nobel de physique en 1973.

J'ai utilisé un transistor NPN 2SC1815 tout ce qu'il y a de plus courant. Lorsque la LED s'éteint, C1 se charge via R4 jusqu'à ce que la tension à ses bornes soit suffisamment élevée pour provoquer un effet d'avalanche dans la jonction base-émetteur de T2. Ce transistor commence alors rapidement à conduire et décharge le condensateur à travers R5. La tension aux bornes de R5 suffit alors pour activer T3 qui agit sur la tension en entrée d'IC1. Le MOSFET devient passant et la charge de la batterie peut reprendre. Dès que la tension aux bornes de la batterie atteindra à nouveau le seuil d'arrêt, le processus se répétera. Si le 2SC1815 s'est montré fiable dans ce circuit, d'autres transistors pourraient l'être moins – pour en comprendre la raison, si vous ne craignez pas une indigestion de mathématiques, il ne vous reste qu'à étudier le travail d'Esaki.



À mesure que la batterie se charge, le temps pendant lequel T3 est passant raccourcit tandis que le temps pendant lequel il est bloqué, déterminé par R4 et C1, ne varie pas. La durée de l'impulsion de courant envoyée à la batterie se raccourcit donc peu à peu. C'est ce qui fait de ce chargeur un modèle à *modulation de durée d'impulsion*.

Pour régler ce circuit, il vous faudra un bon multimètre numérique et une alimentation réglable. Réglez l'alimentation sur 14,9 V, soit les 14,3 V de la batterie plus 0,6 V aux bornes de la diode Schottky. Manœuvrez l'ajustable P1 jusqu'à ce que la LED s'éteigne, c'est le point de basculement, puis commence à clignoter. Il faudra peut-être vous y reprendre à plusieurs fois :

plus la tension de basculement du comparateur est proche de 14,3 V et plus le chargeur sera précis. Déconnectez l'alimentation du contrôleur de charge : le montage est prêt à être relié à un panneau solaire. La valeur de 14,3 V devrait convenir à la plupart des modèles de batterie plomb-acide, qu'il s'agisse d'un modèle au gel ou non, mais cela ne vous dispense pas de jeter un coup d'œil à la feuille de caractéristiques de votre batterie. Le courant fourni par le panneau solaire ne devra pas dépasser l'intensité que la batterie est capable d'encaisser.

Schéma et fichiers de CAO disponibles sur www.elektor.fr/110751.

(110751 - version française : Kévin PETIT)



Liste des composants

Résistances

R1 = 15 kΩ
 R2,R3 = 3,3 kΩ 1%
 R4 = 2,2 MΩ
 R5 = 1 kΩ
 P1 = 5 kΩ

Condensateurs

C1 = 22 µF 25 V, radial

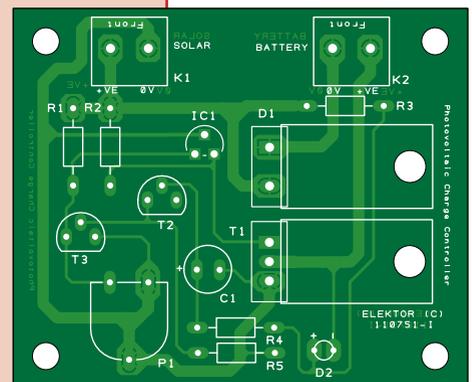
Semiconducteurs

D1 = MBR1645G (ON Semiconductor)
 D2 = LED 5 mm

IC1 = TL431ACLP (Texas instruments)
 T1 = IRFZ44NPBF (International Rectifier)
 T2 = 2SC1815 (Toshiba) (marquage C1815)
 T3 = BC547

Divers

K1,K2 = bornier pour C.I. à deux voies au pas de 5 mm
 circuit imprimé 110751-1





le jeu des différences

Wisse Hettinga

Renesas répond aux tendances du marché en mettant en avant la forte compatibilité Arduino de leur dernière carte : la **GR Sakura**.
Appréciez par vous-même son étendue !

Différence n°1

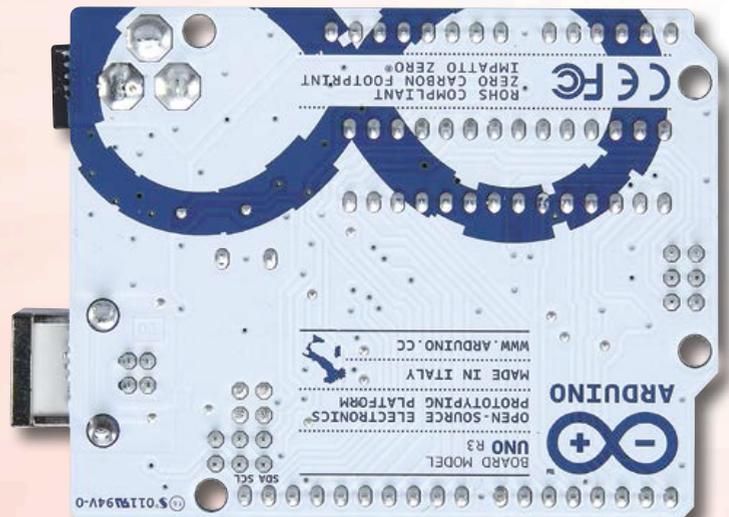
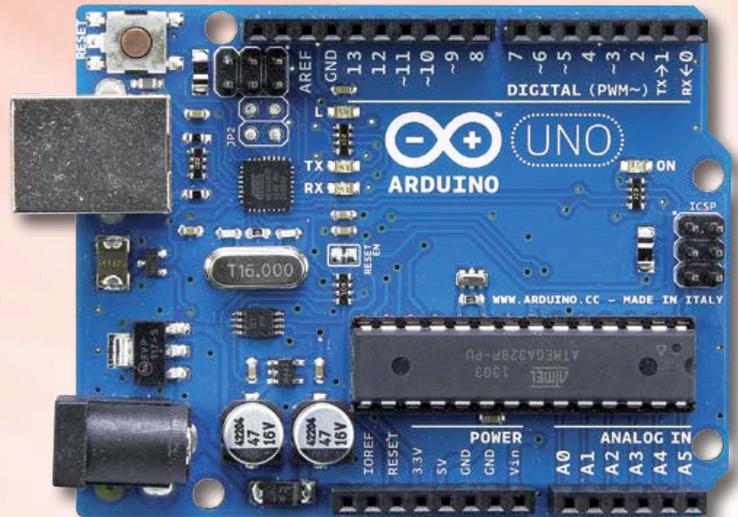
La couleur de la *Sakura*. Avec ça, on les distinguerait même dans le noir. Au fait saviez-vous qu'en japonais, Sakura signifie cerisier ? Ceci explique cela.

Différence n°2

Comparez les 8 bits, 16 MHz et la mémoire limitée de l'AVR aux 32 bits, 96 MHz et à la tonne de mémoire du processeur *Renesas*. La vraie question est de savoir quelles applications mettront à profit tout le potentiel de la *GR Sakura*.

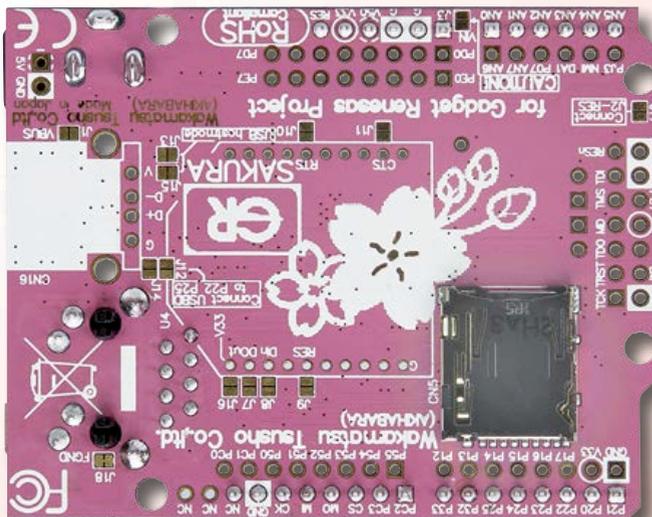
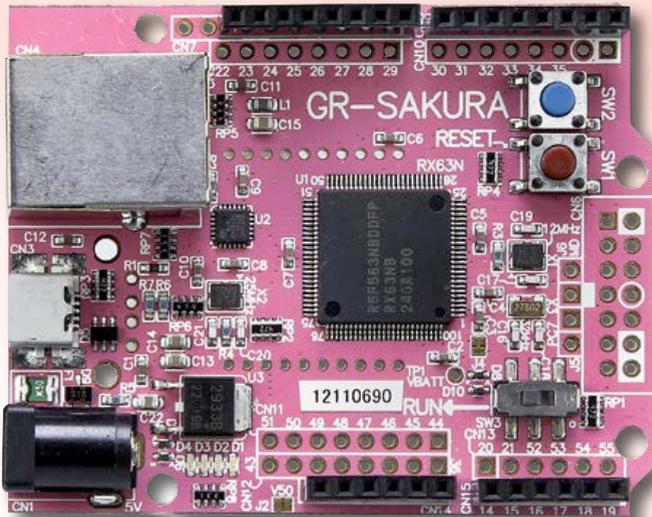
Différence n°3

La fonction hôte USB de la carte *Sakura* est alléchante. Matériellement, c'est un connecteur mini-B mais on peut planter



	Arduino Uno	GR Sakura
tension d'alimentation	5 V pour le processeur	3,3 V pour le processeur 5 V pour la carte
USB	connecteur type B par défaut la carte est alimentée via l'USB	connecteur mini B fonction hôte sur un connecteur type A par défaut la carte est alimentée via l'USB
réseau	aucun	Ethernet RJ45

ARDUINO UNO *contre* GR SAKURA FULL



un connecteur de type A au dos de la carte.

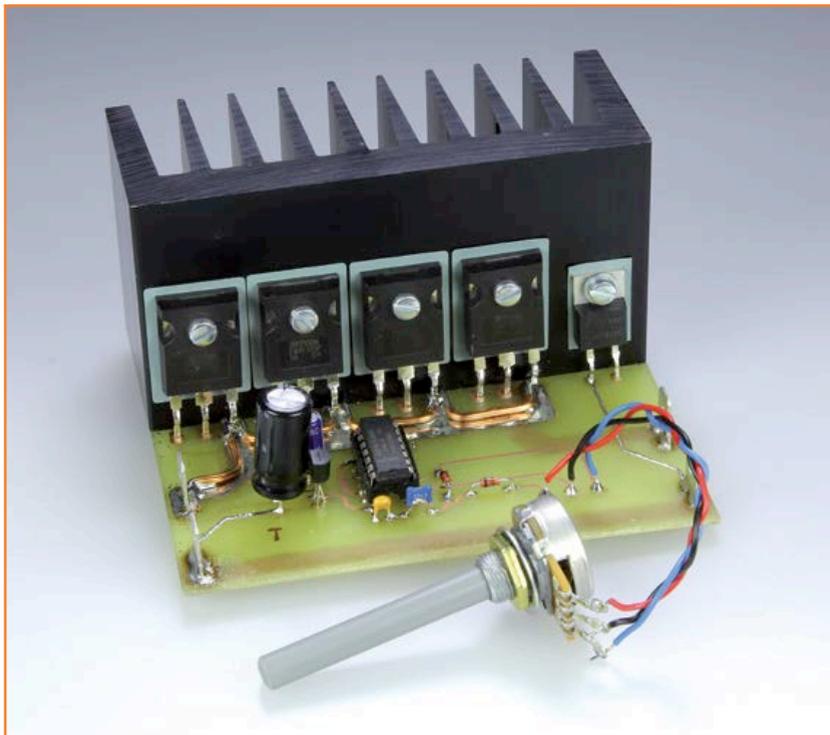
Différence n°4

Plutôt que de son matériel, la vraie puissance d'Arduino vient de la tonne de bibliothèques accessibles facilement (www.arduino.cc). Ne sous-estimons pas cependant les possibilités de programmation de la carte *Sakura* ! Beaucoup de fonctions prennent vie dès lors que vous utilisez le compilateur *dans le nuage*. Reliez la carte à votre PC, manœuvrez les bons boutons – tout est bien documenté – et aussitôt la carte est reconnue comme un périphérique de stockage. Celui-ci contient un lien qui pointe droit vers le site web. Si vous avez un téléphone Android, jetez un œil sur *Gadget Director* : un langage de programmation facile d'accès à base d'icônes. Ça promet.

Références sur www.designspark.com dans la partie *Design Centers*.

(130177 – version française : Kévin PETIT)

	Arduino Uno	GR Sakura
processeur	Atmega328 8 bits 16 MHz	RX63N 32 bits 96 MHz
mémoire	Flash : 32 Ko dont 0,5 Ko pour le chargeur de démarrage SRAM : 2 Ko EEPROM : 1 Ko	Flash : 1 Mo RAM : 128 Ko mémoire Flash données : 32 Ko connecteur MicroSD



Ton Giesberts
(Elektor Labs)

Ce circuit pilote toutes sortes de moteurs CC et supporte jusqu'à 40 A. L'oscillateur sur lequel il repose est un circuit RC rudimentaire construit autour d'un des six inverseurs à trigger de Schmitt (IC1a) d'un 40106. La tension de sortie est maximale lorsque le curseur de P1 est tourné vers D2. Les deux diodes évitent les courts-circuits entre sortie et entrée. Les temps de charge et de décharge sont minimaux aux extrémités de P1. Sur un prototype, nous avons mesuré 1 μ s pour la largeur de l'impulsion négative, et 1,6 μ s pour la positive.

Les deux inverseurs suivants, IC1b et IC1c, nettoient le signal de l'oscillateur et pilotent l'étage tampon composé des trois portes en parallèle IC1d, IC1e et IC1f. Le rôle de R1 est de suspendre l'action des MOSFET en cas d'absence du 40106. La capacité d'entrée totale des quatre MOSFET est de 8 nF, clairement trop grande pour que le tampon charge ou décharge pleinement lorsque P1 est en position extrême. En pratique, cela permet toutefois au pilote de gérer toutes les tensions de sortie (0 à 100 %).

La fréquence de fonctionnement est d'environ 1 kHz (1,07 kHz mesuré sur un prototype). La diode de sortie D3 annule la force contre-électromotrice (f.c.é.m.) induite par les charges inductives comme le sont tous les moteurs CC.

pilote sumo pour moteurs CC

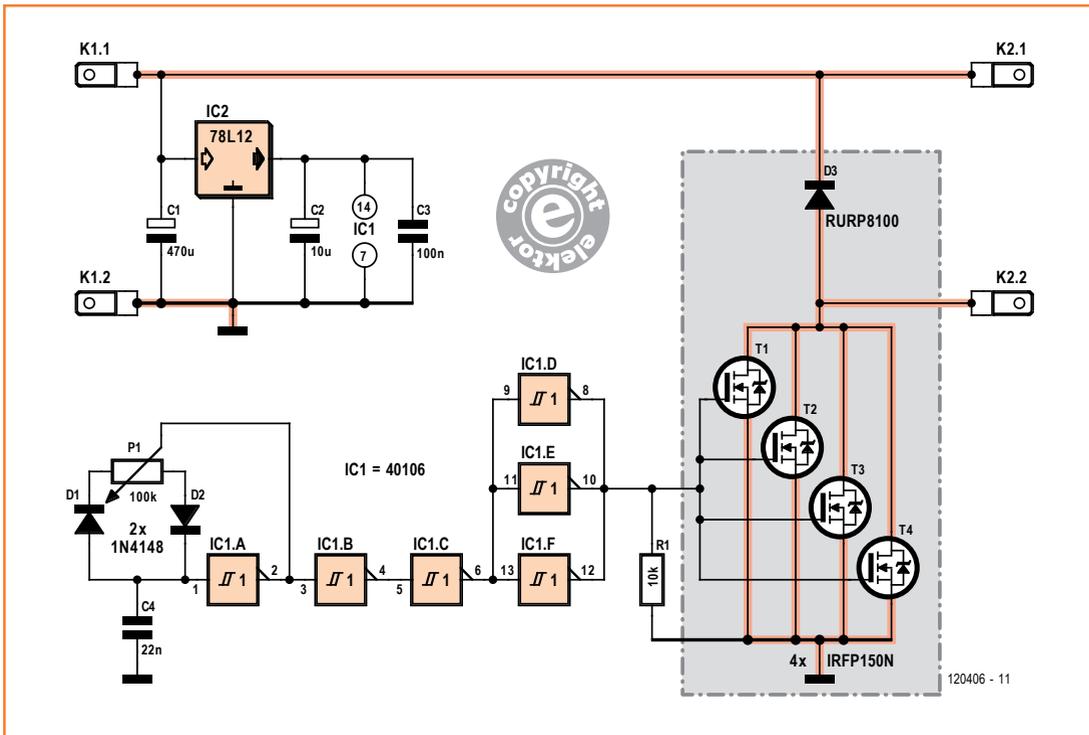
Les courants de sortie intenses et la f.c.é.m. sont problématiques. Sur un premier prototype, les pistes menant à D3 étaient trop étroites et nous en avons grillé une lors d'un essai avec un des moteurs d'Elektor *Wheelie*. À pleine charge, chacun des deux moteurs du *Wheelie* absorbait jusqu'à 20 A sous 24 V. Le circuit a été testé à 40 A et sous 24 V avec une charge résistive.

Telle quelle, notre carte ne supportera pas de telles intensités. Il faut renforcer les pistes par lesquelles passent des courants forts avec du fil de 2,5 mm², ou, plus faciles à souder, deux fils de 1,5 mm² côte à côte. C'est pour cette raison que la carte est dépourvue de réserve de soudure. Sur le schéma, les lignes par lesquelles passent des courants forts sont plus épaisses.

Le 40106 est alimenté par un régulateur de tension 78L12 (IC2) entouré des habituels condensateurs de découplage.

Le potentiomètre de réglage de la vitesse peut être monté hors carte et connecté avec du fil fin. Préférez des vis de 3 mm pour ancrer le dissipateur dans la carte, et veillez à ce qu'il ne touche pas les plots des MOSFET. Déterminez ensuite les positions des vis de montage du transistor et de D3. Coudez légèrement leurs pattes (il existe des outils pour cela) pour réduire les efforts mécaniques qui s'exercent sur elles, puis repérez leurs trous. Taraudez-les pour des vis de 3 mm. N'oubliez pas d'isoler tous les semi-conducteurs du dissipateur. La fréquence de découpage est basse, et vous entendrez probablement gémir le moteur ; ne vous inquiétez pas, ce comportement est normal.

(120406 – version française : Hervé Moreau)



Liste des composants

Résistances

R1 = 10 kΩ, 5 %, 0,25 W
 P1 = pot. lin. 100 kΩ, 20 %, 0,2 W

Condensateurs

C1 = 470 µF 35 V, 20 %, au pas de 3,5 mm
 C2 = 10 µF 25 V, 20 %, au pas de 2 mm
 C3 = 100 nF, 50 V, 20 %, céramique, au pas de 5 mm
 C4 = 22 nF, 100 V, 20 %, céramique, au pas de 5mm

Semi-conducteurs

D1, D2 = 1N4148
 D3 = RURP8100

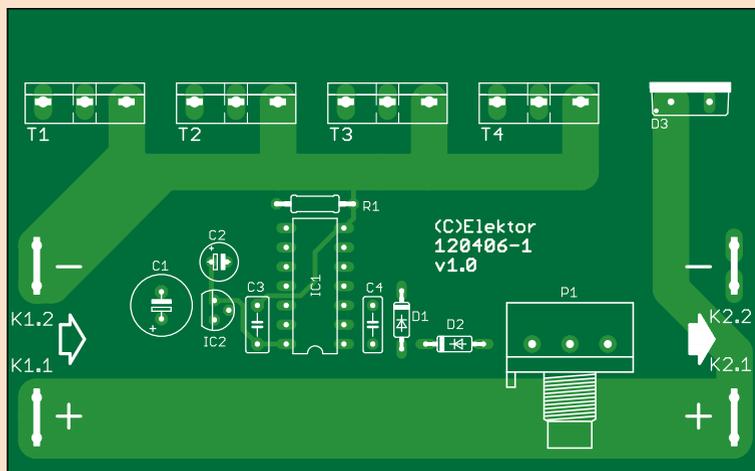
T1-T4 = IRFP150N

IC1 = 40106

IC2 = 78L12

Divers

4 cosses mâles, droites, pour montage sur CI, au pas de 5,1 mm (0,2 pouce)
 dissipateur, 1,9°C/W, 100 x 40 x 50 mm, modèle SK 92/50 SA de Fischer Elektronik
 feuilles thermo-conductrices en élastomère de silicone pour boîtiers TO-3P (T1-T4)
 kit isolant pour boîtier TO-220 ; feuille de mica + entrée de conducteur (D3)
 circuit imprimé 120406-1



convertisseur élévateur MLI Monte là-dessus... !



À partir d'une tension d'entrée comprise entre 8 et 16 V, ce convertisseur élévateur piloté par microcontrôleur délivre une tension de sortie qui peut atteindre 42 V sous 1 A. Cela en fait un chargeur mobile utilisable avec trois accumulateurs de 12 V en série.

la bobine une tension opposée à la tension appliquée précédemment. Cette tension s'ajoute dans notre cas à la tension d'alimentation et provoque la circulation d'un courant à travers la diode vers le condensateur. On peut dire que pendant la phase de conduction l'inductance accumule de l'énergie dans le champ magnétique, situé principalement dans le noyau de ferrite, et que cette énergie est transférée dans le condensateur via la diode pendant la phase de blocage [1].

Wolfgang Schmidt
(Allemagne)

Un convertisseur élévateur (*step-up* ou *boost*) convertit une basse tension d'entrée en une haute tension de sortie. Il est constitué essentiellement d'une inductance, d'un condensateur et d'un commutateur (transistor) alternativement conducteur et bloqué au rythme d'impulsions modulées en largeur (MLI ou *PWM*, *pulse width modulation*). Pendant la phase de conduction du signal MLI, le commutateur est fermé (**fig. 1** en bas). Un cycle de commutation de période T est composé d'une phase haute t_1 et d'une phase basse $T-t_1$. Pendant la phase haute du signal MLI, le commutateur est fermé (fig. 1 en haut). La tension d'entrée U_e est appliquée à l'inductance et un courant I_L , croissant de façon à peu près linéaire, circule à travers la bobine. De ce fait, la quantité d'énergie stockée dans la bobine croît aussi. Quand le commutateur s'ouvre, le champ magnétique de la bobine s'effondre, ce qui induit dans

Le schéma

Le convertisseur élévateur (**fig. 2**) est constitué simplement de L_1 , D_1 , C_8 et du MOSFET T_1 . La fabrication du signal MLI est confiée à un μC (*what else?*) ATmega8-16PU d'Atmel et au micrologiciel qui l'habite. Le signal MLI qui apparaît sur la broche PB1 a une fréquence de 66 kHz puisque nous avons choisi le mode rapide *Fast-PWM*. Comme le μC a pour rôle de réguler la tension de sortie par le biais du rapport cyclique, il doit être informé de la valeur de la dite tension. L'information arrive par R_6 , R_7 et P_2 . L'ajustable est nécessaire parce que la référence de tension interne du μC n'est pas précise. Selon la feuille de caractéristiques, elle se situe entre 2,3 et 3,9 V, mais le calibrage est possible par P_2 . Si la plage de réglage est insuffisante ou si le choix de $R_7 = 43 \text{ k}\Omega$ pose un problème, on peut adapter le logiciel. Pour le réglage, il suffit de connecter un voltmètre en sortie et de comparer l'affichage à celui de l'écran LCD. Le convertisseur analogique-numérique du μC offre une résolution de 10 bits. Le logiciel cal-

cule la tension aux bornes du diviseur de tension 47 kΩ (R7 + P2) et 2,7 kΩ (R6). Il en résulte une résolution de 46 mV ($((49,7 \text{ k}\Omega / 2,7 \text{ k}\Omega) * 2,56 \text{ V}) / 1023$). La valeur affichée à l'écran évolue de ce fait par sauts de 0,004 ou 0,005 V. Les convertisseurs élévateurs ainsi configurés ne comportent aucune limitation d'intensité. Pour éviter une surcharge, le shunt R5 est inséré dans la ligne de sortie et relié à une deuxième entrée A/N du µC. Le logiciel abaisse le rapport cyclique avant que le convertisseur passe en mode discontinu. Pour éviter le bruit sur les deux entrées du CAN, C10, C11 et R8 ont été ajoutés. Le µC commande un afficheur à cristaux liquides qui représente tous les paramètres, les valeurs actuelles de tension et intensité de sortie et – par des menus – les valeurs de consigne. Le circuit comporte trois touches. S1 remet à zéro le µC, S2 et S3 permettent de monter ou descendre, respectivement, la tension de sortie. Une pression sur les deux touches simultanément nous fait passer en mode de limitation d'intensité. On peut alors fixer par les touches une nouvelle valeur maximale du courant de sortie. Peu de temps après la pression sur la dernière touche, l'affichage normal de la tension réapparaît automatiquement. La LED D3 signale la présence d'une tension d'en-

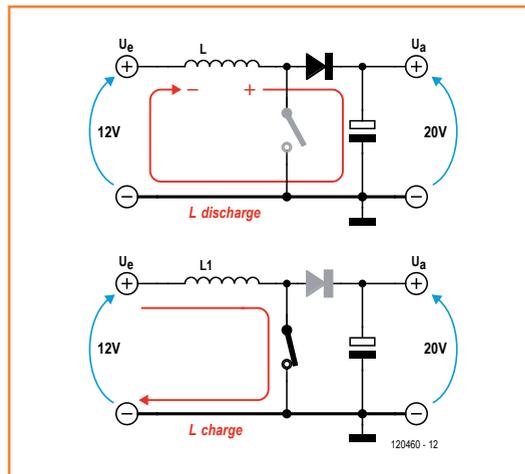


Figure 1. Les deux phases du convertisseur à découpage.

trée. Elle s'allume pour dire que le fusible F1 est coupé et que quelque chose de grave s'est passé dans le circuit. La LED D2 s'allume quand la limitation d'intensité est active.

Construction et installation

Alors que la touche *Reset* est située sur la platine (fig. 3) – elle ne doit servir que rarement –, les touches S2 et S3 sont raccordées par un peu de fil souple aux picots *UP*, *GND* et *DOWN*. Suivant l'installation, on pourra utiliser deux touches tac-

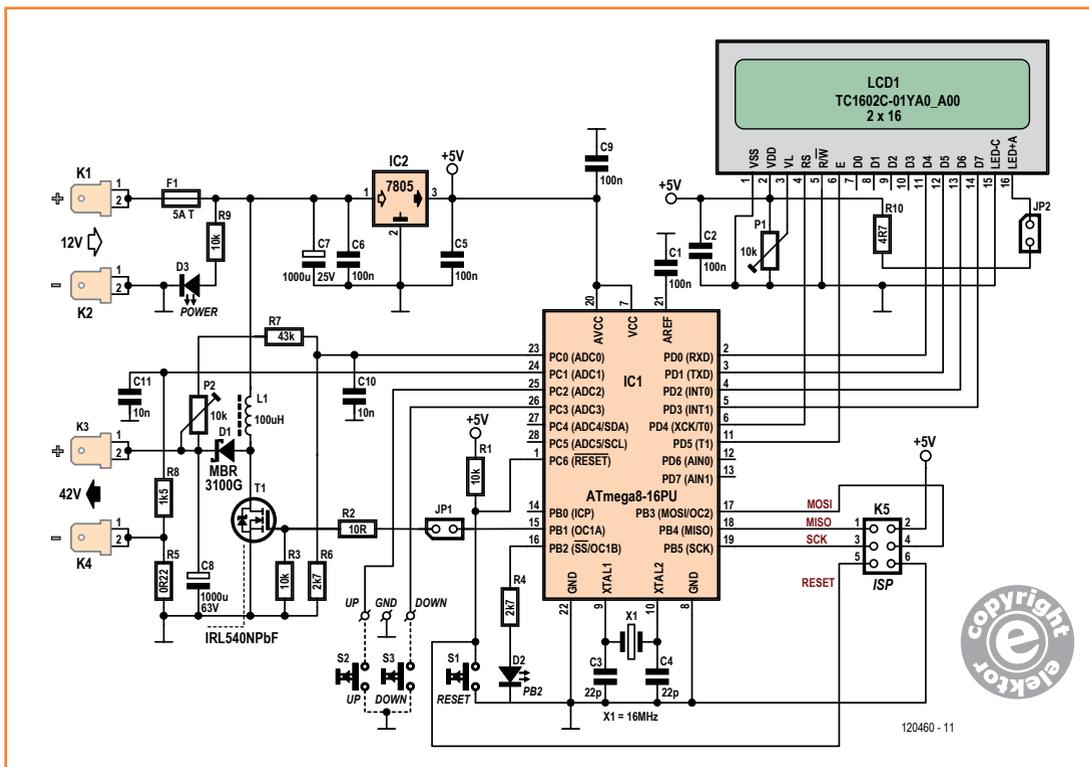


Figure 2. Un convertisseur élévateur à régulation par microcontrôleur.

tiles sur un morceau de platine pastillée ou des modèles pour montage sur châssis directement sur la face avant.

Les deux LED ne sont utiles que bien visibles et non cachées dans le coffret. Le potentiomètre P1 règle le contraste de l'écran à cristaux liquides. Le cavalier JP2 commande le rétro-éclairage de l'écran. On peut monter un interrupteur là aussi. Le refroidissement du MOSFET réclame un petit dissipateur. Une résistance thermique de 21 K/W suffit pour des courants de 1 A au maximum. La diode D1 peut être en boîtier DO201-AD à connexions axiales ou en boîtier TO220. Si vous

optez pour le deuxième type, reportez-vous à la feuille de caractéristiques pour connaître l'orientation.

Les condensateurs utilisés dans les alimentations à découpage méritent une attention particulière. Cela concerne ici C7 et C8, qui sont soumis à la fréquence de découpage élevée de 66 kHz. Des condensateurs ordinaires ne supportent pas une telle fréquence, c'est pourquoi il faut impérativement monter les types indiqués dans la liste de composants.

Lors de tests, nous avons muni l'afficheur LCD d'une barrette mâle droite qui

s'enchâssait dans une barrette femelle correspondante sur la platine. C'était très pratique, mais si on veut monter le convertisseur dans un coffret, il faudra par exemple placer l'écran à l'arrière de la platine. Pour éviter les courts-circuits, interposez alors une feuille isolante solide entre la platine et l'afficheur (j'ai toujours pour cet usage un stock de chutes de carton bakélinisé mince (FR2)).

La programmation du μC passe par l'interface ISP classique accessible par K5. Pendant la programmation, le μC doit être alimenté par le régu-

lateur 7805 de la platine (IC2). Ainsi, la tension présente sur K5 informe l'interface de programmation, par exemple un AVRISP mkII, du niveau de la tension d'alimentation (3,3 ou 5 V).

Pendant la programmation, le niveau des sorties du μC est indéfini. C'est pourquoi le cavalier JP1 dans le circuit de grille du MOSFET est nécessaire. En effet, si la sortie PB1 se trouvait trop longtemps au niveau haut, le MOSFET court-circuiterait la tension d'alimentation. Retirez le cavalier JP1 pendant la programmation ! La résistance R3 tient la grille au niveau de la masse en l'absence de JP1. N'oubliez donc pas de retirer le cavalier pour la programmation ! La résistance R2 élimine l'instabilité du point de commutation due à la forte capacité de grille.

La capacité de grille du MOSFET introduit un temps de commutation appréciable entre les états conducteur et bloqué, dans les deux sens. En conséquence, un MOSFET de puissance s'échauffe notablement, parce que pendant ce temps de commutation, avec la transition relativement lente entre les deux états, un courant d'entrée traverse le MOSFET sous une tension supérieure à zéro. La puissance électrique correspondante est convertie en chaleur. Si on dispose d'un fort courant de commande pour changer l'état de la grille, le temps de commutation est réduit jusqu'à être négligeable. L'ATmega, avec environ 30 mA, n'est qu'une source de courant faible.

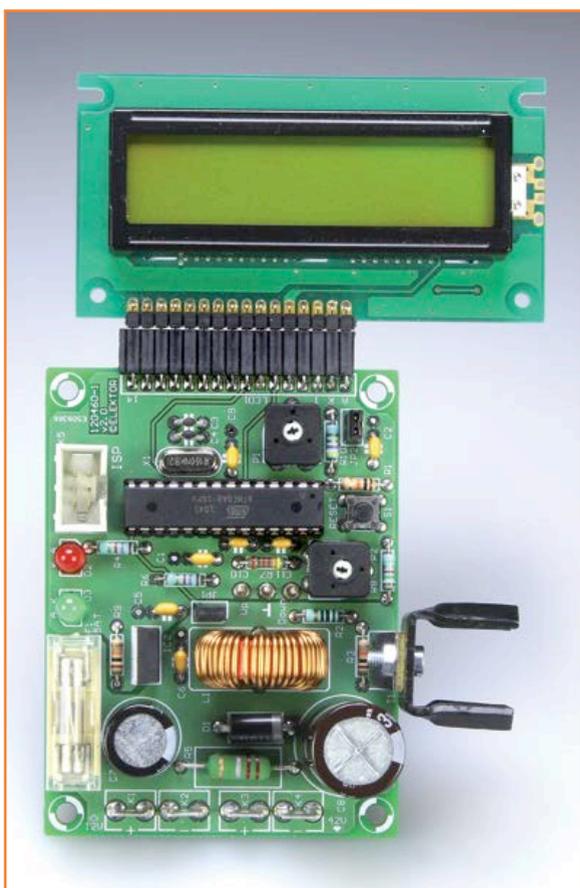
La tension de sortie maximale est déterminée uniquement par la tenue en tension de D1 et T1. Cela donne une première piste d'amélioration du matériel. Ce projet se limite à la mise en œuvre du principe d'un convertisseur élévateur et se veut surtout une incitation.

Susceptible de modification : le logiciel

Le logiciel, dont on trouvera le code source en BASCOM-AVR par le lien [2], représente un régulateur de chargeur très simple, susceptible d'amélioration sur plusieurs points. Il y a parmi eux quelques points critiques aussi bien que la mise en œuvre d'un « vrai » chargeur d'accus au plomb avec différentes phases de charge.

À propos du courant d'entrée : il est jusqu'à trois fois et demie plus fort que le courant de sortie, d'où le fusible de temporisé 5 A.

Testé sur deux types différents d'accumulateurs au plomb à électrolyte gélifié, le chargeur a montré un comportement remarquable. Lors du réglage du courant maximal à la valeur relativement élevée de 0,2 C, la tension de la batterie



Liste des composants

Résistances

R1,R3,R9 = 10 kΩ, 5 %, 250 mW
 R2 = 10 Ω, 5 %, 250 mW
 R4,R6 = 2,7 kΩ, 5 %, 250 mW
 R5 = 0Ω22, 5 %, 1 W
 R7 = 43 kΩ, 1 %, 600 mW
 R8 = 1,5 kΩ, 5 %, 250 mW
 R10 = 4Ω7, 5 %, 250 mW
 P1,P2 = 10 kΩ, 20 %, 0W15, aj.
 à plat

Condensateurs

C1,C2,C5,C6,C9 = 100 nF, 63 V, 5 %, RM5 ou RM7,5, cér.
 C3,C4 = 22 pF, 50 V, 5 %, RM5
 C7 = 1000 µF, 25 V, 20 %, radial, diam. 12,5 mm, RM5 (Panasonic EEUTP1E102 Farnell 1890543)
 C8 = 1000 µF, 63 V, 20 %, radial, diam. 16 mm, RM7,5 (Nichicon UPW1J102MHD Farnell 2112865)
 C10,C11 = 10 nF, 100 V, 10 %, RM5, cér.

Inductances

L1 = 100 µH, 5 A, 20 %, radial 25 mm, RM8 (Würth Elektronik 7447070 chez Farnell 2082537)

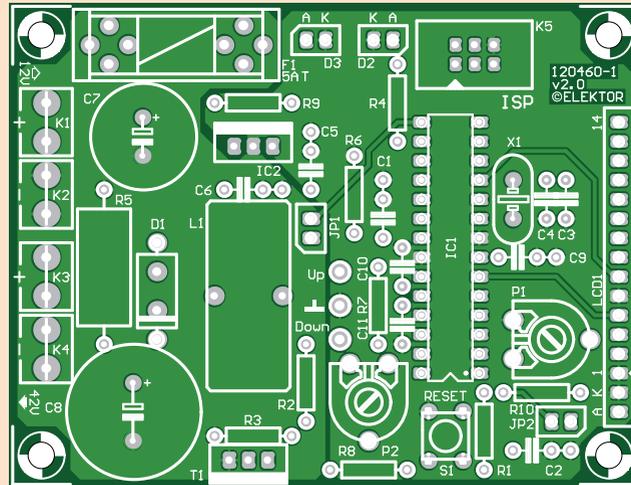


Figure 3. Le circuit imprimé du convertisseur élévateur.

Semi-conducteurs

D1 = MBR3100G
 D2 = LED rouge, 3 mm
 D3 = LED verte, 3 mm
 T1 = IRL540NPbF
 IC1 = ATmega8-16PU
 (programmé : 120460-41)
 IC2 = 7805

Divers

F1 = porte-fusible, 20x5 mm avec capot
 F1 = fusible 5 A (retardé)
 JP1,JP2 = barrette SIL 2 points, pas 2,54 mm, avec cavalier
 K1 à K4 = languette fast-on 5 mm pour circuit imprimé
 K5 = barrette droite double rangée, 2x3 points, pas 2,54 mm
 S1 = touche 6x6 mm 1 pôle NO
 S2,S3 = touche 1 pôle NO, pour CI ou châssis*
 PC1...PC3 = picot 1,3 mm (pour S2,S3)
 radiateur FK230SAL1 Fischer
 X1 = 16 MHz, HC49/US, 50 ppm, Cload 18 pF

LCD1 = LCD 2x16 (Elektor 120061-71)
 circuit imprimé 120460-1

est tombée brusquement, après que la tension maximale ait été atteinte et que le chargeur se soit arrêté. En pratique, il faut observer le processus de charge et savoir exactement quand il convient de l'arrêter manuellement.

Le logiciel n'offre ni interface confortable ni commandes de charges multiples. Il montre plutôt la faisabilité d'un principe de charge et ouvre au lecteur curieux un vaste champ d'expérimentation, grâce aux descriptions complètes.

Suggestion : les accus au plomb (électrolyte gélifié) doivent être chargés en deux à quatre phases [3]. Un accu partiellement déchargé se charge au courant maximal (0,1 à 0,2 C) pendant la phase de charge intensive (*bulk charge*), jusqu'à atteindre une tension par élément de 2,4 V (le logiciel présenté ici se limite à cela ; arrivée à ce point, la batterie est chargée à 80%). Ensuite on abaisse la tension de fin de charge en observant le courant de charge jusqu'à ce qu'il tombe au dixième de la valeur maximale. Cette deuxième phase, dite d'absorption, charge l'accu jusqu'à environ 98 %. Le petit reste est l'affaire de la phase de *floating* : la consigne de tension de sortie est réduite à 2,23 V par élément. Dans cette phase – *floating, standby*, attente en parallèle –,

l'accu peut rester connecté au chargeur indéfiniment sans dégagement gazeux.

Si vous omettez la quatrième phase, vous courez le risque de tomber en décharge profonde (en dessous de 1,75 V par élément). Il faut alors retaper la batterie avec un courant réduit jusqu'à ce qu'elle remonte à cette valeur minimale.

Si vous travaillez à de telles améliorations du logiciel et arrivez à un résultat intéressant, ou si y travaillez, mais n'arrivez à rien, visitez le site du projet [4].

(120460 – version française : Jean-Paul Brodier)

Liens

- [1] Principes des convertisseurs à découpage
http://schmidt-walter.eit.h-da.de/smpps_e/smpps_e.html
 ou <http://goo.gl/FPJ5O>
- [2] www.elektor.fr/120460
- [3] Charge des accus au plomb :
www.batterystuff.com/kb/articles/battery-articles/battery-basics.html#9
 ou <http://goo.gl/4c5wS>
- [4] www.elektor-projects.com

limiteur de courant d'appel X-treme

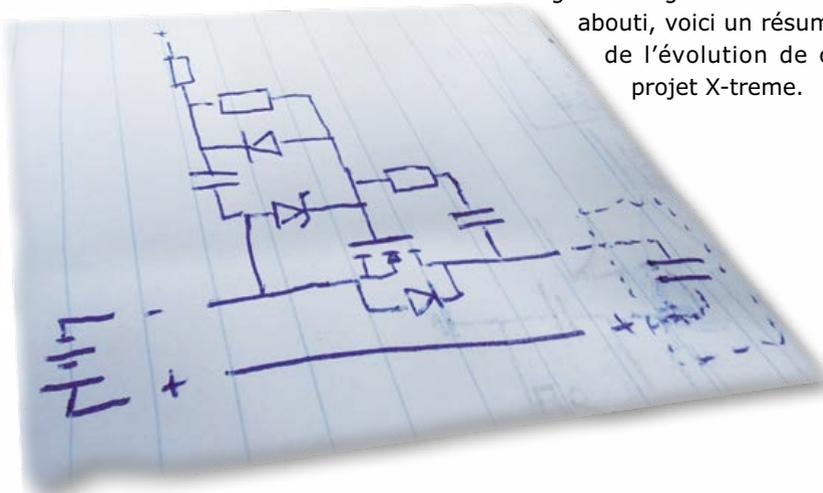
l'ampère fort tranquille



Raymond Vermeulen
(Elektor)

Notre équipe compte un fou. Un fou d'aéromodélisme, qui ne fait voler que des modèles puissants. C'est pour l'aider à braver les lois de la physique que ce projet tout analogique a été conçu. Notre désespéré voyait en effet les connecteurs d'alimentation de son avion s'autodétruire lorsque les accus étaient connectés au moteur. Le coupable ? L'arc dû à l'irruption de courant. Pas de la vulgaire étincelle de meule : avec des connecteurs de 6 mm plaqués or, ça fait cher l'étincelle. Aucun doute, pour éviter le feu d'artifice, il faut réguler le courant de démarrage, mais ce genre de limiteur on ne le sort pas de sa manche. Il m'a fallu plancher dessus un moment. De mes premiers gribouillages au modèle abouti, voici un résumé de l'évolution de ce projet X-treme.

Figure 1.
Esquisse à la Bob Pease d'une idée de limiteur de courant d'appel.



La note d'application AN1542 de Motorola [1] m'a servi de piste d'envol. De sa studieuse lecture, j'ai tiré l'ébauche (fig. 1) d'un limiteur de courant d'appel, conçu pour une batterie de 37 V et une charge de 200 A en fonctionnement normal. Avec deux MOSFET en parallèle, j'obtenais une faible RDS(on) globale. Une modélisation LTspice m'a ensuite fait comprendre que le problème ne venait pas des ampères mais de la capacité de charge provoquant le courant d'appel. Mon circuit était donc conçu pour le pire des scénarios. Il restait à m'assurer que les MOSFET travaillaient dans leur zone de sécurité de fonctionnement. Avec un petit pilote de moteur sans balai triphasé de 10 A, j'ai mesuré une capacité d'entrée de « seulement » 120 μF . Avec un contrôleur pouvant délivrer 120 A, j'ai trouvé 13800 μF (13,8 mF) pour une résistance série équivalente d'environ 2,7 Ω .

Pour la réalisation du circuit, j'ai retenu le MOSFET IPB017N06N3 d'Infineon, pour sa faible résistance à l'état passant RDS(on) de 1,7 m Ω , son prix et sa disponibilité. Restait la question du nombre de MOSFET.

La modélisation LTspice avec le IPB017N06N3 a montré que certaines valeurs de composants devaient être modifiées. J'ai aussi dégotté un

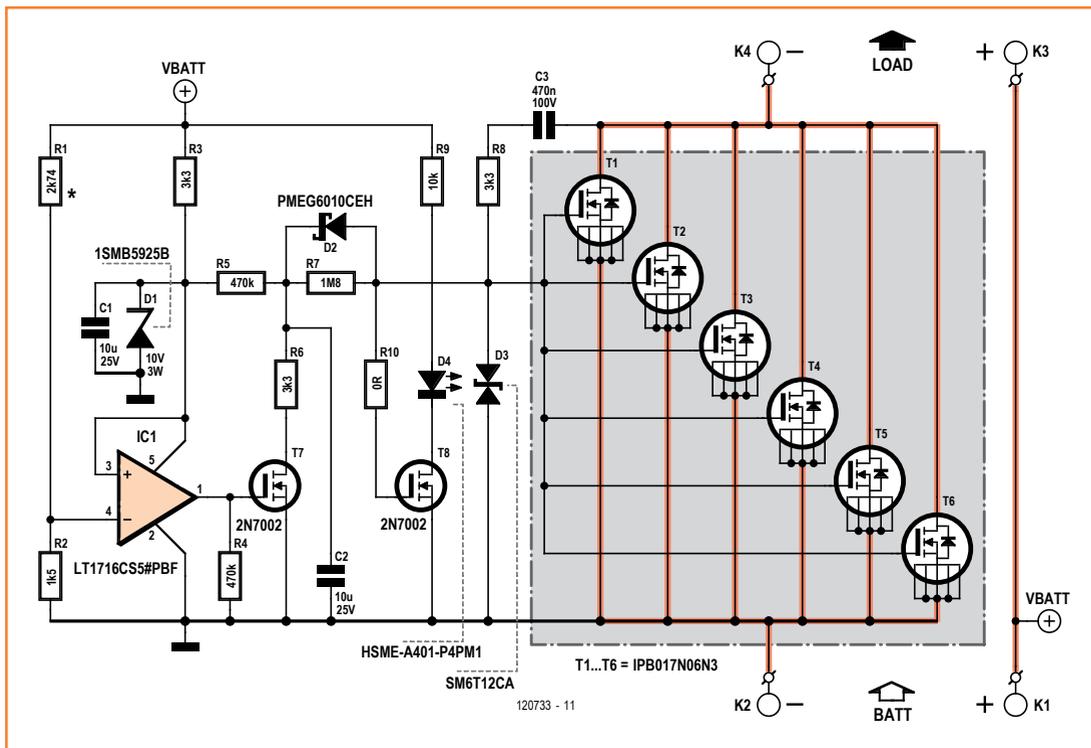


Figure 2. Schéma du limiteur de courant d'appel X-treme. Les pistes « courant fort » sont plus épaisses et surlignées.

radiateur bon marché, de dimensions standard et pré-percé.

Mon schéma (fig. 2) diffère sensiblement du circuit de la note AN1542. Motorola donne au courant une forme d'onde rectangulaire qui produit une irruption de courant et un pic de tension ne s'atténuant que lentement. Avec mon circuit, le courant n'augmente au contraire que progressivement. Sa forme d'onde est en dents de scie, et il en résulte une courbe de dissipation de puissance (PFET) qui prend l'aspect d'une parabole inversée. La figure 3 montre les formes d'onde de base, qui par ailleurs respectent la zone de sécurité des MOSFET bien mieux qu'AN1542. La diode d'écrtage D3 protège les MOSFET des inversions de polarité accidentelles.

Nous avons testé un premier prototype avec un condensateur de 15000 µF (15 mF), avec et sans charge résistive, en reliant le circuit X-Treme à une alimentation de 40 V.

Tout semblait fonctionner comme attendu, bien que sans charge résistive la protection contre les sous-tensions n'ait pas opéré sur le front descendant.

Comme test final, le circuit a été utilisé avec une commande de moteur CC sans balai de 10 kW,

Tableau 1. Relation entre valeur de déclenchement et tension d'alimentation

type d'accu (lithium)	tension accu	V _{décl.}	R1
4S	12 – 16,8 V	11 V	180 Ω
5S	15 – 21 V	14,2 V	620 Ω
6S	18 – 25,2 V	16,8 V	1 kΩ
8S	24 – 33,6 V	22,9 V	1,87 kΩ
10S	30 – 42 V	28,9 V	2,74 kΩ
11S	33 – 46,2 V	31,8 V	3,16 kΩ
12S	36 – 50,4 V	34,9 V	3,6 kΩ

non chargé, absorbant 8,5 A à vitesse continue et de 20 à 30 A en régime de régulation. Nous avons effectué plusieurs démarrages « à froid » sous 37 et 48 V. Les câbles et les connecteurs ont chauffé, mais pas les MOSFET ni le reste du circuit. Nous n'avons pas entendu les habituels coups de butoir caractéristiques des fortes intensités, et aucun condensateur n'a explosé.

Le départ était donc donné pour la conception et la production d'une carte à une seule face (!) mélangeant composants CMS et traversants (fig. 4). La valeur de R1 fixe la tension de déclenchement et dépend donc de l'alimentation (tableau

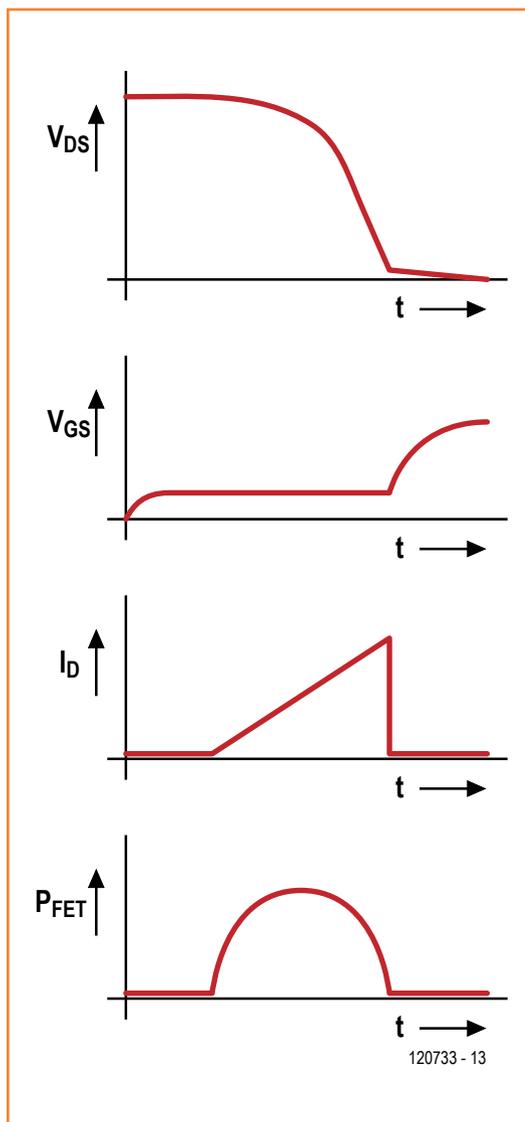


Figure 3. Avec notre circuit, la courbe de dissipation des MOSFET prend la forme d'une parabole inversée.

1). Notez que le circuit ne doit pas être utilisé avec des alimentations inférieures à 12 V. Ce cas est heureusement rare dans les applications qui utilisent des moteurs CC sans balai de forte puissance – vous devinez la raison.

Qu'ils soient « de surtension » ou « continus », des courants très forts peuvent traverser la carte. Les pistes BATT- et BATT+ ainsi que celles menant aux sources et drains des MOSFET doivent donc être renforcées sur toute leur longueur avec du fil de cuivre de 2,5 mm², de préférence avec 2 fils côte à côte. La plupart de cette tuyauterie est à poser dans la zone qui sera couverte par le dissipateur. Le fil de 1,5 mm² convient, mais dans ce cas placez 3 ou 4 fils côte à côte. Soyez généreux avec la soudure le long des pistes et des fils de cuivre. Si votre carte est pourvue d'une réserve de soudure, grattez doucement le masque pour mettre le cuivre à nu (sans le blesser !). Ensuite étamez puis posez les fils de renfort.

Les connecteurs d'alimentation et de charge, K1-K2 et K3-K4, doivent posséder des bornes de grande qualité, si possible plaquées or. Peu importe qu'elles soient rondes ou plates (cosses « FastOn ») pourvu que vous les soudiez directement sur les pistes de la carte. Car souvenez-vous : chaque milliohm compte dans ce circuit. Qui veut perdre de la puissance ou du couple au moment du décollage ? Pour prévenir toute inversion de polarité, vous ajoutez des connecteurs femelle et mâle aux lignes + et - de l'alimentation. Idem pour les lignes de sortie + et -. Les MOSFET reposent sur la carte. Le dissipateur

Liste des composants

Résistances

(toutes 0,25 W, 1%, CMS 1206)

R1 = 2,74 kΩ *

R2 = 1,5 kΩ

R3, R6, R8 = 3,3 kΩ

R4, R5 = 470 kΩ

R7 = 1,8 MΩ

R9 = 10 kΩ

R10 = 0 Ω

Condensateurs

C1, C2 = 10 μF 10 % 25 V, X5R, 1206

C3 = 470 nF 10 % 100 V, X7R, 1206

Semi-conducteurs

D1 = 1SMB5925B diode zener, SMB (Newark/Farnell 1894811)

D2 = PMEG6010CEH, diode Schottky, NXP, SOD-123F (Newark/Farnell 1510694)

D3 = SM6T12CA, diode d'écrêtage (TVS), STmicroElectronics, SMB (Newark/Farnell 9885870)

D4 = HSME-A401-P4PM1, LED, verte, Avago, PLCC-4 (Newark/Farnell 1058419)

IC1 = LT1716CS5#PBF, comparateur, Linear Technology, SOT-23-5 (Newark/Farnell1417738)

T1, T2, T3, T4, T5, T6 = IPB017N06N3, MOSFET type N, Infineon, TO-263-7 (Newark/Farnell 1775519)

T7, T8 = 2N7002, MOSFET type N, Diodes Inc., SOT-23 (Newark/Farnell 1713823)

Divers

K1 à K4 = connecteurs pour courants forts, mâles & femelles, plaqués or *

dissipateur, format ½ brique, modèle 241204B92200G de Aavid Thermalloy, 60,96 mm x 57,91 mm x 11,4 mm (Newark/Farnell 1703176)

circuit imprimé 120733-1

* composant librement configurable, voir texte

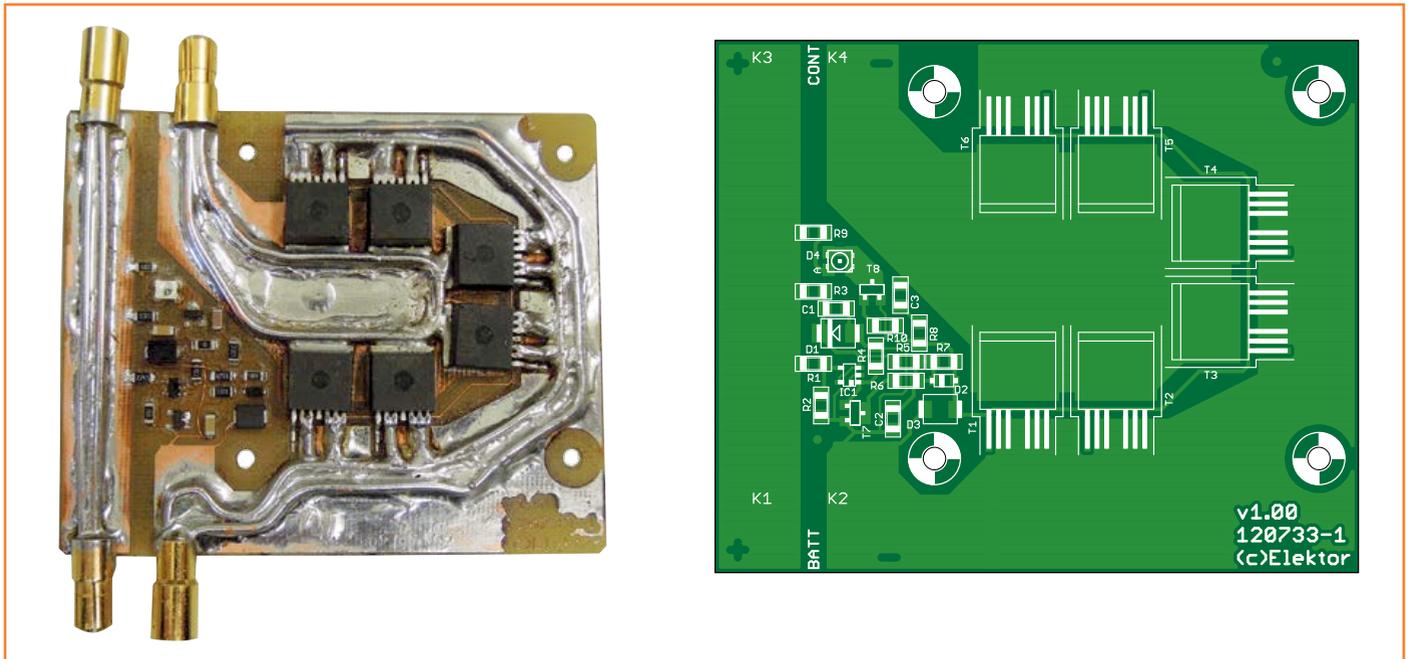


Figure 4. La carte est compacte et a été conçue pour que le dissipateur recouvre les MOSFET. En pratique, les pistes de cuivre qui transportent le courant de charge doivent toutes être renforcées avec des fils de cuivre.

qui les recouvre est composé de feuilles thermo-conductrices. Il se fixe à l'aide de quatre vis ou boulons M3. Deux écrous M3 par boulon servent d'entretoise entre la surface de la carte et le côté plat du dissipateur. La hauteur totale de séparation est d'environ 5 mm. Ne serrez que légèrement les boulons pour ne pas trop comprimer les feuilles thermo-conductrices.

Bien que nous ayons surtout évoqué les contrôleurs de moteurs utilisés en modélisme, le circuit est adapté à toute charge CC de 12 V à 40 V à très faible résistance initiale, y compris aux gros condensateurs électrolytiques de filtrage et aux filaments de lampes.

(120733 – version française : Hervé Moreau)



- [1] AN1542: www.bonavolta.ch/hobby/files/MotorolaAN1542.pdf
- [2] fiche technique du IPB017N06N3 : <http://goo.gl/Xd43I>
ou www.infineon.com/dgdl/IPB017N06N3_Rev2.2.pdf?folderId=db3a30431441fb5d01148ca9f1be0e77&fileId=db3a30431ddc9372011e264a7ab746ea

Elektor USB Power Pack : chargeur de secours pour batterie de téléphone

Ton Giesberts
(Elektor)

Quand on a un besoin, au milieu de nulle part et loin de toute prise de courant, de recharger d'urgence son téléphone ou sa tablette tactile, un modeste jeu de batteries AA NiMH chargées peut rendre d'immenses services. Associées à un chargeur comme celui que nous proposons, elles peuvent revigorer la batterie à plat d'un téléphone ou d'une tablette à travers son connecteur USB.



Ce pack de puissance est donc composé d'un jeu de batteries NiMH courantes et d'un chargeur.

Combien de batteries ?

Le circuit doit produire 5 V et fournir un courant d'1 A. La tension cumulée de quatre batteries NiMH bien chargées dépasse allègrement les 5 V. Il serait donc prudent de se contenter de trois. Néanmoins, dans le jargon USB, « 5 V » est une valeur nominale qui en pratique varie de 4,35 V à 5,40 V. Ceci pourrait plaider pour une quatrième batterie, mais nous tenons à produire précisément 5 V, car certains circuits utilisent la tension de l'USB comme référence. Ce sera donc trois. Accessoirement, comme la tension d'une seule

batterie Li-ion est grosso modo équivalente à celle de trois batteries NiMH en série, il est envisageable de remplacer les batteries NiMH par une batterie Li-ion.

Convertisseur Boost TPS61030

Avec trois batteries le coupleur est plus petit, mais il faut un convertisseur élévateur (*boost*), comme l'excellent élévateur synchrone TPS61030 de Texas Instruments, capable de commuter 4 A et arborant un rendement de 96 % (qui bien entendu dépend de la tension d'entrée et du courant de sortie). Ce convertisseur est aussi doté d'un comparateur (*Low Battery Comparator*) optionnel pour empêcher la décharge profonde. Une sécurité contre les sous-tensions (1,6 V) lui permet d'éviter de fonctionner de manière erratique dans ce cas. La référence de tension interne de 0,5 V facilite le calcul du diviseur réglant la tension de sortie. Ici R3 vaut 1,8 M Ω et R4 200 k Ω . D'après la feuille de caractéristiques, un condensateur de stabilisation supplémentaire, en parallèle avec R3, est nécessaire si R4 est significativement inférieure à 200 k Ω . Pour être tranquille, j'ai utilisé une capa de 10 pF.

La résistance R2 doit être suffisamment petite pour éliminer le courant d'entrée du comparateur (environ 10 nA). Une valeur de 500 k Ω est recommandée. La tension du comparateur est d'environ 500 mV avec une hystérésis de 10 mV. J'ai choisi un seuil de 3 x 1,1 V pour définir une batterie vide. En prenant 1,8 M Ω pour R1 et 330 k Ω pour R2, le seuil théorique se situe à

3,23 V. Lorsque la tension totale de la batterie descend sous ce seuil, la sortie du comparateur (LBO), utilisée pour désactiver le circuit de sortie, passe au niveau bas.

La tension d'entrée est découplée à l'aide C1 et C2, conformément aux recommandations de la feuille de caractéristiques. Le découplage de la tension de sortie dépend de l'ondulation maximale. L'idéal serait quelques millivolts, mais l'ESR des condensateurs ainsi que le routage de la carte entraînent une valeur plus élevée en pratique. Théoriquement, l'ondulation devrait être d'environ 1 mV avec un condensateur réservoir de 220 µF. En pratique, j'ai mesuré environ 60 mV aux bornes de C5 (3,50 V en entrée, charge de 1 A). L'ESR nominale de C5 est de 20 mΩ à 100 kHz. La fréquence de commutation de 600 kHz, bien au-delà, est responsable des fortes ondulations. Afin de s'en débarrasser, la perle de ferrite L2 a été placée en série dans le circuit de sortie ; le condensateur C8 figrole le travail.

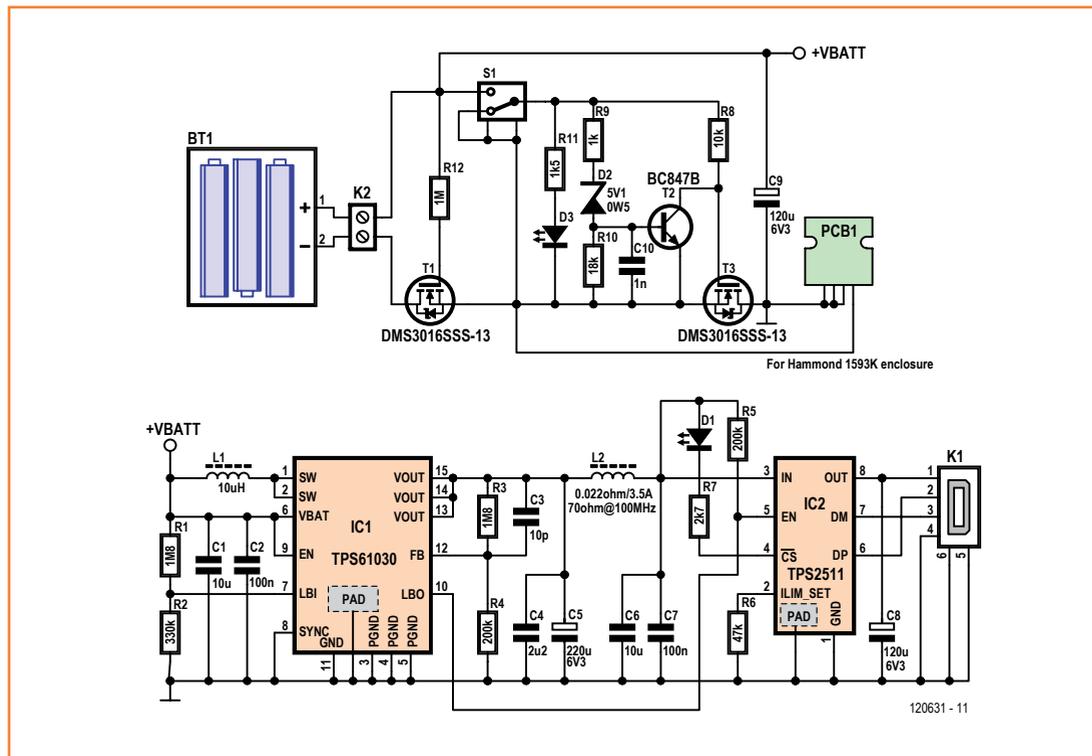
Pour le calcul de la valeur de L1, nous avons pris en compte un changement de 10 % du courant maximal moyen dans l'inductance. À 3,20 V, celui-ci est proche de 2 A. La formule de la feuille de caractéristiques (SLUS534E) donne dans ce cas une valeur d'environ 10 µH.

La broche Sync fixe le mode de fonctionnement

Mesures et spécifications

Tension d'entrée	3,3 – 4,1 V
Courant d'entrée maximal	1,7 A ($V_{in} = 3,33$ V)
Tension de sortie	4,93 V (à vide)
	4,92 V (charge de 0,3 A)
	4,82 V (charge de 1 A)
Seuil de batterie faible	3,25 V
Protection contre les surtensions	4,30 V
Rendement	95 % ($3,52 V_{in}$; 0,3 A en sortie)
	90 % ($3,52 V_{in}$; 1 A en sortie)
Courant consommé (à vide)	4,5 mA ($V_{in} = 3,6$ V)
La LED de mise sous tension s'allume à 1,6 V	
Pertes mesurées pour un courant de sortie de 1 A :	
Aux bornes de T1 et T3 (chacun)	23 mV (1,7 A en entrée)
De L2 au connecteur USB (PCB)	85,3 mV
Aux bornes d'IC2	80 mV
Connecteur USB (chaque patte)	13 mV

du convertisseur. Nous l'avons mise à la masse pour le mode *Économie d'énergie* qui améliore l'efficacité aux faibles charges. Le convertisseur ne fonctionne alors que lorsque la tension de sortie descend au-dessous d'un certain seuil. Côté inconvénients, l'ondulation en sortie augmente légèrement. À vide, nous avons observé un signal en dent de scie de 80 mV d'amplitude et 150 ms



de période. Mais les choses s'améliorent rapidement lorsque l'on augmente la charge.

Un TPS2511 pour l'USB

Texas Instruments appelle *USB Dedicated Charging Port Controller and Current Limiting Power Switch* le TPS2511 utilisé pour commander la sortie, mais cela ne nous a pas fait fuir. Souvent, il ne suffit pas d'appliquer 5 V sur un connecteur USB pour que l'appareil alimenté fonctionne. Que les fabricants utilisent de plus en plus un connecteur USB pour la recharge est une bonne nouvelle, mais ce n'est pas demain la veille que les char-

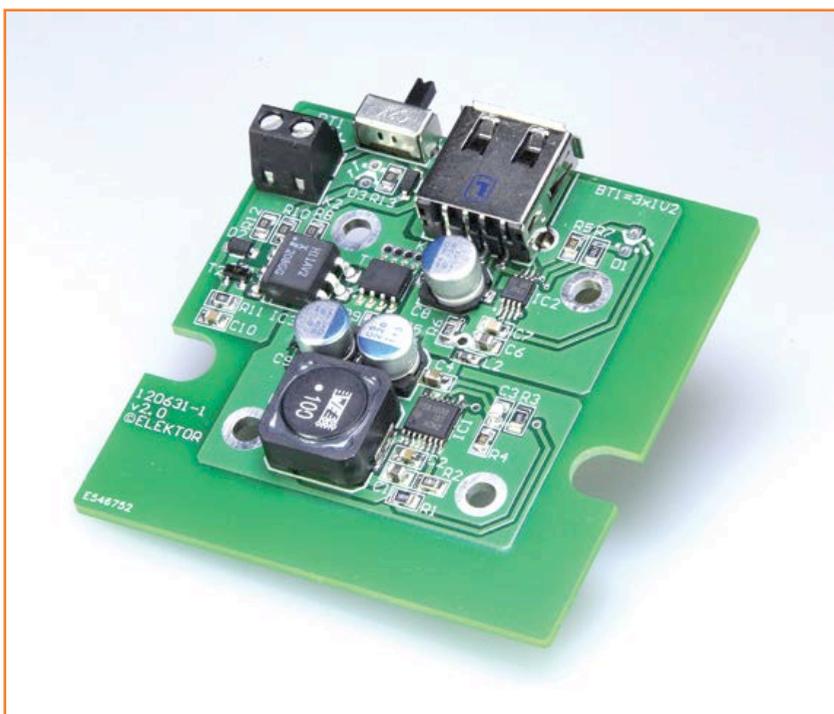
(*SLUSB18*) donne une liste exhaustive des fonctions.

Le TPS61030 fournit jusqu'à 2 A sous une tension de batterie de 3,3 V, et le TPS2511 s'en accommoderait. Mais avec 1 A en sortie et une tension d'entrée de 3,33 V, les batteries doivent déjà fournir 1,7 A. Avec 2 A en sortie, ce courant serait plus que doublé à cause des pertes plus fortes. Aussi, la capacité des batteries décroît lorsqu'on leur demande un fort courant. C'est pour cela que nous avons choisi de limiter le TPS2511 à 5 W. Sa broche DP a été reliée à la broche D- du connecteur USB, DM est reliée à D+. La limite de courant a été réglée légèrement au-dessus du nécessaire avec $R6 = 47 \text{ k}\Omega$ afin d'éviter que la tension de sortie ne soit limitée prématurément.

La broche *Current Sensing Report* n'est pas utilisée pour sa fonction première. Plutôt que de compenser les pertes de tension en changeant la rétroaction du convertisseur (pas vraiment nécessaire avec un courant max de 1 A), cette broche pilote la LED D1. Lorsque D1 s'allume, parcourue par un peu plus de 1 mA, vous savez que la charge consomme plus que la moitié du courant maximal. Nous avons vu que la sortie du comparateur du convertisseur pilote l'entrée d'activation EN du TPS2511 afin que sa sortie soit coupée lorsque les batteries sont à plat. R5 est nécessaire parce que, lorsque la sortie du comparateur est inactive, elle est à l'état haute impédance.

Protections

La connexion de l'ensemble de batteries au circuit imprimé se fait via un bornier à vis au pas de 3,8 mm. Une inversion accidentelle de polarité ne peut être exclue. Afin de prévenir d'éventuels dommages au circuit dans ce cas tout en réduisant au minimum les pertes lorsqu'elles sont reliées correctement, un petit MOSFET (T1) à canal N a été utilisé à l'envers. Lorsque l'on relie les batteries dans le bon sens, sa diode de corps conduit, le MOSFET est passant et sa tension grille-source, aux bornes de R12, est positive. Qu'un courant circule de la source vers le drain n'est pas un problème. Si les batteries sont reliées dans le mauvais sens, la tension de grille est négative, le MOSFET est bloqué et la diode de corps bloque la tension de batterie. La tension maximale supportée par la grille du MOSFET est de 12 V, le circuit ne survivra donc pas à plus. Lorsque le courant d'entrée est de 1,7 A, la chute



geurs seront compatibles entre les plateformes au parfum de fruits et les autres. Certains appareils attendent p. ex. une tension spécifique sur les lignes de données, ou tout simplement qu'elles soient reliées (résistance) pour reconnaître un chargeur (*Dedicated Charger Port*).

Le TPS2511 prend en charge trois des protocoles les plus courants :

- USB Battery Charging Specification, Revision 1.2 (BC1.2);
- Chinese Telecommunications Industry Standard YD/T 1591-2009;
- Divider Mode.

La feuille de caractéristiques du TPS2511

de tension dans le MOSFET n'est que de 23 mV (mesuré sur un prototype).

Pour éviter l'usage d'un commutateur de puissance, plutôt cher, la protection contre les surtensions est combinée avec un petit interrupteur bon marché. Le fonctionnement de cette protection est très simple. Lorsque la tension d'alimentation est trop élevée, la zener D2 entraîne la conduction du transistor NPN T2 qui à son tour interrompt la tension de grille du MOSFET T3. La zener 5,1 V conduit déjà avant sa tension nominale. Lorsque la tension de batterie est de 3,60 V, le courant à travers D2 est d'environ 12 μ A. À 4,25 V, c'est plus de 30 μ A. La mesure peut se faire facilement aux bornes de R9, qui empêche le courant à travers la diode zener de faire boue de neige lorsque la tension d'entrée dépasse environ 5,70 V. Si jamais la protection contre les surtensions se déclenche prématurément (à cause de la tolérance sur la diode zener), il suffit d'adapter R10 : plus sa valeur est faible et plus le seuil est élevé. Cette protection contre les surtensions est prévue pour le cas où on brancherait un bloc d'alimentation AC (j'espère réglé sur moins de 12 V) ou une pile de 9 V. Le TPS61030 est capable de supporter 7,00 V (maximum absolu, 5,50 V recommandés). L'inconvénient d'un convertisseur élévateur est que sa tension de sortie augmente lorsque la tension d'entrée dépasse la tension de régulation (ici 5,00 V nominal).

Assemblage

Le circuit a été dessiné pour un boîtier *Hammond*. Il est bon marché et facile à adapter à votre application. Il sera fixé à l'aide de quatre vis autotaraudeuses, et les moitiés supérieure et inférieure du boîtier avec deux plus longues. L'avant et l'arrière sont deux panneaux séparés. Il faudra percer trois trous dans l'un des deux. Les trous pour le connecteur USB et l'interrupteur devront être alignés avec ces composants. Idem pour le passage des LED qu'il faudra ménager dans la moitié supérieure du boîtier. Il est aussi possible d'utiliser un jack, ce n'est pas la place qui manque, mais gare à la résistance de contact, l'efficacité du montage s'en trouverait diminuée. Les trous de fixation du circuit imprimé servent aussi à relier les plans d'alimentation des deux faces. Vous remarquerez que le trou à proximité du bornier à vis K2 n'est pas relié à la masse, mais au signal entre T1 et T3. Le relier à la masse ne créerait aucun dommage, mais cela déclencherait le circuit. Les trois autres trous seront reliés

Liste des composants

Résistances

(0805, 125 mW)
 R1,R3 = 1,8 M Ω 1%
 R2 = 330 k Ω 1%
 R4,R5 = 200 k Ω 1%
 R6 = 47 k Ω , 1%
 R7 = 2,7 k Ω , 5%
 R8 = 10 k Ω , 5%
 R9 = 1 k Ω , 5%
 R10 = 18 k Ω , 5%
 R11 = 1,5 k Ω , 5%
 R12 = 1 M Ω , 5%

Condensateurs

C1,C6 = 10 μ F 10 V 20%, X5R, 0805 (Taiyo Yuden LMK212 BJ106MG-T)
 C2,C7 = 100 nF 50 V 10%, X7R, 0805
 C3 = 10 pF, 50 V, $\pm 0,5$ pF, COG/NP0, 0805
 C4 = 2,2 μ F 6,3 V, 10%, X5R, 0805
 C5 = 220 μ F 6,3 V, 20%, CMS, Ir=2,8 A (Nichicon PCS0J221MCL1GS)
 C8,C9 = 120 μ F, 6,3 V, 20 %, CMS, Ir=2,8 A (Nichicon PCS0J121MCL9GS)
 C10 = 1 nF, 50 V, 10%, X7R, 0805

Inductances

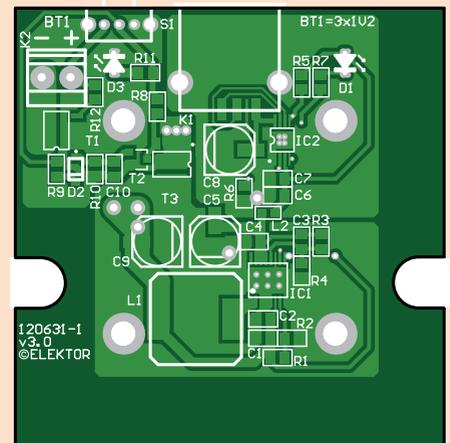
L1 = 10 μ H, 5 A, 25 m Ω , 20% (Würth Electronics 74477110)
 L2 = perle de ferrite, 70 Ω @ 100 MHz, 3,5 A, 22 m Ω , 0603 (Murata BLM18KG700TN1D)

Semiconducteurs

D1,D3 = LED rouge 3 mm (à faible courant)
 D2 = diode zener 5,1 V 0,5 W (SOD123), Diodes Inc. MMSZ5231B-7-F
 IC1 = TPS61030PWPG4 (Texas Instruments)
 IC2 = TPS2511DGN (Texas Instruments)
 T1,T3 = DMS3016SSS-13 (SO8)
 T2 = BC847B

Divers

K1 = embase USB type A CMS encartable
 K2 = bornier vertical deux voies au pas de 3,81 mm (Phoenix Contact MKDS 1/2-3.81)
 S1 = interrupteur à glissière coudé SPDT 100 mA (C&K Components OS102011MA1QN1)
 boîtier 66,22 x 67,22 x 28,00 mm (Hammond Manufacturing 1593KBK)
 vis pour C.I. (#4 x 1/4" autotaraudeuses 6,4 mm, Hammond Manufacturing 1593ATS50)
 BT1 = coupleur pour 3 piles AA (Keystone 2475) + clip pour batteries (BUD Industries HH3449)
 3 batteries NiMH
 circuit imprimé 120631-1 v3.0



à la masse, celui proche d'IC2 sera la masse de la sortie.

Ne touchez jamais la jonction R3/C3/R4 lorsque le circuit fonctionne. C'est un point à haute impédance et tout bruit qui y serait introduit pourrait entraîner la destruction d'IC1.

(120631 – version française : Kévin PETIT)



huit relais et plus encore

module d'extension pour les cartes à μ C Linux et autres

Benedikt Sauter [1]
et Jens Nickel

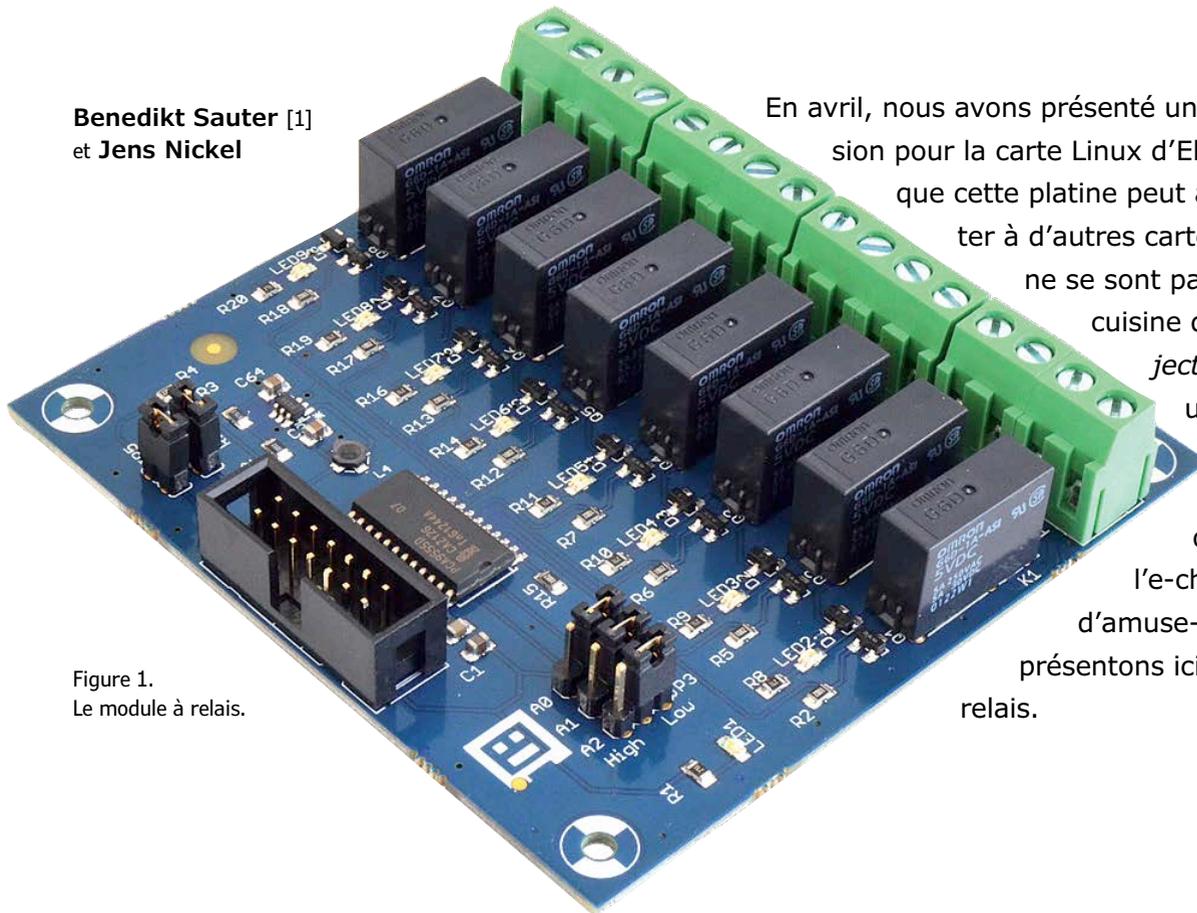
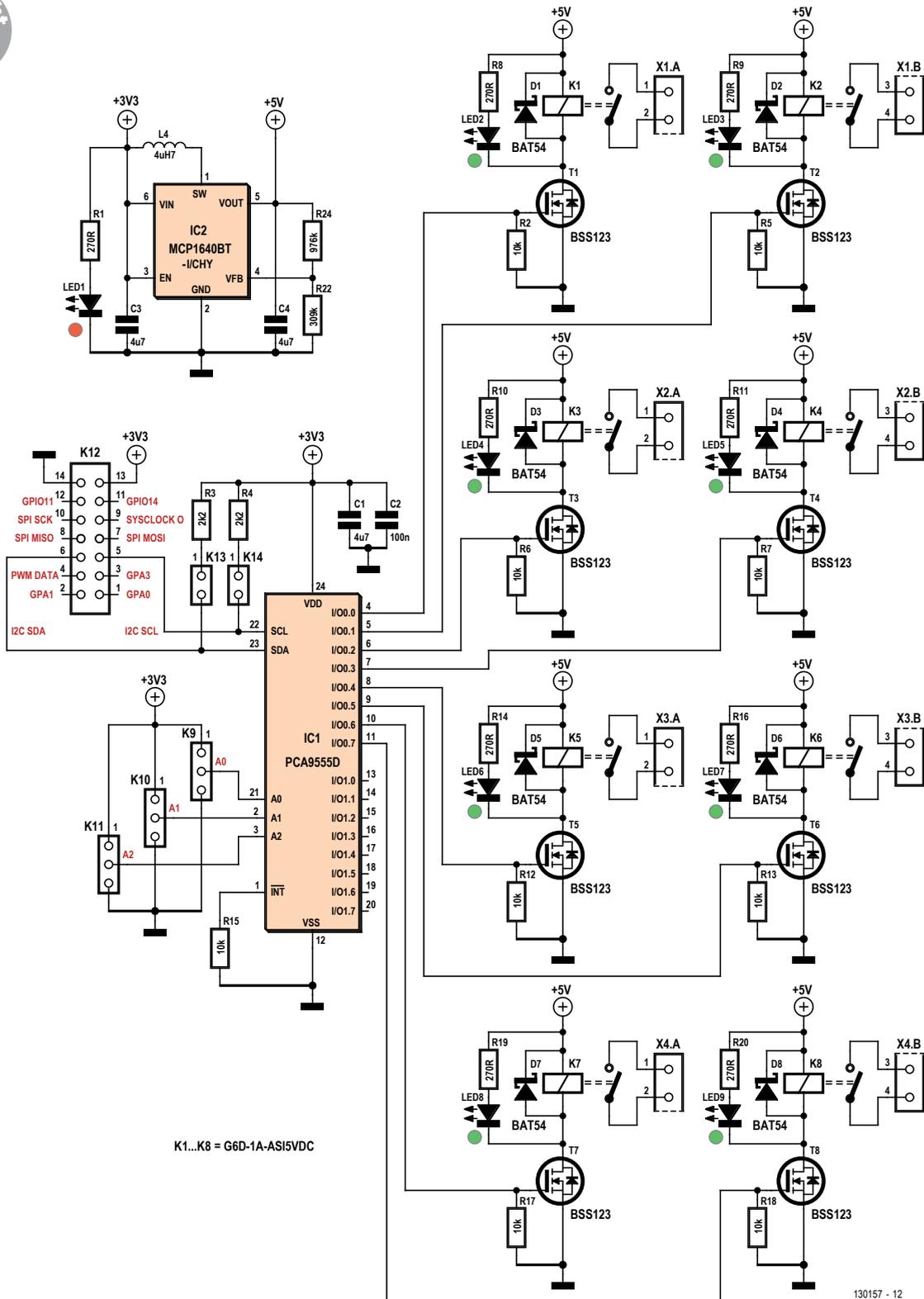


Figure 1.
Le module à relais.

En avril, nous avons présenté une platine d'extension pour la carte Linux d'Elektor et signalé que cette platine peut aussi se connecter à d'autres cartes à μ C. Les feux ne se sont pas éteints dans la cuisine d'*Embedded Projects*, qui a concocté une série d'autres platines d'extension, également disponibles dans l'e-choppe Elektor. À titre d'amuse-gueule nous vous présentons ici une carte à huit relais.

Le module à relais de la figure 1 se raccorde par le connecteur *GnuBlin* à 14 points, comme la *Linux Extension Board* du mois d'avril. Ce connecteur d'extension se trouve aussi sur la carte *Xmega-Webserver* d'Elektor (prochaine livraison) et d'autres cartes à μ C équipées ainsi sont dans les cartons. Cette carte d'extension conviendra donc aux débutants pour qui Linux est (encore) trop compliqué ; elle conviendra aussi aux utilisateurs chevronnés qui préfèrent sculpter leurs programmes « dans le métal brut » (sans l'aide d'un système d'exploitation). La platine à relais fait partie de la famille *Embedded Projects* autour de Benedikt Sauter.

Il s'agit d'une série de cartes d'extension (voir l'encadré) qui acceptent toutes le connecteur cité plus haut, lequel transporte les signaux SPI, I2C, PWM, entrées analogiques et entrées-sorties TOR (tout ou rien). Au labo d'Elektor, au contraire, on travaille à un connecteur d'extension à 10 points (*Embedded Communication Connector*) pour établir les contacts par UART/TTL (voir l'encadré). Tout cela constitue une petite ménagerie de cartes à μ C et de cartes d'extension qui peuvent se combiner de façon souple – les fans de microcontrôleurs peuvent se réjouir à l'approche d'un deuxième semestre intéressant !



130157 - 12

Figure 2.
Schéma du module à relais.

Modules d'extension pour le connecteur GnuBlin/Embedded Extension (Sélection)

8 relais (130212-91)
 Afficheur 4x20 caractères (130212-92)
 Pilote de moteur pas-à-pas (130212-93)
 IO-Expander (130212-94)
 Capteur de température (130212-95)
 Carte de répartition *Bridge Module* (130212-71)
 Adaptateur Raspberry Pi *GnuPi* (130212-72)

Ces platines et d'autres sont disponibles chez Elektor. Les composants CMS sont tous soudés, les composants traversants sont fournis dans le kit en [2].

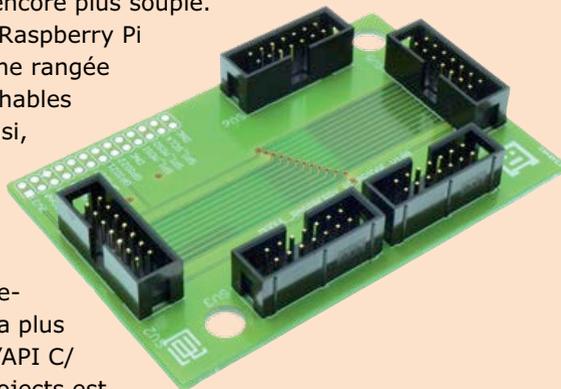
Adaptateur Raspberry Pi

La platine adaptateur Raspberry Pi *GnuPi* rend le système de cartes d'extensions encore plus souple.

Elle s'enfiche sur le Raspberry Pi et offre à nouveau une rangée de connecteurs enfichables GnuBlin/ECC [2]. Ainsi, toutes les cartes d'extension présentées pourront se connecter sur la plateforme informatique la plus tendance. Brillant : l'API C/C++ d'Embedded Projects est utilisable sur Raspberry Pi aussi.

Il suffit, pour convertir une application de la carte Linux GnuBlin/ Elektor vers un Raspberry Pi, de remplacer une ligne de code :

```
#define BOARD_GNUBLIN devient #define BOARD_RASPBERRYPI
```



Modules d'extension pour le connecteur Embedded Communication

Interface RS485 (étude en cours)
 Interface RS232 (prévue)
 Module radio 433 MHz (étude en cours)
 Module bluetooth avec BTM-222 (prévu)
 WLAN avec WizFi220 (prévu)
 USB avec BOB (prévu)

Davantage d'informations sur le site Elektor .Labs [8]

Carte à relais

Le schéma de la carte à relais est présenté à la **figure 2**. Comme sur la carte *Linux Extension*, l'interface parallèle PCA9555 (IC1), pilotée par I2C, augmente de 16, dont 8 sont utilisées, le nombre d'entrées-sorties tout ou rien. L'adresse I2C de la puce est fixée par les cavaliers K9 à K11. Les cavaliers K13 et K14 connectent les résistances de polarisation du bus I2C.

Les sorties tout ou rien IO 0.1 à IO 0.7 de l'interface parallèle attaquent chacune un FET qui commande à son tour un relais. L'état de chacune est signalé par une LED.

Le connecteur GnuBlin/Embedded Extension comporte une broche 3,3 V qui permet d'alimenter les cartes d'extension de la carte à µC. Pour produire la tension de 5 V des bobines de relais, un convertisseur élévateur (IC2) est incorporé. La liaison entre la carte d'extension et la carte µC passe par un câble en nappe. Les concepteurs d'*Embedded Projects* ont déjà prévu aussi la possibilité de connecter plusieurs cartes d'extension en même temps ; la carte de répartition de la **figure 3** est disponible chez Elektor [2].

API C/C++

Nous avons déjà montré en [3] et [4] comment on manœuvre les sorties de l'interface parallèle sous Linux. Toutefois, il y a une possibilité encore plus simple. Benedikt Sauter et son acolyte ont écrit une API (*Application Programming Interface*) complète en C/C++ qui simplifie le pilotage des cartes d'extension. On peut en utiliser les fonctions dans son propre programme, et elle propose en plus une série d'outils en ligne de commande. Nous en reparlerons dans la prochaine livraison, où nous présenterons aussi les autres cartes d'extension.

Les listings donnent un avant-goût de l'API en C/C++. Le listing 1 montre comment on peut accéder aux entrées-sorties TOR de la carte Linux Elektor. Le listing 2 démontre comment on saisit les valeurs des entrées analogiques. Le listing 3 enfin illustre l'utilisation de la carte à relais.

La nouvelle API [5] aplanit pour les débutants le chemin de l'entrée dans le monde Linux ; elle ne comporte aucune des complexités du C, comme les pointeurs par exemple. Les développeurs ont choisi pour nommer les fonctions de rester proches des fonctions Arduino. Les curieux sont invités, bien sûr, à jeter un coup d'œil dans le texte source [6].

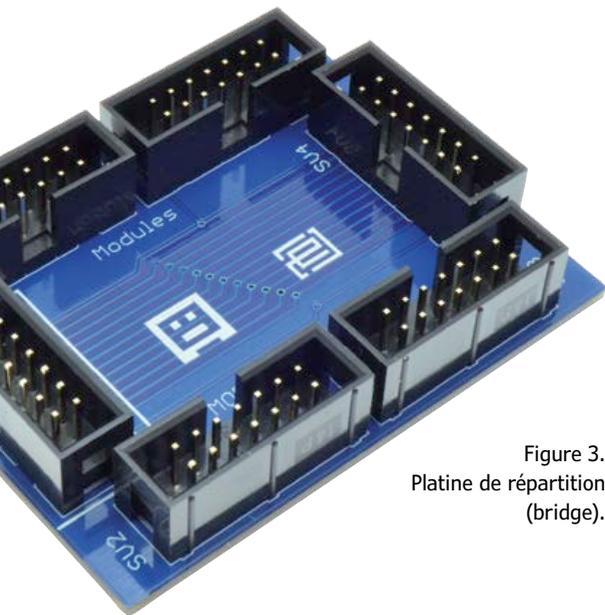


Figure 3.
Platine de répartition
(bridge).

Listing 1 : commande de sortie TOR de la carte Linux Elektor

```
#define BOARD_GNUBLIN
#include "gnublin.h"

int main()
{
    gnublin_gpio gpio;

    gpio.pinMode(3,OUTPUT);

    while(1){
        gpio.digitalWrite(3,HIGH);
        sleep(2);
        gpio.digitalWrite(3,LOW);
        sleep(2);
    }
}
```

Debian pour la carte Linux Elektor

Le système Gnublin Linux ne s'est pas enrichi seulement d'un nouveau matériel, il fait aussi une avancée en matière de logiciel. Celui qui le souhaite peut installer sur la carte Linux Elektor un système Debian (au lieu du système de fichiers ELDK). Peu importe qu'il s'agisse de la version 8 Mo ou 32 Mo de la carte, Debian s'installe simplement avec une carte SD ; un mode d'emploi est disponible sur l'internet [7].

(130157 – version française : Jean-Paul Brodier)

Listing 2 : lecture de la valeur d'une entrée analogique

```
#define BOARD_GNUBLIN
#include "gnublin.h"

int main()
{
    gnublin_adc ad;

    while(1){
        printf("AD value %i \n",ad.getValue(1));
    }
}
```

Listing 3 : commande de la carte à relais

```
#define BOARD_GNUBLIN
#include "gnublin.h"

int main() {
    gnublin_module_relay relay;

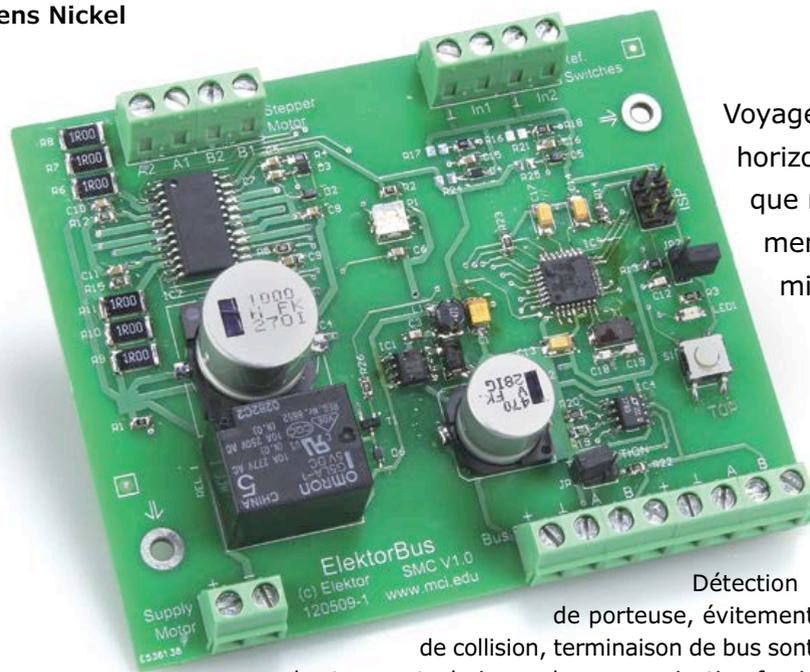
    relay.setAddress(0x24);
    relay.switchPin(4, ON);
    sleep(2);
    relay.switchPin(4, OFF);
}
```

Liens

- [1] sauter@embedded-projects.net
- [2] www.elektor.com/gnublin
- [3] www.elektor.fr/120596
- [4] www.elektor.fr/120518
- [5] <http://wiki.gnublin.org/index.php/API>
- [6] <https://github.com/embeddedprojects/gnublin-api>
- [7] <http://wiki.gnublin.org/index.php/GNUBLIN-Elektor>
- [8] www.elektor-labs.com/ECC

pilote de moteur pas à pas pour ElektorBus

Ronald Stärz,
Mathias Gfall &
Jens Nickel



Voyager avec l'ElektorBus ouvre de nouveaux horizons. Grâce à sa modularité logicielle autant que matérielle, ce bus accélère le développement d'applications. Exemple pratique : la mise au pas des moteurs !

Détection de porteuse, évitement de collision, terminaison de bus sont des termes techniques de communication familiers aux étudiants autant qu'aux amateurs. Tout devient plus palpitant dès qu'on peut s'adonner à ses projets et ses circuits personnels sur le terrain des systèmes de bus. Et en la matière, l'ElektorBus définit des bases solides pour réaliser des applications auxquelles il tient lieu de référence. C'est ainsi que l'ElektorBus joue à présent un rôle primordial dans les études d'électromécanique au MCI, le *Management Center d'Innsbruck* [1] en Autriche. Cette carte pour moteur pas à pas est le résultat exemplaire du travail d'un étudiant

pour son projet de fin d'études. Elle est disponible auprès d'Elektor, comme circuit imprimé nu ou comme module assemblé et testé. Le logiciel pour démarrer une application personnelle est à télécharger gratuitement, comme d'habitude, sur le site [2].

Premiers pas

Le circuit (**fig. 1**) se répartit en plusieurs blocs. Le noyau est constitué du microcontrôleur et du pilote de moteur séquentiel. En orbite, on trouve l'interface de bus, une alimentation à découpage et des contacts de position et de fin de course, adaptables à l'infini.

La **figure 2** laisse entrevoir ce dont est capable la puce L6208 de STMicroelectronics en matière de commande du moteur pas à pas. On y reconnaît les diodes de roue libre intégrées, indispensables à la protection des semi-conducteurs de puissance en présence de charges inductives et de tout ce qui comporte des bobinages. Nous avons choisi ce module parce qu'il contient déjà les étages de puissance et un automate pour élaborer les signaux de commande nécessaires. En outre, il régule le courant dans les bobines du moteur et en surveille l'intensité. Il simplifie sérieusement la conception en produisant les quatre signaux numériques requis pour le sens de rotation (CW/CCW), l'horloge de pas (CLK), le mode de tra-

Caractéristiques techniques

- pilote de moteur pas à pas L6208 de STMicroelectronics
- pour moteur pas à pas bipolaire jusqu'à 60 V
- courant moteur maximum 2,5 A
- interface RS485
- 1 LED, 1 bouton de test, débogage et mise en service
- 1 relais de sécurité pour la consommation du moteur, sous contrôle du logiciel
- fichiers contrôleur et carte pour EFL à télécharger
- bibliothèque du moteur pas à pas pour EFL à télécharger
- compatible ElektorBus, source ouverte, application de démonstration à télécharger

vail (HALF/FULL) et une validation (EN). D'autre part, une telle intégration de fonctions réduit à un minimum le nombre de composants externes (fig. 3). Les résistances R6 à R11 sur les entrées de capteurs servent à la mesure du courant, les chutes de tension qu'elles occasionnent représentent les grandeurs de mesure pour le régulateur de courant interne. La valeur de commande est appliquée aux entrées Vref par P1. Avec les valeurs de composants indiquées, le courant de bobine est de 100 mA lorsque la tension de référence est mise à 33,3 mV. Les réseaux R12/C10 et R15/C11 définissent les temps morts des MOSFET de puissance à l'intérieur du montage en pont [3]. Pour produire les tensions de grille, le L6208 se sert d'une pompe de charge complétée de D2, D3, R4, C5 et C7. Le réseau R5 et C9 procure le niveau de tension convenable à l'entrée de validation.

Nous avons choisi comme contrôleur un ATmega328, avec ses 32 Ko de flash, il procure suffisamment de mémoire pour l'utilisation d'une bibliothèque modulaire de logiciels telle que notre *Embedded Firmware Library* (EFL), comme nous allons le voir. Le circuit du microcontrôleur comprend, en plus des composants classiques pour stabiliser la tension et produire le signal d'horloge, une possibilité de réarmement manuel par JP2. Le bouton-poussoir S1 sur PD5 sert à la mise en service et la LED1 sur PD4 à l'indication du bon débogage d'une application. Au niveau matériel, la liaison au bus passe par le LT1785, un transcepteur RS485 ; si la carte moteur est la première ou la dernière participante du réseau, le cavalier JP3 et la résistance R22 en assurent le bouclage. Comme de coutume sur les nœuds de l'ElektorBus, des borniers à vis permettent un raccordement aisé des signaux RS485 et de l'alimentation de 12 V, ils sont prévus en double exemplaire pour faciliter le câblage des quatre lignes du bus.

Alimentations

Le régulateur à découpage IC1, un LM2675M-5 procure la tension d'alimentation de la logique. Avec l'aide de L1, D1, C1 et C3, on obtient pratiquement sans perte les 5 V nécessaires, à partir d'une tension d'entrée comprise entre 7 et 24 V. L'alimentation de l'électronique de puissance est séparée, elle arrive sur des bornes de raccordement pour ne pas devoir transiter par les lignes à 12 V de l'ElektorBus vers les moteurs pas à pas gloutons. Dans cette configuration, il y a toutefois un inconvénient avec le circuit pilote L6208 : si

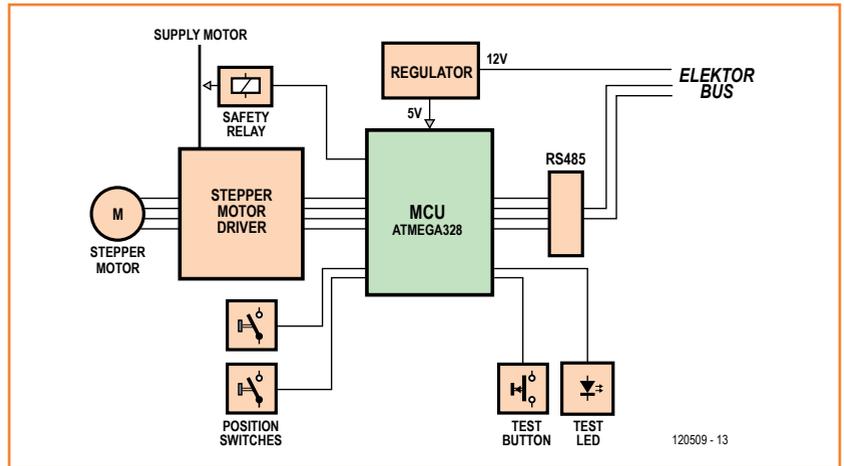
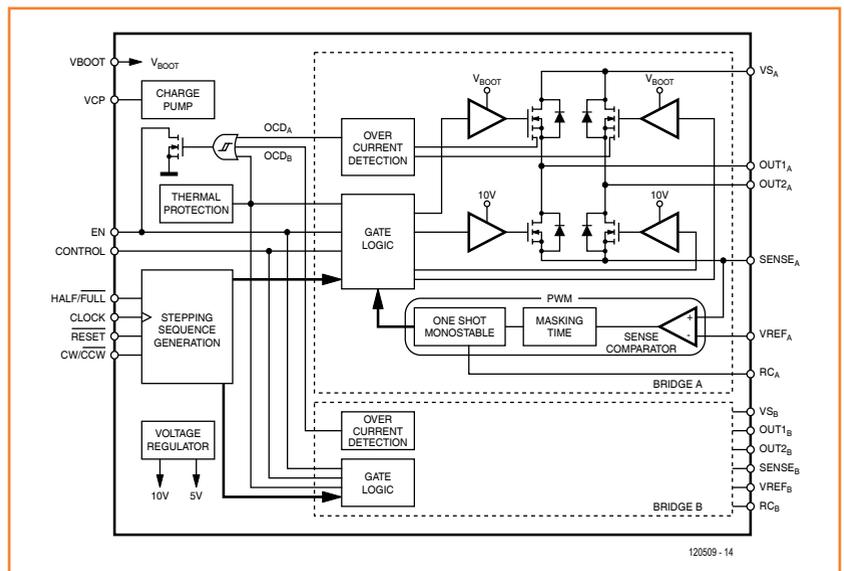


Figure 1. Diagramme fonctionnel de la carte de moteur pas à pas pour l'ElektorBus.

l'alimentation des composants de puissance est appliquée, alors que le 5 V pour la logique ne l'est pas, le L6208 n'apprécie pas : signaux de fumée garantis ! C'est pourquoi nous avons prévu d'alimenter les moteurs par un relais (REL1) qui, sous 5 V, est commandé par le logiciel via la ligne PD6 du contrôleur. Ainsi, toute la section de puissance pourrait, le cas échéant, être mise hors tension via l'ElektorBus. Les courants de plusieurs centaines de milliampères pourraient provoquer des chutes de tension problématiques sur les conducteurs de masse. Aussi avons-nous constitué un point de masse en étoile situé près de R1, une résistance de 0 Ω. R1 permet de séparer la masse du circuit logique de l'alimentation en cas de panne. Quand un moteur séquentiel ne sert pas seulement de moteur à vitesse variable, mais est

Figure 2. Diagramme fonctionnel du pilote de moteur pas à pas L6208 de STMicroelectronics [3].



employé pour déterminer une position, il faut y ajouter des interrupteurs de fin de course et de position. Ces commutateurs peuvent être mécaniques ou constitués de détecteurs de proximité par effet inductif ou capacitif, voire des barrières lumineuses, pour ne citer que quelques exemples. Pour répondre à ces diverses possibilités, la carte dispose d'un circuit d'entrée adaptable à volonté sur deux canaux, qui comprend R16, R17, R24, C15 et D4 ainsi que R18, R21, R25, C16 et D5. Les circuits de protection et de filtrage composés de C15/C16 et des Zener D4/D5 sont en tout cas

indispensables, les résistances étant soumises à des exigences spécifiques.

Différentes possibilités sont répertoriées dans l'encadré. Dans l'état d'origine, les cartes ne sont pas équipées des résistances de 0 Ω R17/R21 et R16/R18 pour pouvoir y implanter un simple bouton-poussoir ou un interrupteur mécanique. R24 et R25 sont des résistances de polarisation haute optionnelles à insérer, mais celles du contrôleur AVR, que l'on peut sélectionner par logiciel, suffisent à un fonctionnement correct.

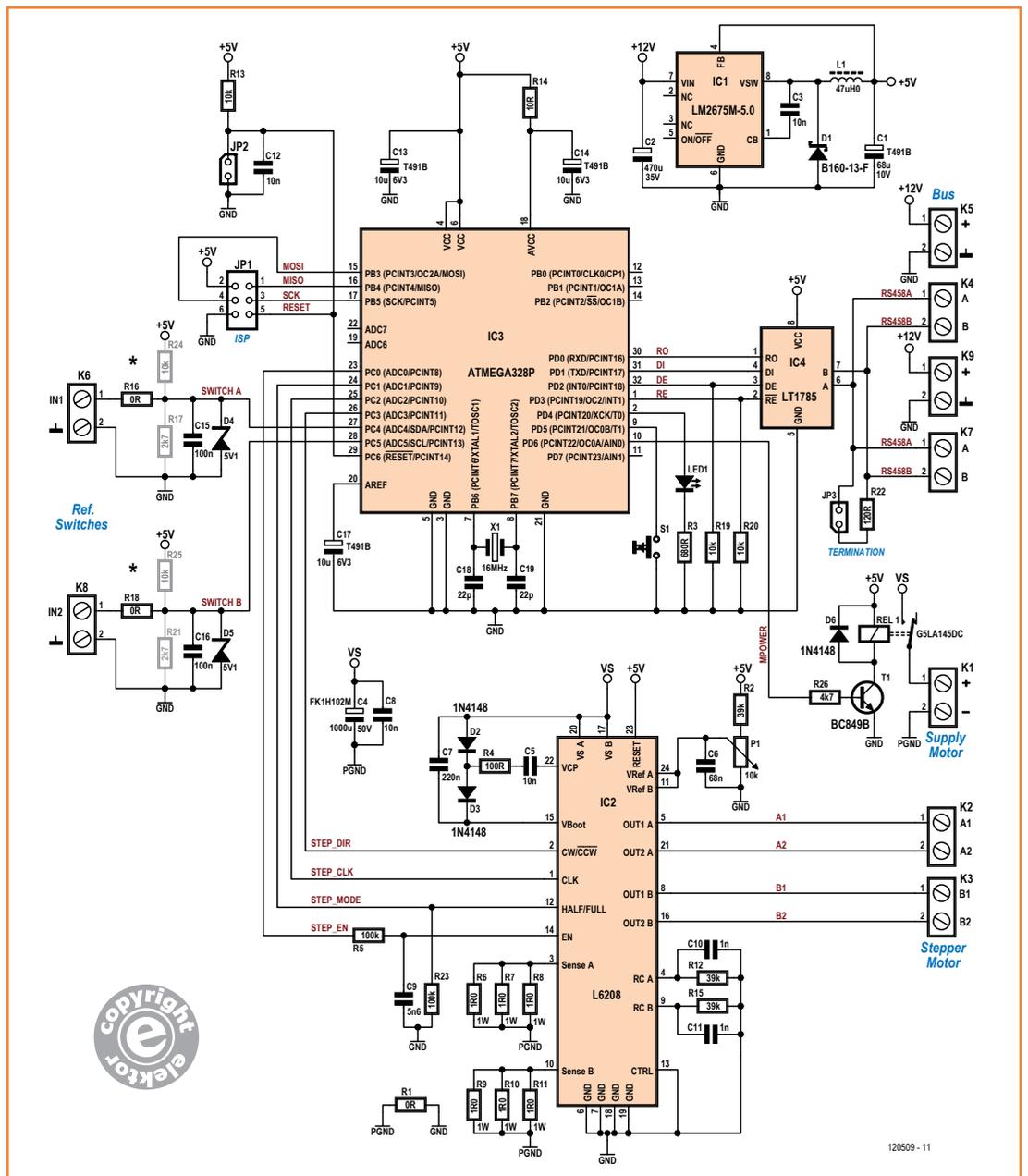


Figure 3. Schéma de la carte de moteur pas à pas. Voyez l'encadré pour la configuration de l'interrupteur de position.

Configuration de l'interrupteur de position

Switch :

Si un interrupteur est branché, il faut mettre une résistance de 0Ω pour R16, rien pour R17. La résistance de polarisation haute R24 de $10 \text{ k}\Omega$ est optionnelle, pour le cas où le contrôleur en service n'en posséderait pas en interne. Si l'interrupteur est fermé, l'entrée correspondante du microcontrôleur est à la masse (active bas). C'est la configuration comme réglage d'usine sur la carte à commander chez Elektor.

Voltage Input 5 V :

Si R16 est pontée et R24 absente, le signal de sortie à 5 V filtré d'un interrupteur de référence se trouve à l'entrée du microcontrôleur.

Voltage Input 12 V :

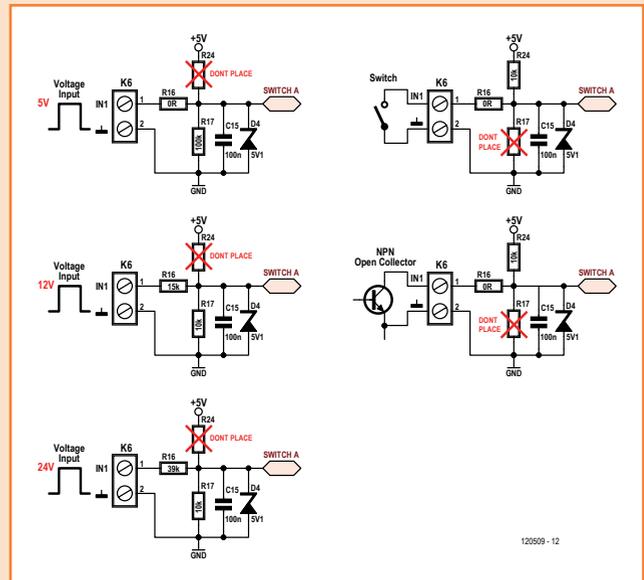
Un diviseur de tension définit un rapport de $25 \text{ k} / 10 \text{ k} = 2,5$, ce qui représente une tension de 4,8 V sur le microcontrôleur.

Voltage Input 24 V :

En raison du diviseur de tension R16 / R17, le signal de sortie est de 4,9 V.

NPN Open Collector :

Les interrupteurs avec ce genre de sortie n'ont aucun signal de sortie sans un câblage adéquat. Mieux vaut adapter le niveau par une résistance de polarisation haute en fonction des besoins du circuit. S'il y a une tension sur la base du transistor, la diode collecteur-émetteur est passante et le potentiel de l'entrée du microcontrôleur est attiré à la masse.



Logiciel

Ce n'est pas par hasard que nous avons choisi l'ATmega328 ; il nous avait déjà donné toute satisfaction sur le nœud expérimental de l'ElektorBus. Et comme la série d'articles sur l'ElektorBus [5] utilisait un contrôleur compatible avec l'ATmega88, une bonne partie du logiciel est réutilisable tel quel ici, modifié ou pas.

Du code réutilisable ou à adapter facilement pour cette carte, nous en avons en suffisance, le magazine de mai a présenté EFL, la bibliothèque de micrologiciel embarqué [6] et avec elle, tout ce qui concerne la communication sur l'ElektorBus (fig. 4). Qui plus est, EFL soutient l'ATmega328 avec un « Controllerfile ». Des blocs périphériques comme le pilote RS485, la LED et le bouton sont raccordés aux mêmes broches du contrôleur que dans le nœud expérimental pour lequel nous avons déjà un fichier de carte et son code de base en EFL. Nous pouvons alors initialiser dans ce « Boardfile » la seconde LED, supprimer l'autre bouton-poussoir et le connecteur d'extension, puisqu'il n'y en a pas sur la carte moteur. Si nous ajoutons encore le moteur pas à

pas et d'autres blocs périphériques, nous aurons développé notre propre application en toute indépendance du matériel choisi. Mais puisque nous avons encore prévu d'autres cartes ElektorBus avec différentes fonctions supplémentaires, engageons-nous sur une meilleure piste. Les fonctions périphériques qui reviennent sur toutes les cartes d'ElektorBus comme le pilote RS485, les capsules de la LED et du bouton de test, mettons-les dans un fichier de carte « BoardEFL.h/.c » que nous insérons en code de base dans le dossier « ElektorBusCore ». Les fonctions spéciales sur la carte, comme ici le moteur pas à pas, seront initialisées dans un fichier d'extension (ExtensionEFL.h/.c dans le dossier ElektorBusStepperMotor). En fait, nous disons simplement que l'ATmega328 est connecté au pilote du moteur par l'intermédiaire d'un port d'extension virtuel (fig. 5).

Code du moteur pas à pas

Vous obtiendrez le code source correspondant sous [2]. Le projet EFL s'appelle *ElektorBusStepperMotor*, vous l'ouvrez en Studio d'Atmel d'un clic sur ElektorBusStepperMotor.atsuo. Dans le

Figure 4. Avec les modules logiciels d'EFL, on rédige le micrologiciel en très peu de temps.

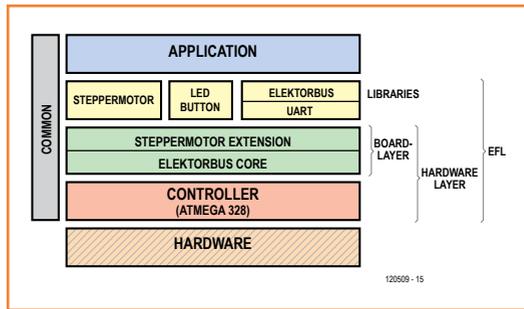
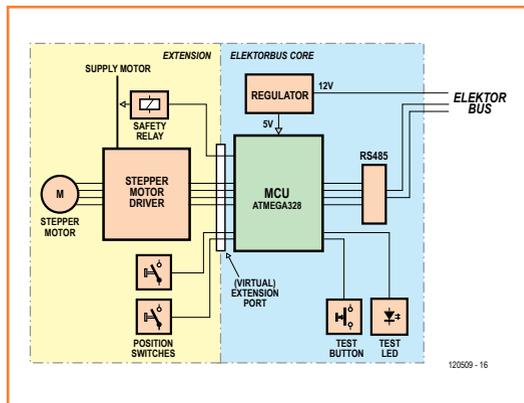


Figure 5. La carte est (virtuellement) divisée en un noyau ElektorBus et un domaine d'extension. Le même code de carte peut alors être réutilisé sur tous les circuits imprimés ElektorBus



dossier de projet Hardware, on reconnaît les fichiers de carte BoardEFL.h/.c avec une initialisation des fonctions de base de l'ElektorBus et du connecteur d'extension (virtuel). Dans le fichier d'extension se trouve la fonction Extension_Init() à appeler au démarrage de l'application. Il y a dans la fonction un bloc pour le pilote du moteur. Dans le tableau de Boardpin (le document annexe [6] sur EFL explique de quoi il s'agit), trois broches sont inscrites au port d'extension pour les signaux EN, CLK et DIR, tandis que le signal MODE n'est pas utilisé dans le logiciel. Tant que nous y sommes, inscrivons encore un bloc pour le relais de sécurité et un autre pour les deux interrupteurs de position. Tout cela, on peut toujours l'adresser tout en restant bien indépendant du matériel. On ne doit plus connaître les broches du contrôleur, seulement le type et le numéro du bloc périphérique. Bon, nous pouvons maintenant utiliser les signaux EN, CLK et DIR indépendamment du câblage de la carte, mais nous ne pouvons pas certifier que n'importe quel moteur pas à pas peut être commandé pareillement avec seulement trois entrées. Pour que le développeur d'application ne soit pas obligé d'aller chaque fois consulter le cahier de caractéristiques et que le code reste indé-

pendant du matériel, il faut implémenter dans le fichier d'extension quelques autres fonctions de bas niveau pour les moteurs pas à pas. Par exemple, fixer le sens de rotation avec un bit de direction 0 ou 1 :

```
void StepperMotorDirection(uint8
StepperMotorBlockIndex, uint8 Direction)
```

ou un déplacement d'un pas :

```
void StepperMotorStep(uint8
StepperMotorBlockIndex, uint8
MillisecondsDelay)
```

Cette fonction Step bloque le déroulement du programme. Pour une exigence plus professionnelle, nous avons implémenté une autre fonction qui fait tourner le moteur pendant un certain temps. Cette fonction demande encore une chaîne de caractères comme paramètre pour construire une rampe de vitesse. Le moteur démarre lentement, accélère pour atteindre une vitesse constante ; à la fin, il refait le chemin inverse. Remarquez que dans ce mode, il n'est pas possible d'interroger un interrupteur de position.

On commence par initialiser le temporisateur avec :

```
void StepperMotorTimerSetup(uint8
StepperMotorBlockIndex, uint8
MillisecondsDelay)
```

où l'on indique le délai en millisecondes par pas pour que le moteur atteigne une vitesse de base donnée. Comme horloge, la fonction recherche le premier temporisateur à 16 bits libre du contrôleur.

Avec la fonction :

```
void StepperMotorTimerSteps(uint8
StepperMotorBlockIndex, uint16 Steps,
uint8* RampData, uint8 RampMax, uint8
StepsShiftForNextRampIndex)
```

on lance maintenant le fonctionnement du moteur par minuterie. RampData et RampMax décrivent la rampe, une valeur de 128 dans la chaîne de caractères donne la vitesse de base, une valeur plus petite entraîne une vitesse plus élevée. Avec la variable StepsShiftForNextRampIndex, on peut déterminer le nombre de pas nécessaires pour atteindre la vitesse suivante. Il s'agit ici d'une

puissance de deux et avec un 2, le moteur fait toujours quatre pas à la même vitesse. Quand toutes les valeurs de la chaîne ont été utilisées, le moteur continue à tourner à vitesse constante. À la fin du mouvement, les valeurs de la chaîne sont relues en sens inverse.

Les fonctions de bas niveau dans la couche matérielle demandent beaucoup de travail de la part du développeur. Pourtant, il nous faut encore rédiger une fonction pour interroger l'interrupteur de position et encore une autre pour l'étalement du mouvement du moteur. Nous les avons toutes deux emballées dans une petite bibliothèque pour moteur pas à pas au-dessus de l'interface de bas niveau décrite précédemment. Comme d'habitude avec les bibliothèques EFL, elles servent à commander plusieurs blocs périphériques du même type, dans ce cas-ci jusqu'à huit moteurs pas à pas sur une carte. Le code source StepperMotorEFL.c se trouve dans le classeur Libraries. Vous trouverez une description complète des fonctions dans la documentation Doxygen archivée pour le téléchargement via [2].

Petite démonstration

Pour un test et une démonstration, nous avons construit un montage avec un moteur pas à pas de 12 V (Nanotec SP2575M0206-A), un interrupteur de position et deux de fin de course (**fig. 6**). La carte moteur est branchée sur l'ElektorBus avec une vieille connaissance, le convertisseur RS485/USB, lui-même connecté au PC.

Le code de l'application se trouve dans le fichier principal du projet (ElektorBusStepperMotor.c). Nous avons déjà décrit [6] [7] la fonction principale EFL toujours pareille et l'initialisation des bibliothèques LEDButton, UARTInterface et ElektorBus dans la fonction ApplicationSetup().

Avec :

```
StepperMotor_LibrarySetup(SwitchEventCallback, 0, 0);
```

nous initialisons la bibliothèque du pilote de moteur. Le premier paramètre identifie une fonction appelée dans le code d'application dès qu'un contact de fin de course est atteint. Cette fonction, située plus loin dans le fichier principal, nous l'avons implémentée de manière à faire clignoter la LED sur la carte. Les autres paramètres (0, 0) donnent la possibilité d'un décalage des numéros de blocs entre le moteur pas à pas, la position

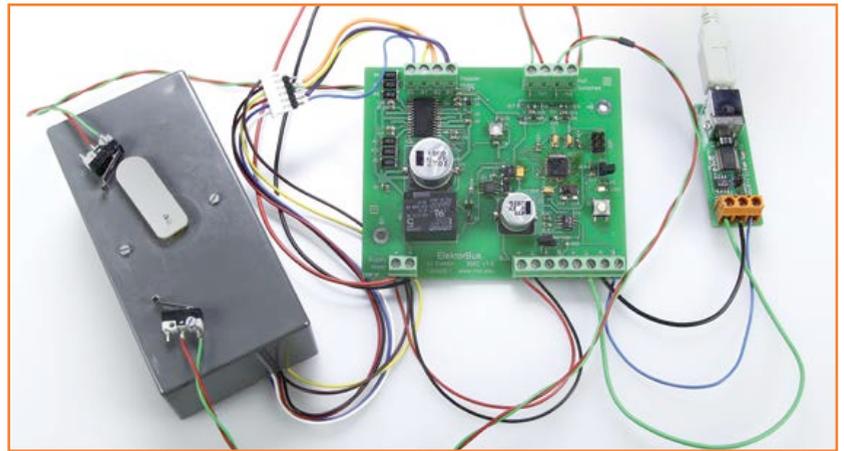


Figure 6. Une démonstration avec interrupteur de position. Le circuit imprimé de moteur pas à pas est relié au PC par un convertisseur RS485/USB.

des contacts et le relais de sécurité. Dans notre cas, nous avons le bloc interrupteur #0, le bloc relais #0 pour le premier moteur #0, mais sur une autre carte, ce pourrait être différent.

Quand l'utilisateur appuie sur le bouton de la carte, il déclenche un étalement automatique. Le moteur se met à tourner dans un sens puis dans l'autre jusqu'à ce qu'il rencontre un contact de position et le nombre de pas nécessaires est comptabilisé. La fonction de bibliothèque StepperMotorCalibration(...) reconnaît en même temps dans quel sens le moteur a tourné avec les bits de direction 0 et 1 et le retient pour une utilisation ultérieure.

Du coup, on peut envoyer le moteur à un endroit déterminé à l'intérieur du domaine exploré, les positions finales seront alors 0 et 1023.

Un appel de

```
StepperMotor_GotoMotorPosition(0, 512, 4);
```

envoie le moteur pas à pas en position centrale à l'allure d'environ 4 ms par pas.

Nous voulons évidemment le piloter par l'ElektorBus. On se programme vite une petite interface HTML qui tourne sur le PC, mais comme d'habitude, la même interface peut servir sur un smartphone Android ou une tablette [8] [9]. Le fichier HTML se trouve dans le dossier UIBus, il suffit de le prendre et de le glisser sur le bureau. La suite est bien connue : lancer ElektorBusBrowser.exe, choisir le port COM et cliquer sur Connect, puis mettre en route le superviseur. Les boutons HTML amènent alors les messages ElektorBus du PC à la carte du moteur et la position voulue du moteur est alors transmise sur le canal 0 comme une valeur sur 10 bits.

Dans le micrologiciel, nous recevons les mes-

sages par la bibliothèque ElektorBus qui appelle la fonction ProcessPart(). Elle traite les données des Parts entrantes et fait se déplacer le moteur jusqu'à la position voulue.

Comme toujours, ce logiciel de démonstration n'est qu'une suggestion en vue d'autres expériences. Une application possible pourrait être une commande de volet roulant ou d'un store vénitien pour obscurcir progressivement une chambre quand la lumière est trop intense. Matériel et logiciel pour une cellule photoélectrique sur ElektorBus sont déjà prêts [6] [10].

Dans le prochain numéro, vous aurez droit à une autre Arlésienne, vous en avez déjà largement entendu parler : la carte Xmega-Webserver qui dispose aussi d'une connexion RS485 pour être parfaitement compatible avec l'ElektorBus.

(120509 – version française : Robert Grignard)

Liens

- [1] www.mci.edu
- [2] www.elektor.fr/120509
- [3] www.st.com/st-web-ui/static/active/en/resource/technical/document/datasheet/CD00002294.pdf
- [4] www.elektor.fr/110258
- [5] www.elektor.com/elektorbus
- [6] www.elektor.fr/120668
- [7] www.elektor.fr/130154
- [8] www.elektor.fr/110405
- [9] www.elektor.fr/120097
- [10] www.elektor.fr/110428



Liste des composants

Résistances :

- (0805 sauf indication contraire)
- R1 = 0 Ω
- R2, R12, R15 = 39 kΩ
- R3 = 680 Ω
- R4 = 100 Ω
- R5, R23 = 100 kΩ
- R6..R11 = 1R0 (Vishay CRCW25121R00FKEG)
- R13, R19, R20 = 10 kΩ
- R14 = 10 Ω
- R16, R18 = 0 Ω (cf. encadré)
- R17, R21 = supprimé (cf. encadré)
- R22 = 120 Ω
- R24, R25 = 10 kΩ opt. (cf. encadré)
- R26 = 4k7
- P1 = 10 kΩ aj. (POT4MM-2)

Condensateurs :

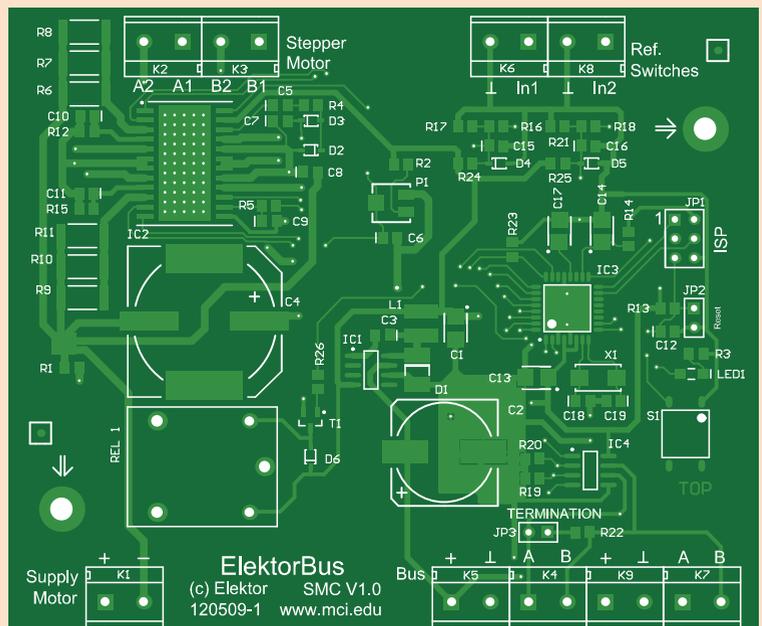
- (0805 sauf indication contraire)
- C1 = 68 μF / 10 V tantale (AVX TPSB686K010R0600)
- C2 = 470 μF / 35 V électrolytique (Panasonic EEEFK1V471AQ)
- C3, C5, C8, C12 = 10 nF
- C4 = 1000 μF / 50 V électrolytique (Panasonic EEVFK1H102M)
- C6 = 68 nF
- C7 = 220 nF
- C9 = 5,6 nF
- C10, C11 = 1 nF
- C13, C14, C17 = 10 μF / 6V3 (AVX TCJA106M006R0300)
- C15, C16 = 100 nF
- C18, C19 = 22 pF

Inductances :

- L1 = 47 μH (744773147)

Semi-conducteurs :

- D1 = Schottky 1 A / 60 V (B160-13-F)
- D2, D3, D6 = 1N4148
- D4, D5 = Zener 5V1 (BZX384-B5V1)



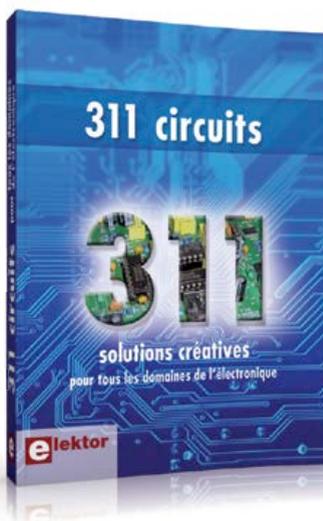
- T1 = BC849B, SOT-23
- LED1 = LED verte (5988270107F)
- IC1 = LM2675M-5.0
- IC2 = L6208D (SO24)
- IC3 = ATmega328P-AU
- IC4 = LT1785CS8

Divers :

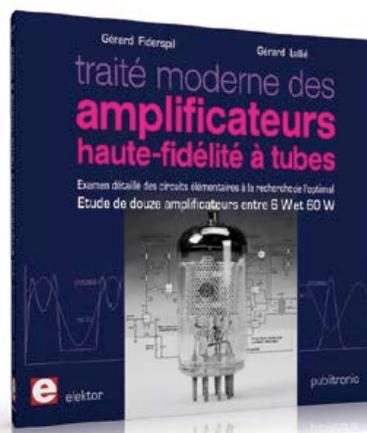
- JP1 = embase à 2x3 picots, 2,54 mm
- JP2, JP3 = embase à picots 1 rangée 2,54 mm
- K1 à K9 = bornier à vis encartable au pas de 5,08 mm
- Rel1 = relais SPDT (Omron G5LA145DC)
- S1 = poussoir (Omron B3S-1000)
- X1 = quartz 16 MHz, 50 ppm, 16 pF (Epson Toyocom FA-365)
- circuit imprimé 120509-1



1



2



3



4

Électricité gratuite

1 Photovoltaïque

Nous avons tous peu ou plus caressé l'espoir de connaître un jour l'autonomie énergétique grâce à une production électrique photovoltaïque appropriée ! D'abord on en rêve, puis vient l'étape du calcul : comparons le coût du kilowattheure d'origine nucléaire (environ huit centimes d'euro) à celui du kilowattheure d'origine photovoltaïque (environ 22 centimes d'euro*), et nous saurons où en sont nos motivations d'écocitoyens.

Ce livre décrit trois projets simples, dans un contexte domestique, et montre comment électrifier soi-même, à partir de l'énergie photovoltaïque, trois sites ordinaires : un abri de jardin, un garage et un mobile-home.

254 pages – ISBN 978-2-86661-181-1 – 33,00 €

Solutions créatives pour tous les domaines de l'électronique

2 311 circuits

Cet ouvrage est un trésor : il réunit 311 schémas d'électronique analogique, logique ou numérique, des programmes, des liens vers des sites internet, des tableaux de caractéristiques de composants et des

dessins de circuit imprimé. Il est le onzième volume de la collection « 3xx circuits ». Ses deux tables des matières alphabétique et thématique vous permettent de trouver rapidement et facilement parmi les 311 articles proposés ceux qui répondront à vos besoins. Ces articles viennent des numéros doubles récents de la revue Elektor. C'est une source d'inspiration inépuisable, et à partir de laquelle chacun élaborera ses propres variantes qu'il combinerait ensuite à sa guise avec d'autres circuits. Tous les domaines familiers et usuels de l'électronique sont abordés.

448 pages • ISBN 978-2-86661-184-2 • 36,00 €

Étude de 12 amplificateurs entre 6 W et 60 W

3 Traité moderne des amplificateurs haute-fidélité à tubes

Les amplificateurs haute-fidélité à tubes électroniques, restés les favoris des audiophiles, reviennent sur le devant de la scène. Pendant cinq ans, les deux auteurs ont effectué 6000 simulations informatiques et disséqué pas moins de 1200 maquettes de circuits à tubes pour en livrer les secrets et en découvrir les limites.

Dans ce livre, le lecteur acquiert une connaissance

approfondie de la conception des amplificateurs à tubes pour mieux apprécier la qualité du matériel existant et optimiser ses propres créations.

Ce livre est le fruit du plaisir intellectuel de l'hagiographie, c'est-à-dire réécrire de manière linéaire une logique intellectuelle mûrie en 60 ans dans un foisonnement d'idées souvent géniales et dans une créativité désordonnée et logique à la fois.

342 pages – ISBN 978-2-86661-160-6 – 46,00 €

Kit complet avec circuit imprimé, afficheur et microcontrôleur

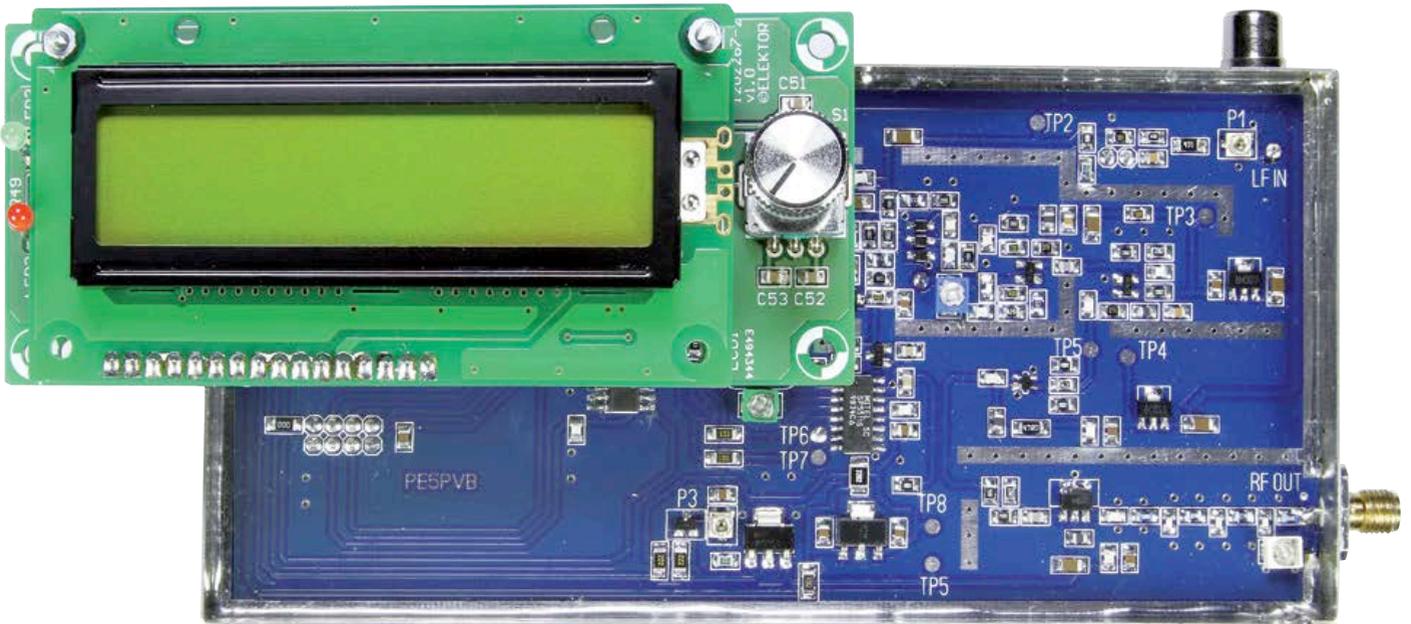
4 Détecteur de rayonnement

Il ne faut guère plus qu'une photodiode PIN et un amplificateur de capteur adapté pour mesurer un rayonnement radioactif. Nous présentons ici un préamplificateur optimisé couplé à un compteur particulier : un microcontrôleur qui gère la durée de la mesure et affiche la fréquence des impulsions en « counts per minute ».

Réf : 110538-71 – 39,95 €

www.elektor.fr/e-choppe

émetteur FM large spectre en bande 70 cm puissance : 130 mW



Sjef Verhoeven
(Pays-Bas)

La plupart des émetteurs HF pour radioamateurs ne sont pas conçus pour transmettre des signaux audio de grande qualité, avec une large bande passante. Cet émetteur-ci s'attache précisément à couvrir un domaine de 20 Hz à 100 kHz, en modulation de fréquence, avec une couverture complète de la bande des 70 cm, de 430 à 440 MHz.

Caractéristiques techniques

- Domaine de fréquence : 430 à 440 MHz par pas de 25 kHz
- Domaine BF : 20 Hz à 100 kHz
- Temps d'accrochage de la PLL < 1 seconde
- Tension d'alimentation : 12 à 15 V
- Consommation : env. 250 mA à 13 mW de puissance de sortie HF

Nous avons tous joué avec des petites radios ou des talkies-walkies. Ils permettent la conversation sans fil, mais la qualité du son est faible et il faut toujours attendre que le correspondant se taise (*over*) pour lui répondre. Cette manière de

communiquer a été instituée au début du siècle passé par les radioamateurs. Ceux-ci parviennent à établir, grâce au choix judicieux de l'appareil, de la fréquence et de l'antenne, des liaisons radio de toutes sortes dans le monde entier avec un spectre pourtant limité.

À quoi bon un spectre de modulation plus large ? C'est déjà le cas, en audio jusqu'à 2 x 15 kHz, des stations de radiodiffusion dans la bande FM. L'émetteur décrit ici opère dans le domaine de 430 à 440 MHz et permet de transmettre une large gamme de basses fréquences jusqu'à 100 kHz environ. De quoi s'en servir facilement

pour des conversations en réseau, avec de multiples correspondants. À comparer avec *Teamspeak*, mais sans accrocs, distorsions de codec, échos ni retards. Avec une bonne antenne, à la hauteur adéquate, la portée est de dizaines de kilomètres et lors de conditions atmosphériques favorables, jusqu'à 300 km. Aussi, l'utilisation de ce transmetteur demande-t-elle de posséder une autorisation d'émission comme radioamateur dans la catégorie Novice ou une licence complète. Et pour l'obtenir, il faut avoir réussi l'examen technique. Le but est de posséder les connaissances théoriques suffisantes pour utiliser la radio sans créer de perturbations, en toute sécurité et conformément à la législation. Pas question donc de diffuser à la ronde n'importe quelle musique. Cet examen est organisé plusieurs fois par an en différents endroits. Renseignez-vous auprès de l'organisme responsable dans votre pays, l'Agence nationale des fréquences en France, l'IBPT en Belgique, l'OFCOM en Suisse, etc.

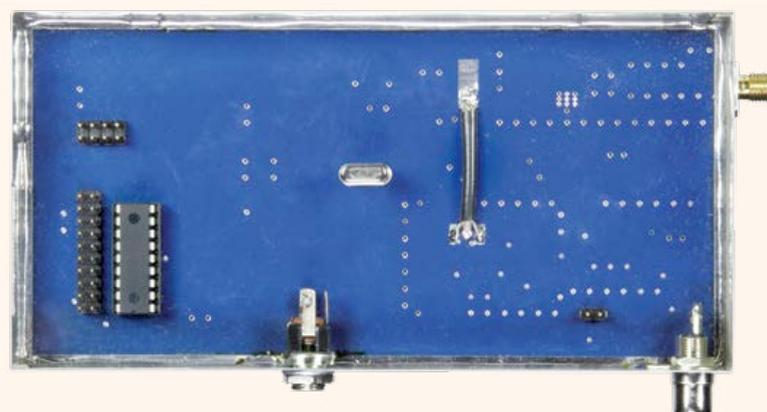
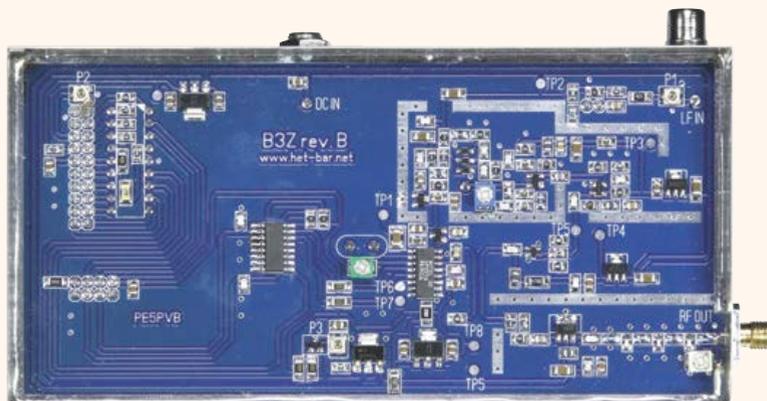
Pour recevoir le signal FM, on peut utiliser un scanner ou un récepteur FM à large bande. On se sert souvent d'un convertisseur pour changer les fréquences entre 430 et 440 MHz vers la gamme de 90 à 100 MHz p. ex., avant d'appliquer le signal à un récepteur FM normal.

Le schéma

C'est un oscillateur Colpitts qui délivre le signal à radiofréquence. Doté d'un MOSFET à double grille, T2, il oscille directement à la fréquence de travail. Un Colpitts détermine la fréquence au moyen d'une combinaison d'un diviseur capacitif avec une bobine en parallèle, ce sont ici C24, C25 et L4. La bobine est constituée d'un bout de câble coaxial qui fonctionne en principe comme une ligne triplaque (*stripline*). L'utilisation de câble souple, comme le RG174, évite considérablement les perturbations dues aux vibrations mécaniques. Pour obtenir une oscillation, il faut une réaction, un couplage en retour (*feedback*) assuré par R25. Vous verrez couramment que sur un Colpitts, la réaction est directe, sans résistance, mais en pratique, il apparaît qu'une résistance apporte plus de stabilité et une onde plus propre. Il y a, pour amener l'oscillateur dans la gamme voulue, le condensateur d'ajustage TR2. Les diodes D3 et D4 sont à capacité variable commandée en tension. On obtient ainsi un réglage électrique dans une plage d'à peu près 30 MHz.

Conseils pour la construction

- Utilisez uniquement des condensateurs à la céramique, même au-dessus de 1 μ F.
- Résistances et condensateurs 0805, parfois même 1206.
- Si vous fabriquez le circuit imprimé vous-même, prévoyez assez de traversées. Elles sont très importantes, surtout en HF.
- Les selfs doivent être exemptes de ferrite, surtout dans l'oscillateur. Les ferrites produisent du bruit, c'est connu.
- L'ensemble doit être construit dans une boîte en fer. Le format du circuit imprimé respecte les dimensions standardisées.
- Commencez par la boîte, les trous pour le continu, la BF, la HF. Placez alors le circuit imprimé nu dans la boîte et soudez-y les deux faces sur tout le pourtour. N'implantez les composants que quand la platine est complètement installée dans le blindage.



- Utilisez une tension d'alimentation bien propre comme source pour l'émetteur. Certaines alimentations à découpage laissent des parasites dans la tension de sortie qui sont directement audibles sous forme de bruit et de pics. Une alimentation conventionnelle reste la meilleure solution.

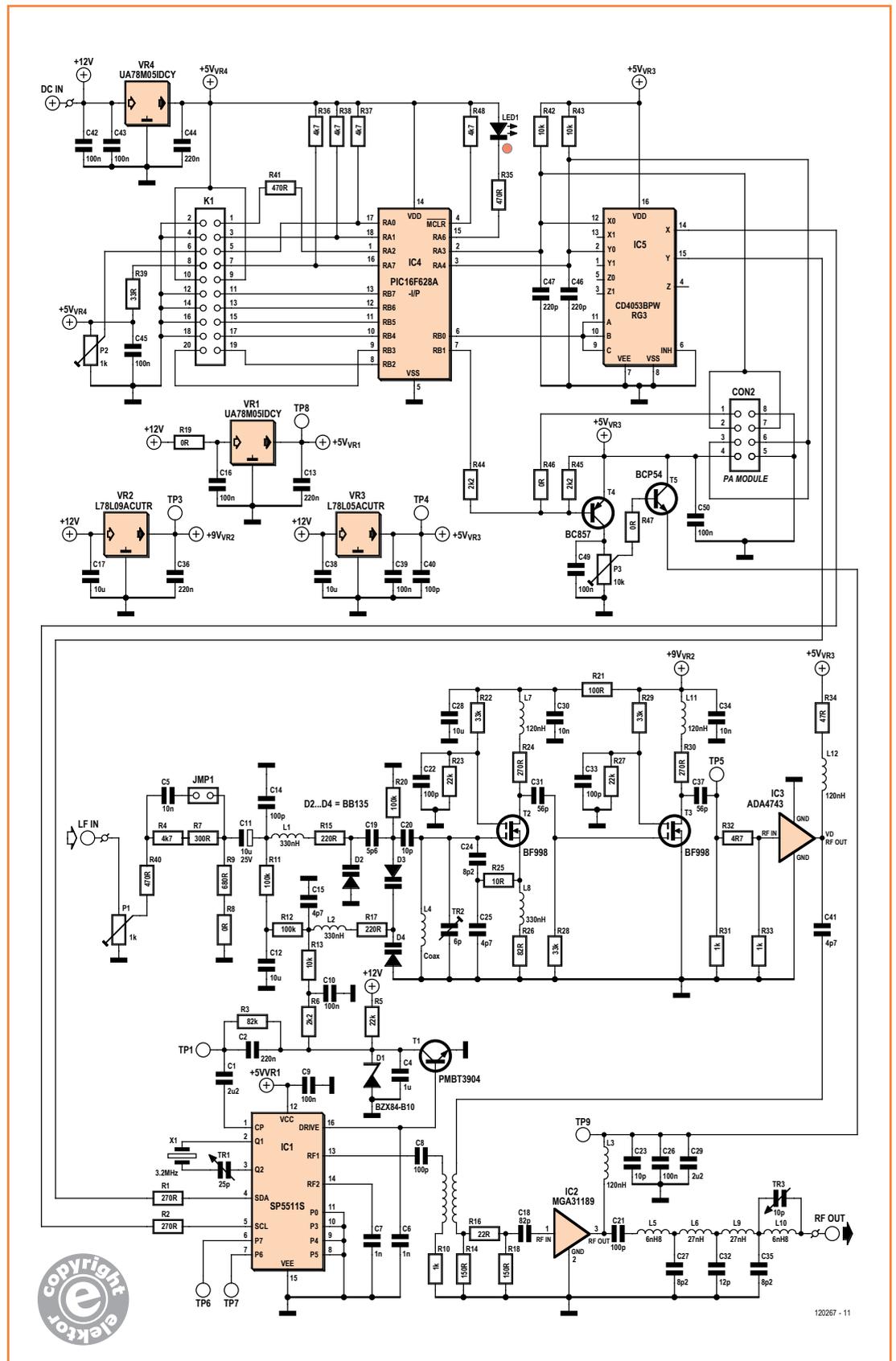


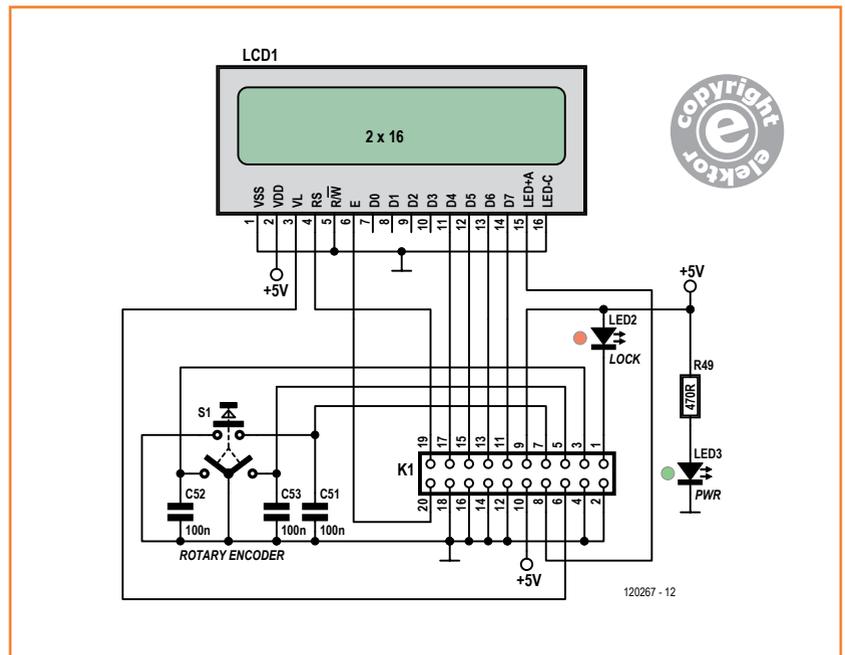
Figure 1. Schéma du circuit imprimé principal avec le contrôleur et la partie HF.



Comme on travaille en FM, la fréquence de l'onde porteuse doit varier au rythme du signal à basse fréquence. C'est la diode à capacité variable D2 qui s'y emploie. Tout changement de tension sur elle influence la capacité du réseau LC, donc la fréquence. Pour que le résultat soit aussi linéaire que possible, la modulation est obtenue avec une autre diode que celle qui sert à faire l'accord. Une grande différence de profondeur de modulation existe selon la hauteur de la basse fréquence, aussi faut-il prévoir une polarisation auxiliaire de la diode par un diviseur à haute impédance, R11 + R12, auquel est adjoint C12 pour plus de stabilité. L1 réduit la charge sur la ligne capacitive de modulation, pour éviter tout décrochage. C14 découple l'entrée de modulation des signaux à haute fréquence pour empêcher que la suite du montage n'ait une influence sur le rapport LC du circuit oscillant. Enfin, C5, R4, R7, R8 et R9 constituent un circuit de préaccentuation pour une utilisation éventuelle. On peut augmenter la valeur de R8 pour diminuer son effet. La valeur inscrite produit une préaccentuation de 50 µs.

En sortie de l'oscillateur, c'est un étage tampon, lui aussi équipé d'un MOSFET à double grille, T3, qui assure l'uniformité de la charge, gage de la stabilité du signal HF.

Ce signal d'environ 1 mW est alors atténué pour l'adapter à l'amplificateur suivant. Ces derniers temps, le marché a été inondé d'un grand assortiment de MMIC. Pareille puce, au nom dérivé de **monolithic microwave integrated circuit**, est destinée à l'amplification de signaux à très haute fréquence, d'où le terme micro-ondes. L'avantage majeur de ce genre de puces est d'assurer l'amplification du signal HF en toute simplicité, sans circuits accordés, de quoi éviter les circuits imprimés complexes, gagner de la place et réduire sensiblement le prix de l'ensemble. J'ai choisi des MMIC d'*Avago Technologies* qui répondent parfaitement à mes désirs et sont, de plus, aisément disponibles. Le premier, IC3, un ADA4743, amplifie le signal HF jusqu'à 40 mW. Puis il subit une atténuation pour l'appliquer à un MGA31189 (IC2), un MMIC capable de fournir une puissance de 250 mW. Réduire la tension d'alimentation de IC2, fournie par T4/T5, à 4,2 V maximum, ramène la puissance de sortie à quelque 120 ou 130 mW, largement assez pour attaquer l'étage final, la plupart du temps un module de puissance HF.



On fait appel à une boucle à phase asservie (PLL) pour rester exactement réglé sur la fréquence voulue. Par cette technique, on compare en permanence la fréquence d'émission, abaissée par un diviseur programmable, à une fréquence de référence. En cas de dérive constatée, une tension de commande la corrige jusqu'à ce que le signal d'émission divisé se retrouve en phase avec le signal de référence. Cette tension de commande est appliquée aux deux diodes à capacité variable D3 et D4 après filtrage dans un filtre de boucle, lequel est absolument indispensable pour empêcher une correction trop brutale dans la PLL. L'émission se fait toujours en FM et la fréquence varie en fonction de la modulation BF introduite. Si le rattrapage de phase dans la boucle est trop rapide, cela aura inévitablement des conséquences sur la qualité de la modulation. Il importe donc de trouver un juste milieu entre la largeur de bande maximale, environ 180 kHz, et la stabilité du signal. Comme PLL, j'ai choisi pour ce projet une puce (IC1) largement disponible et d'un prix abordable, du type SP5511 ou TSA5511. Elle était prévue à l'origine comme réglage des tuners TV (satellite). On en trouve souvent dans les anciennes télévisions, les enregistreurs vidéo ou les récepteurs de satellite. Le plus petit pas de fréquence que cette PLL peut effectuer est de 50 kHz avec l'aide d'un quartz de 3,2 MHz. Mon but était cependant d'obtenir une résolution de 25 kHz. Solution simple : faire

Figure 2. Le tableau de bord se compose d'un codeur rotatif et d'un écran LCD de 2 x 16 caractères.

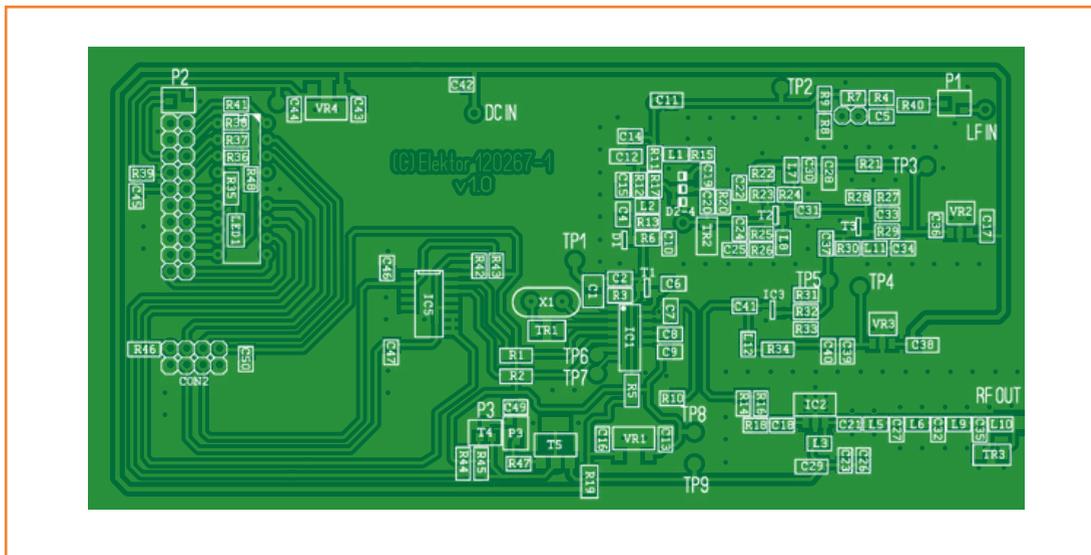


Figure 3.
Le circuit imprimé principal porte des composants sur les deux faces.

opérer la PLL sur l'harmonique 2 de l'émetteur. Ce signal est prélevé entre IC3 et IC2 par couplage inductif.

C'est le microcontrôleur IC4, un PIC 16F628A, qui commande la PLL par l'intermédiaire d'un bus I²C. Un gros désagrément de cette puce PLL est que de la diaphonie se produit entre le bus I²C et la pompe de charge. Il en résulte des bruits audibles au rythme des données I²C. Mon idée était d'ajouter éventuellement un étage final ou d'autres matériels à commander par le même bus I²C. Ces données, qui ne sont alors pas destinées à la PLL, sont quand même entendues dans la BF. J'ai résolu la question dans ce projet en plaçant un HEF4053, IC5, entre le bus I²C et la puce PLL. Au moment où les données ne concernent pas la PLL, la ligne I²C vers la PLL est interrompue. Une solution tout à la fois bon marché et efficace.

L'accord sur la fréquence d'émission s'effectue à l'aide d'un codeur rotatif avec bouton-poussoir incorporé. Il se trouve sur un circuit imprimé séparé (schéma à la **figure 2**) en compagnie de deux LED et d'un écran. Il suffit de sélectionner la fréquence et d'appuyer sur le bouton pour confirmer. Les données acquises sont alors envoyées à la puce PLL et son état est l'objet d'un suivi. Dès que la PLL annonce qu'il n'y a plus de différence de phase entre le signal de référence et celui d'émission, celui-ci est transmis au microcontrôleur qui allume la LED2, indiquant l'accrochage. Ce n'est qu'alors que le dernier MMIC est mis sous tension, de manière à

ce que l'émetteur ne fournisse de la puissance HF que quand il est exactement accordé sur la fréquence réellement choisie. On règle la tension du dernier MMIC avec P3 qui permet d'amener sans à-coup la puissance de l'émetteur à 1 mW environ. C'est ce qui le rend universellement adaptable aux différentes sortes d'étages finaux. Le connecteur CON2 est une embase à 8 broches qui permet de commander éventuellement de l'appareillage externe à partir du microcontrôleur, un étage final, par exemple.

Quand j'ai entamé ce projet, je n'étais pas encore très à l'aise avec la programmation des microprocesseurs. J'ai commencé avec le simulateur IDE de PIC. C'est un logiciel très simple pour programmer, il utilise une variante du Basic. Ce qu'il a de chouette, c'est qu'il dispose d'un simulateur complet, vous pouvez parfaitement simuler le logiciel avant de le programmer dans le contrôleur. En revanche, il n'a que des possibilités limitées de calcul. Pour faire simple, j'ai choisi de travailler avec deux compteurs indépendants. L'un pour la mesure de la fréquence, l'autre pour le dividende. En d'autres mots, si vous programmez, sur l'écran, l'émetteur 25 kHz plus haut, le compteur augmente de « 1 » le dividende. Aux extrémités de la bande, à 430 et 440 MHz, le compteur pour le dividende est bloqué à une valeur programmée d'avance.

Vous trouverez sur le site d'Elektor [1] toutes les informations utiles pour ce projet : dessins des circuits imprimés, liste des composants, codes

source et hexadécimal pour le contrôleur.

Le maniement

La commande de tout l'émetteur s'opère à l'aide d'un seul bouton rotatif à poussoir. Il suffit de faire tourner son axe pour sélectionner une nouvelle fréquence. Vous voyez à droite, dans le haut de l'écran : « TUNE ». Dès l'appui sur le bouton du codeur, la nouvelle fréquence est mise en mémoire et va être accordée. Dès que la PLL a accroché, la LED Lock s'allume et l'étage de sortie est mis en service. Si la PLL n'arrive pas à s'accrocher, la LED reste éteinte et le message PLLERR s'affiche. Si aucune donnée de la PLL n'est reçue, l'avertissement I2CERR apparaît. Dans un cas comme dans l'autre, le microcontrôleur recommence sans relâche à programmer la PLL jusqu'à obtenir l'accrochage.

La construction

En fabrication de circuits à haute fréquence, à fortiori en très haute fréquence, le mot d'ordre est de faire au plus court avec les connexions. Et aussi de garder la masse la plus « froide » possible. Dans mon projet de circuit imprimé – son tracé est représenté en réduction à la **figure 3** – vous y verrez quantité de traversées (*vias*). Si vous avez l'intention de le graver vous-même, pensez qu'il est indispensable de les réaliser tous. Le circuit imprimé est conçu pour être monté dans un modèle courant de blindage, une boîte en fer de 74 x 148 mm, à laquelle les deux faces de la carte sont soudées sur tout le pourtour. C'est le contact de masse optimal. L'insertion dans une boîte en fer et la soudure périphérique sont d'une grande importance, un boîtier fait soi-même en circuit imprimé ne convient pas pour ce circuit.

Voici la manière dont je procède d'habitude pour la construction. D'abord fixer la platine sans composants dans la boîte en fer et l'équiper des éléments nécessaires du châssis. Pour le raccordement de l'antenne, il est préférable d'utiliser un composant de châssis à monter en surface. C'est un modèle plat qui se soude le mieux sur le fer. Il ne faut dès lors pas de boulons. Vous êtes libre de l'emplacement des connexions d'alimentation et de l'entrée de signal, pour laquelle la prise tulipe audio pour châssis a généralement les faveurs.

Tous les CMS sont regroupés sur une face du circuit imprimé, de l'autre côté, il y a trois embases

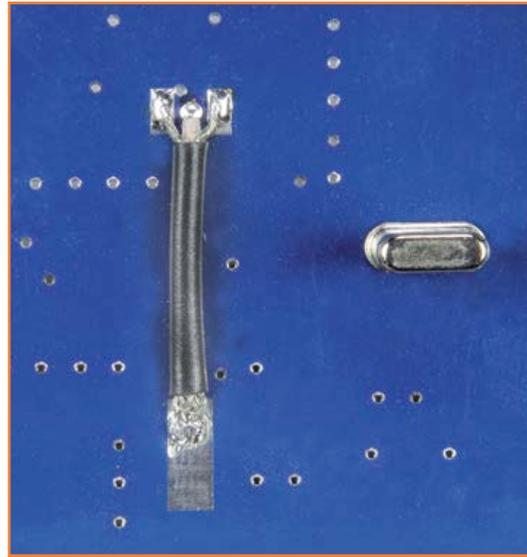


Figure 4. Du côté pile de la carte, on trouve quelques embases, le microcontrôleur et un bout de câble coaxial qui fonctionne comme bobine d'oscillateur.

à picots, la ligne triplaque et le microcontrôleur. La **figure 4** vous montre le détail de la ligne triplaque. Elle est faite d'un morceau de câble coaxial flexible de 4 cm de long, soudé au bout du circuit imprimé. D'un côté du câble, l'âme est soudée au blindage et au circuit imprimé. À l'autre bout, il faut en souder le blindage plié en deux sur le plan de masse (comme sur la photo), l'âme est soudée dans le trou de l'œillet.

Construisez le circuit bloc par bloc, en commençant par l'oscillateur. Vérifiez qu'il opère entre 400 et 500 MHz, à l'aide d'un fréquencemètre ou d'un analyseur de spectre. Montez ensuite le tampon et les deux étages amplificateurs MMIC. Utilisez une connexion temporaire pour l'alimentation du dernier MMIC pour qu'il travaille de concert. Vérifiez alors le signal HF, avec un millivoltmètre HF par exemple.

Quand la partie à haute fréquence fonctionne, construisez le circuit autour de la PLL et du microcontrôleur, pour lequel vous utilisez un support à 18 broches. Programmez le microcontrôleur avant de l'insérer dans le support.

Le tableau de commande est sur un circuit imprimé séparé, il est illustré à la **figure 4**. Sa construction n'est pas critique, tous les composants, à l'exception d'un seul, s'implantent côté composants. Un connecteur à 16 broches et son correspondant servent au raccordement de l'affichage. Sur la face arrière du circuit imprimé, il y a une embase à 20 contacts pour relier le cir-

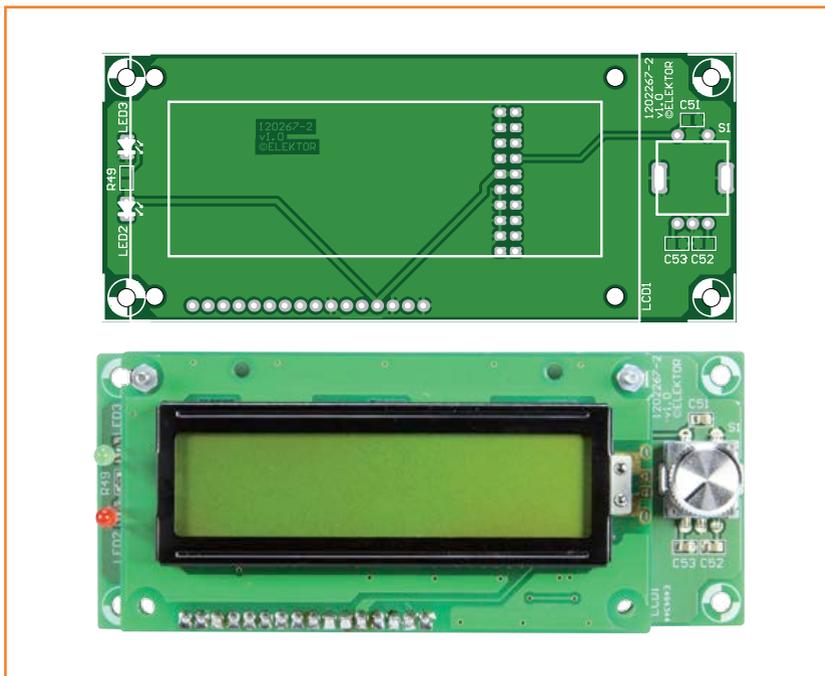


Figure 5.
Le circuit imprimé du tableau de bord se construit en un rien de temps. L'embase à 2 x 10 broches pour la liaison à la carte principale est installée à l'arrière.

cuit imprimé principal à K1 par un câble plat à 20 conducteurs.

Le réglage

Raccordez l'émetteur et l'affichage à l'alimentation. Réglez P2 pour le contraste maximum de l'afficheur.

Accordez l'émetteur sur 435 MHz. À l'aide d'un tournevis en céramique ou en plastique, faites tourner TR2 jusqu'à ce que l'émetteur soit sur 435 MHz. Mesurez la tension sur TP1 et tournez TR2 jusqu'à ce que la tension atteigne environ 6 V. Vous pouvez à présent régler exactement l'émetteur sur 435,0 MHz par TR1 à l'aide d'un fréquencemètre ou d'un analyseur de spectre.

Réglez alors P3 sur le maximum de puissance d'émission. Il faut encore ajuster le creux de l'atténuateur, vous aurez besoin pour cela d'un analyseur de spectre. Mesurez le deuxième harmonique, sur 870 MHz, et réglez TR3 sur le minimum de puissance. Si vous n'avez pas d'analyseur de spectre, réglez TR3 sur le maximum de puissance de sortie avec un millivoltmètre HF. Tenez compte du fait que le filtrage n'est pas optimal. Il faudra s'en occuper dans l'étage final, placé derrière l'émetteur.

Vérifiez sur l'émetteur s'il travaille bien sur 430

et 440 MHz. Installez un cavalier sur la position de préaccentuation et allumez une source audio, un lecteur de CD par exemple. Réglez alors P1 pour un maximum de puissance audio, avant que la distorsion ne se manifeste.

Vous êtes maintenant en possession d'un émetteur FM bon pour le service dans la bande des 70 cm.

Tout radioamateur ne dispose pas nécessairement de l'expérience suffisante dans le soudage manuel de CMS. Je suis prêt, si l'intérêt suscité est suffisant, à construire et régler pour vous un certain nombre d'émetteurs en échange d'une rémunération raisonnable, auquel cas, vous pouvez me contacter via le lien [2].

(120267 – version française : Robert Grignard)

Liens

[1] www.elektor.fr/120267

[2] pe5pvh@het-bar.net

Réglages

P1	profondeur de modulation BF
P2	contraste LCD
P3	puissance de sortie
TR1	ajustage de l'oscillateur de référence
TR2	réglage brut de fréquence d'émission
TR3	creux pour 2e harmonique

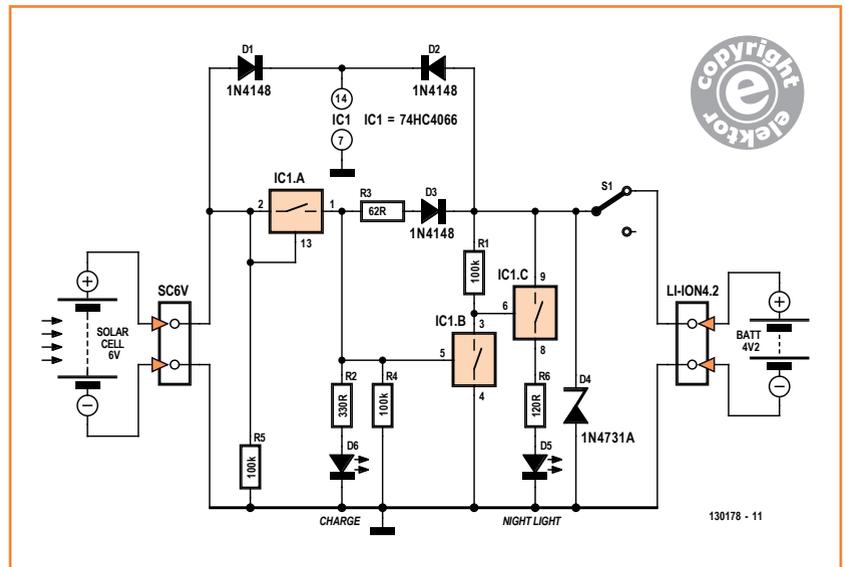
Points de mesure

TP1	tension de la PLL
TP2	préaccentuation de la modulation
TP3	tension d'oscillateur (9 V)
TP4	tension de commande (5 V)
TP5	tension d'étage final (0 à 4 V)
TP6	fréquence PLL après division
TP7	fréquence de référence PLL

veilleuse solaire avec accu Li-ion de secours

Cette veilleuse est alimentée par une pile solaire possédant une tension de crête d'environ 6 V, et par un accu Li-ion dont la tension est comprise entre 3,7 et 4,2 V. Trois (des quatre) interrupteurs d'un 74HC4066N (IC1) contrôlent le fonctionnement du circuit. IC1 est alimenté par celle des deux diodes D1 ou D2 qui est reliée à la source de plus forte tension, donc opère entre 3,7 V et 6 V. En journée, la tension délivrée par la pile solaire atteint une valeur crête type de 6 V, et IC1a est fermé par le niveau haut de son entrée de commande (broche 13). Environ 10 mA chargent alors l'accu via R3 et D3 en série. La LED D6 s'allume pour indiquer que la charge est en cours. L'interrupteur IC1b est lui aussi fermé, ce qui entraîne l'ouverture d'IC1c et laisse hors tension la LED D5.

Lorsque la tension fournie par la pile solaire ne dépasse plus 1/3 de la tension d'alimentation d'IC1, c.-à-d. devient inférieure à environ 1,3 V, IC1a s'ouvre et la LED de témoin de charge s'éteint. La tension de l'entrée de commande d'IC1b devient nulle et l'interrupteur s'ouvre. C'est alors IC1c qui se ferme et connecte la LED « veilleuse » à l'accu via R6, qui limite le courant de LED entre 10 et 13 mA. Utilisez la couleur de LED de votre choix, celle du prototype est verte. Le régime de charge ainsi que la luminosité des



LED se règlent en modifiant les valeurs de R3, R2 et R6 ; veillez à ce que le courant qui traverse les interrupteurs ne dépasse pas 20 mA. La zener D4 protège l'accu contre les tensions de charge excessives.

Michael A. Shustov
(Russie)

Ouvert, l'interrupteur S1 évite la décharge de l'accu lorsque le circuit n'est pas utilisé.

(130178 – version française : Hervé Moreau)

Liste des composants

Résistances

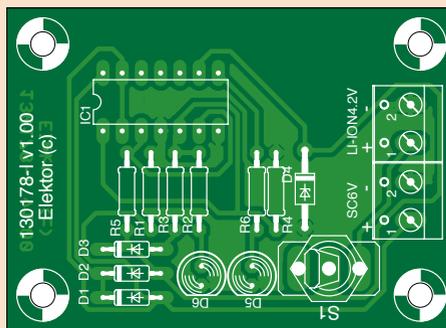
R1, R4, R5 = 100 k Ω , 1 %, 0,25 W
R2 = 330 Ω , 1 %, 0,25 W
R3 = 62 Ω , 1 %, 0,25 W
R6 = 120 Ω , 1 %, 0,25 W

Semi-conducteurs

D1, D2, D3 = 1N4148
D4 = diode zener 1N4731A (4,3 V)
D5 = LED, 5 mm, couleur au choix
D6 = LED, rouge, 5 mm
IC1 = 74HC4066

Divers

S1 = interrupteur à bascule, Newark/Farnell 1310879
circuit imprimé 130178-1

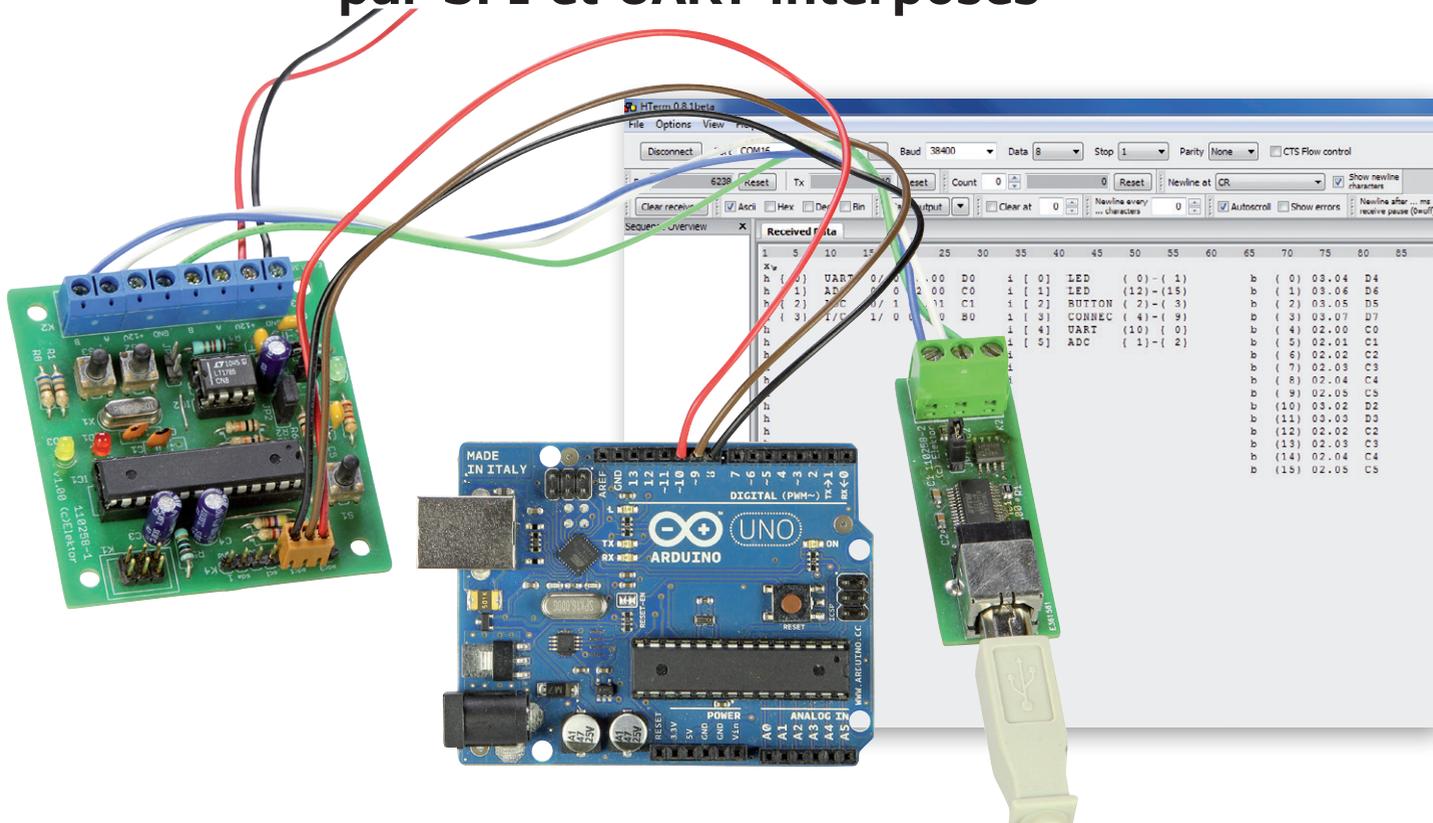


connecteurs marqués SC6V et LI-ION 4,2 V = connecteurs encartables, au pas de 5 mm



pilotez vos cartes depuis le PC

par SPI et UART interposés



Jens Nickel

Comment commander des cartes électroniques depuis un terminal sur PC par un protocole textuel ? Le micrologiciel nécessaire est facile à programmer grâce à EFL, notre bibliothèque de micrologiciels embarqués présentée le mois dernier. C'est indépendant du type de contrôleur comme de l'interface de bus utilisée. Et le protocole présenté ici sert aussi bien pour le développement et les essais.

Dans le précédent numéro, nous avons présenté la bibliothèque modulaire de micrologiciels embarqués EFL [1]. Elle s'adresse aussi bien aux débutants qu'aux concepteurs chevronnés et leur permet d'assembler vite le code nécessaire à leurs projets embarqués tout en restant indépendants du matériel, ce qui permet la portabilité de carte à carte comme d'un contrôleur à l'autre. C'est une couche matériel

parfaitement encapsulée qui assure la portabilité. Elle contient un fichier de code pour la carte et un autre pour le microcontrôleur. Mais la modularité d'EFL va encore plus loin. Les bibliothèques de protocole sont programmées indépendamment du canal de transmission utilisé. Peu importe que les commandes et les données passent par UART, RS232, RS485 ou TCP/IP/Ethernet. Quand on veut changer de

canal de communication dans l'application, il n'y a que quelques lignes de code à changer. C'est ce que nous allons démontrer dans cet article et en même temps présenter un protocole adapté aux tâches de commande simples comme aux fins de développement.

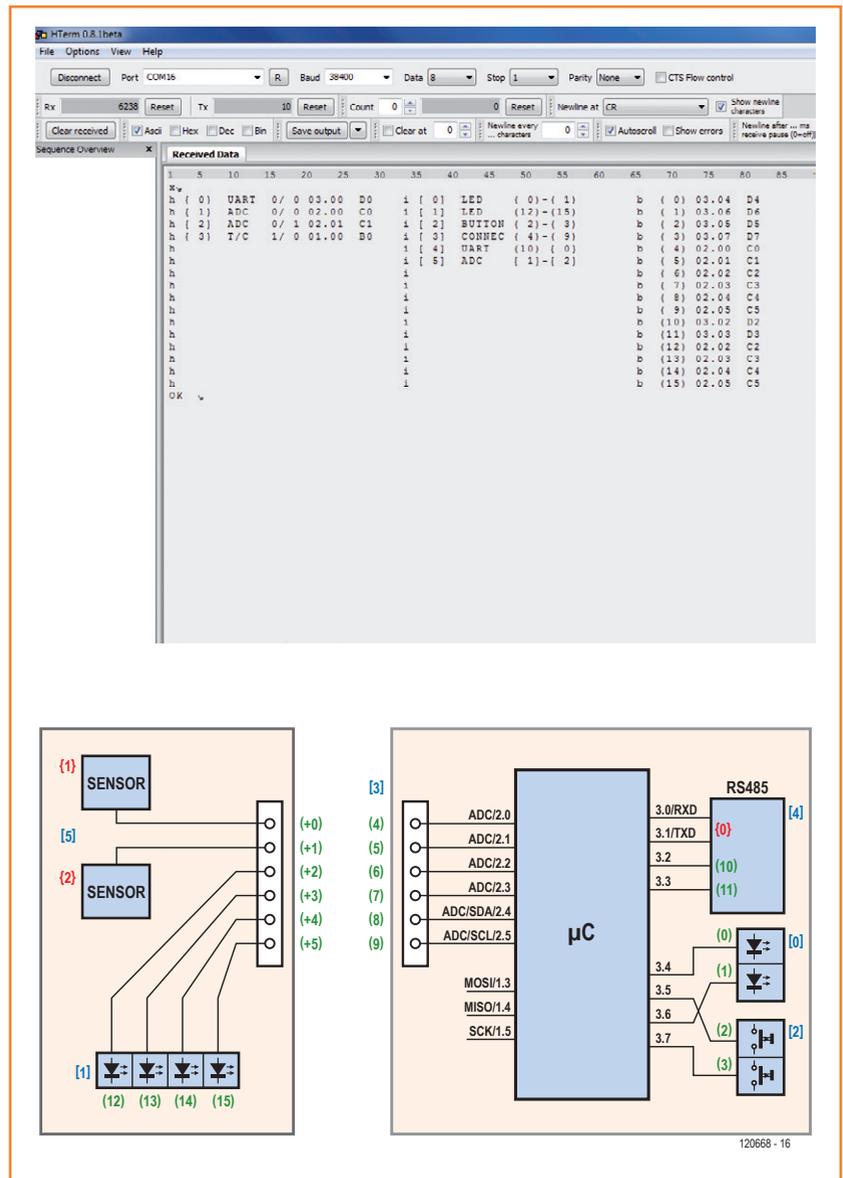
Miniprotocole

Lors de l'élaboration du code, des tests simples, comme la commutation d'une LED ou l'interrogation d'une entrée numérique, sont très utiles. Souvent, une carte à microcontrôleur n'a même pas de bouton pour faire un test. Il est alors intéressant de commander la carte à partir d'un PC à travers une interface sérielle. Si l'on se limite aux caractères ASCII, on peut s'installer confortablement aux commandes par l'intermédiaire d'un programme de terminal, un logiciel spécial pour PC n'est même pas nécessaire.

Avec notre miniprotocole, appelé `BlockProtocol`, vous pouvez, de votre terminal sur PC, mettre des broches au niveau haut ou bas. Si vous commandez une broche à laquelle est reliée une LED, vous verrez directement le résultat, sinon, mettez-y un multimètre ou un oscilloscope. En outre, il y a des commandes simples pour sonder des états numériques ou les valeurs d'un CAN depuis des entrées analogiques (voyez l'encadré `BlockProtocol`). Vous pouvez aussi étendre ou adapter de diverses manières le code du module de la bibliothèque correspondante. Qui sait, certains lecteurs pourraient être tentés de s'inspirer des protocoles d'Andreas Eppinger [2] et Uwe Altenburg [3] pour offrir davantage de fonctions.

Map, Blocks et Boardpins

Spécialement en vue du développement d'EFL, le `BlockProtocol` dispose de la commande « x ». La carte renvoie alors au PC un vidage (*dump*) du tableau EFL. La couche matériel en a besoin pour identifier, lors des appels de fonction, les broches du contrôleur et le registre approprié. Ces valeurs sont spécifiques de la carte utilisée et éventuellement d'une extension. L'écran de la **figure 1** montre ce qui s'affiche sur le terminal à propos des cartes dont les diagrammes fonctionnels sont en dessous. À gauche, on voit le « Map » avec les fonctions du contrôleur qui ont été initialisées par *Board-Init*, tels que UART et CAN. Au milieu, la liste des blocs périphériques et à droite, une liste de



Boardpin, les broches de carte. Une information précieuse pour quiconque programme ou adapte un fichier de carte EFL pour une nouvelle carte. De plus amples informations sur le contenu d'EFL sont dans le document annexe sur EFL à télécharger sur [1].

Commander par UART

Montrons par un exemple la facilité d'installer le protocole dans son application personnelle de contrôleur. Nous utilisons le nœud expérimental bien connu de la série ElektorBus, une carte très compacte équipée d'un ATmega328. Ces nœuds sont comme toujours reliés par RS485 à un convertisseur USB/RS485 pour accéder

Figure 1. Sortie des tableaux EFL dans un programme de terminal. À gauche, on voit le Map avec les fonctions utilisées du contrôleur (chiffres rouges dans les diagrammes), au milieu le tableau des blocs (chiffres bleus) et à droite les broches de cartes (chiffres verts).

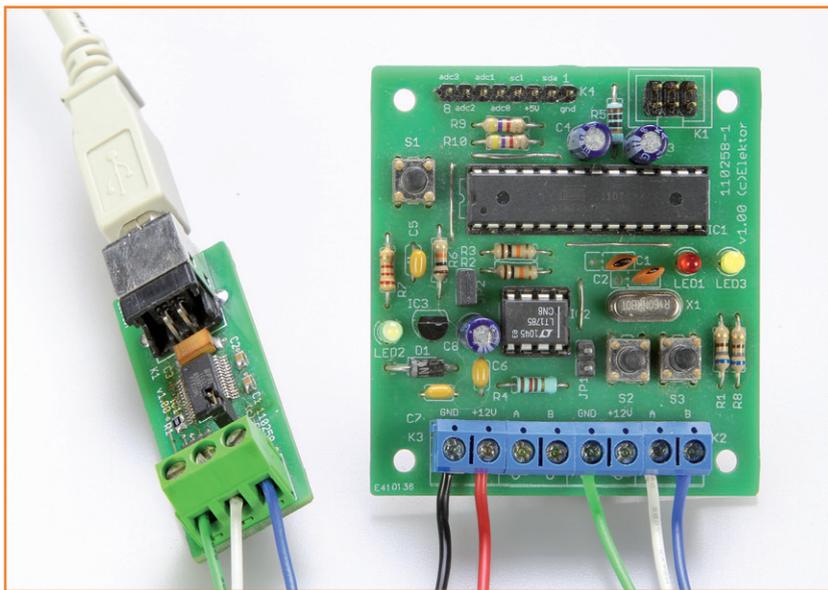


Figure 2. Nous pouvons piloter la petite carte à microcontrôleur avec un simple protocole de texte ASCII par l'intermédiaire de l'interface UART/RS485.

au PC comme à la **figure 2**. Mais maintenant le *BlockProtocol*, basé sur du texte, remplace le protocole ElektorBus.

Le micrologiciel adéquat pour le contrôleur est disponible sur [4], en plus l'application en base de code EFL sur [5], dont une synthèse se trouve dans le document annexe. On ouvre le projet en Studio Atmel d'un double clic sur *ExperimentalUART.atsIn*, voyez-en l'écran à la **figure 3**. On reconnaît du côté droit les fichiers inclus. Les fichiers *Controller.h/.c* et *Boad.h/.c* constituent la couche matériel. Il s'agit du même code que dans le logiciel d'exemple d'EFL

Listage :
structure de base d'une application EFL

```
int main(void)
{
    Controller_Init();
    Board_Init();
    Extension_Init();

    ApplicationSetup();

    while(1)
    {
        ApplicationLoop();
    }
};
```

du dernier numéro, avec la même carte et le même contrôleur. On trouve dans le dossier *Libraries* les fichiers *UARTInterface.h/.c*. C'est la bibliothèque pour l'interface UART sur la carte, dans le cas présent sous la forme du pilote de notre RS485, c'est la couche physique de la communication. En outre vient un module pour le protocole utilisé : *BlockProtocolEFL.c/.h*. Les deux modules de bibliothèque forment ensemble une interface de code défini, ce qu'on appelle une interface logicielle, que nous allons examiner en pratique.

Code court

Le code proprement dit de l'application, présent comme toujours dans le fichier principal, nous devons le relier aux bibliothèques :

```
#include "UARTInterfaceEFL.h"
#include "BlockProtocolEFL.h"
```

La fonction principale est toujours construite de la même façon dans tout projet EFL, voyez le **listage 1**. Dans la fonction d'initialisation de l'application, qui est chaque fois appelée au début de l'application, nous initialisons les bibliothèques :

```
UARTInterface_LibrarySetup();
UARTInterface_SetBaudrate(0, 38400);
BlockProtocol_LibrarySetup(UARTInterface_Send, 0, UARTInterface_GetRingbuffer(0));
```

La deuxième ligne fixe à 38400 bauds le débit binaire du Block #0 de *UARTInterface*, auquel est raccordé notre pilote RS485.

La troisième ligne demande une explication supplémentaire. On fait savoir à la bibliothèque *BlockProtocol* qu'elle doit utiliser la fonction *UARTInterface_Send* quand il y a des données à transmettre depuis la carte. Le deuxième paramètre est le numéro du bloc d'interface UART à utiliser. Sur le nœud expérimental, nous ne disposons que d'une interface UART. Le troisième paramètre est un pointeur vers la mémoire tampon en anneau qui emmagasine les octets reçus. Avec le pointeur sur l'anneau correspondant, la fonction *UARTInterface_GetRingbuffer(0)* nous enverra en retour la valeur qui contient les octets reçus. Cette fonction est également implémentée dans la bibliothèque *UARTInterface*.

Dans la fonction *ApplicationLoop* régulièrement appelée, une seule ligne suffit :

```
BlockProtocol_Engine();
```

C'est la fonction principale de la bibliothèque *BlockProtocol*. Elle vérifie si des caractères venus du PC ont atterri dans le tampon en anneau. Dès qu'un <CR> (ASCII 13) arrive, il est interprété comme la fin d'une commande. Elle est alors exécutée et une réponse composée : soit Ok, une valeur HIGH ou LOW ou encore la sortie de variables EFL sous forme de tableau.

Essayons un peu...

Après avoir compilé le programme et l'avoir flashé dans la carte, essayons-le avec encore un programme de terminal comme HTerm [6]. Réglons d'abord le port COM et le débit binaire. Dans le domaine d'entrée du programme (Input Control), nous devons faire en sorte qu'une action sur la touche Entrée écrive un <CR> derrière les signes introduits, avant que la chaîne complète ne soit envoyée (cf. **figure 4**).

Nous essayons d'abord au niveau inférieur, avec une indication directe de la broche de port du contrôleur. L'examen du schéma du nœud expérimental [7] nous apprend que la LED rouge est reliée à la broche de port PD4. Le port D sur le contrôleur AVR correspond au PortIndex 3. Alors, écrivons dans le programme de terminal :

```
p 3 4 + <ENTER>
```

La LED rouge sur la carte doit s'allumer. Avec « p 2 3 + » ou « p 2 0 - », nous pouvons par exemple actionner la broche PC0 sur le connecteur d'extension. Si nous branchons une carte d'extension, par exemple notre circuit imprimé de capteurs et LED de l'épisode précédent, ou bien la carte à relais de la série ElektorBus [8], nous aurons déjà un petit système à télécommander depuis le PC.

À la place du programme de terminal, nous pouvons aussi utiliser un logiciel pour PC qui donne simplement les signes « p 2 0 + <CR> » sur le port COM adéquat pour activer le relais. Ou bien raccordons au nœud expérimental un smartphone via la carte de pont RS485/UART AndroPOD [9] et rédigeons une petite application Android. Comme base de départ,

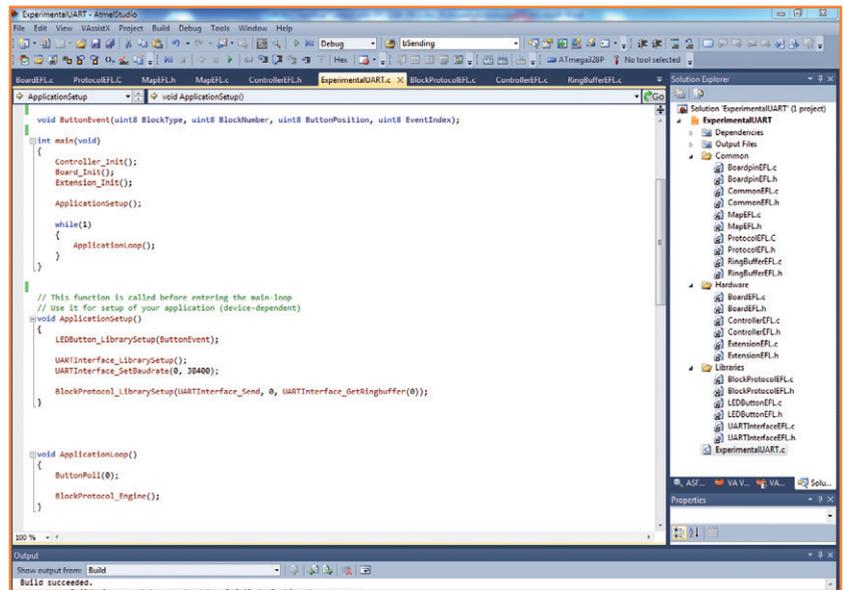
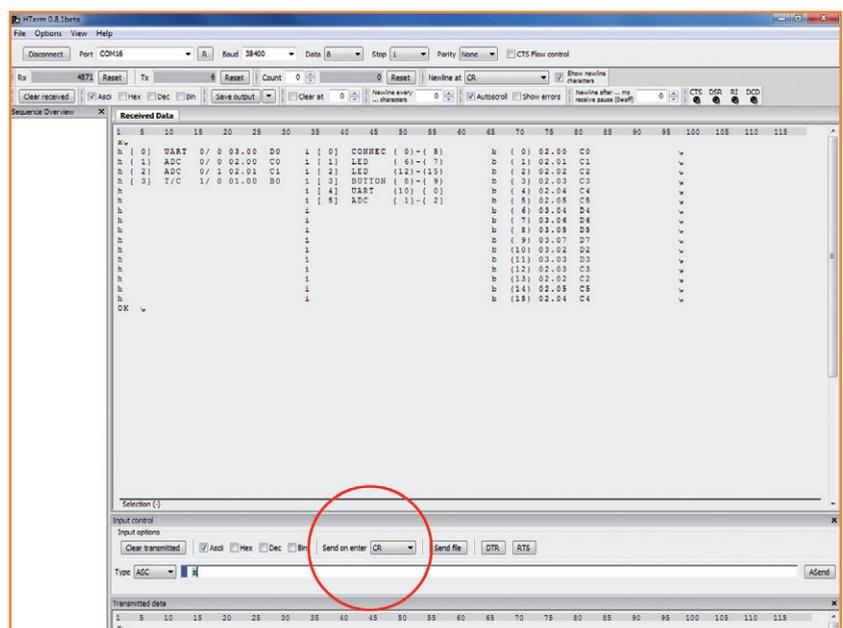


Figure 3. Le code pour commander toute son électronique par interface UART.

Au niveau supérieur

Pourtant, aller spécifier explicitement une broche de port ne cadre pas du tout avec la philosophie d'EFL. Si jamais on veut utiliser pour le système à télécommander une autre carte de contrôleur avec un câblage différent, il faudra modifier la chaîne de caractères dans le PC ou le logiciel Android.

Figure 4. Le programme de terminal HTerm permet d'ajouter un <CR> à la fin d'une chaîne de caractères à envoyer.



C'est le moment de faire appel au miniprotocole qui peut gérer la commande d'E/S numériques de manière indépendante du matériel employé, exactement comme nous l'avons fait dans le dernier numéro avec les lignes de code. Avec la chaîne de caractères

L 0 1 + <CR>

la LED 1 s'allume dans le bloc de LED #0 (c'est la jaune sur notre carte).

Et avec

B 0 0 ? <CR>

on demande l'état du bouton de test (le bouton 0 du premier bloc de boutons #0). Quant au relais, on l'active par

R 0 0 + <CR>

L'interface à 3 fils

La télécommande en question n'est possible que si, de l'extérieur, nous avons accès à un UART du contrôleur, par exemple si les broches RX/TX arrivent sur un connecteur. En outre, il faut disposer d'un fichier EFL pour contrôleur pratiquement achevé et qui nous donne les fonctions d'E/S et CAN en plus des fonctions UART.

Quand on tombe sur un contrôleur inconnu d'EFL (on voudrait par exemple développer un fichier EFL de contrôleur que d'autres lecteurs pourraient utiliser) il faut probablement commencer par les fonctions simples d'E/S. Après avoir étudié le cahier de caractéristiques, au moins pour savoir comment changer le niveau d'une broche ou le lire, que l'on dispose d'au moins trois broches d'E/S à usage général (*GPIO*), ce qui est normalement le cas sur la plupart des cartes, on a déjà fait un bon bout de chemin. À ce moment-là, on peut déjà piloter la carte au moyen du protocole évoqué plus haut. Comme canal de transmission, on peut alors utiliser une interface à 3 fils sur base du logiciel pour SPI. Une des lignes servira à l'horloge, une autre permettra aux octets de s'écouler du maître (le PC) vers l'esclave (la carte) et la troisième ligne fera de même dans l'autre sens.

Selon la spécification SPI, l'interface à trois fils permet aussi, tant au maître qu'à l'esclave, d'initier la communication. Pour ce faire, on met au niveau haut sa propre ligne de sortie et l'on attend que l'autre participant fasse de même avec la sienne. Alors, la conversation commence, pendant laquelle, toujours à la mode SPI, c'est le maître qui fournit l'horloge et les octets sont échangés dans les deux sens. Quand il n'y a plus de données à transmettre, on envoie le signal stop, <LF> = ASCII 10. Si les deux participants envoient un « 10 », la communication est terminée.

Bien sûr, tout cela, nous l'avons déjà encapsulé dans un module de la bibliothèque EFL du nom

Le BlockProtocol

Toutes les commandes commencent par un seul signe, ensuite un ou deux nombres décimaux. EFL en déduit la broche concernée. Le caractère final définit l'action.

x

Sort la carte des tableaux EFL, blocs, broches de carte (cf. document annexe sur EFL [1]).

*p x y +, p x y -, p x y ?, p x y #, p x y **

broche de contrôleur port x broche y : sur niveau haut, sur niveau bas, demande du niveau, demande de valeur CAN (sur une broche de CAN), clignoter à intervalle défini

*b x +, b x -, b x ?, b x #, b x **

broche de carte avec indice x dans le tableau des broches de carte : haut, bas, demande, valeur CAN, clignoter

*i x y +, i x y -, i x y ?, i x y #, i x y **

broche dans le bloc d'indice x dans le tableau des blocs, position de la broche y dans le bloc : haut, bas, demande, valeur CAN, clignoter

*C x y +, C x y -, C x y ?, C x y #, C x y **

connecteur x broche y : haut, bas, demande, valeur CAN, clignoter

*L x y +, L x y -, L x y ?, L x y **

bloc de LED x position de LED y : haut, bas, demande, clignoter

R x y +, R x y -, R x y ?

bloc de relais x position du relais y : haut, bas, demande

B x y ?

bloc de boutons x position du bouton y : demande

A x y #

bloc CAN x position de broche CAN y : demande de valeur CAN

*

arrêter de clignoter à intervalle défini

de *ThreeWireInterfaceEFL*. Aussi, rien de plus simple que d'adapter la communication à partir de l'interface UART vers le logiciel de l'interface SPI.

Les lignes de code pour la fonction d'initialisation de l'application deviennent :

```
ThreeWireInterface_LibrarySetup();
```

```
BlockProtocol_
LibrarySetup(ThreeWireInterface_Send, 0,
ThreeWireInterface_GetRingbuffer(0));
```

Avec quoi, nous communiquons à la bibliothèque *BlockProtocol* que les données devront être émises et reçues sur l'interface à trois lignes (on peut comparer la ligne avec le code ci-dessus pour *BlockProtocol_LibrarySetup*)

Dans la routine en boucle de l'application, il faut encore l'instruction :

```
ThreeWireInterface_Listen(0);
```

Elle vérifie que l'autre participant a déjà passé sa ligne au niveau haut. La communication peut alors commencer. Dans le cas de l'UART comme canal de transmission, cette fonction d'écoute n'est pas nécessaire, puisque le contrôleur reconnaît l'interruption quand des signes entrent et que les octets sont alors automatiquement placés dans le tampon en anneau.

Prise de contact avec Arduino

Comme carte de démonstration pour tout ceci, nous avons choisi l'Arduino Uno. Le code correspondant se trouve dans le projet *ArduinoUnoEFL* [4] [5]. Comme lignes GPIO pour donner accès à la carte, nous utilisons PB0, PB1 et PB2, disponibles sur le connecteur d'extension *Digital* sous les appellations *Digital8* à *Digital10*. Un petit connecteur de carte à 3 broches, tel que le Conrad 741221, rend bien des services (cf. **figure 5**).

Évidemment, il nous faut aussi une liaison au PC, mais quel utilisateur de PC n'a pas ce fameux câble à trois conducteurs ? Avec lui, nous relierons notre nœud expérimental à une passerelle qui convertit les données du canal de transmission à 3 fils vers UART/RS485 et retour. Le micrologiciel correspondant est bien entendu à télécharger de [4] et [5], le projet en Studio d'Atmel se trouve



Figure 5. Trois conducteurs nous donnent accès à la carte Arduino Uno.

dans le dossier *ExperimentalSPI*.

Pour la fonction de passerelle, il y a un module particulier en bibliothèque, il se nomme *OneToOneGateway*.

Par un appel de

```
OneToOneGateway_Engine();
```

à l'intérieur de la fonction *Application Loop*, qui est interrogée régulièrement pour savoir si une chaîne de caractères est arrivée dans le tampon en anneau, elle sera refermée par un <CR>. Si c'est le cas, la chaîne de caractères est alors retransmise chaque fois sur un autre canal de communication. Par la suite, on vérifie le tampon en anneau de cet autre canal de transmission pour obtenir les signes qui y sont entrés et ainsi de suite.

Il va de soi que le module de passerelle a été programmé pour être tout aussi indépendant du canal de communication choisi et pouvoir ainsi se plier à toute application possible de passerelle. La fonction de bibliothèque d'initialisation du *OneToOneGateway* prend en revanche deux paramètres triples qui déterminent quels canaux de communication seront couplés à la passerelle. Dans ce cas-ci, l'appel sera :

```
OneToOneGateway_
LibrarySetup(UARTInterface_Send,
0, UARTInterface_GetRingbuffer(0),
ThreeWireInterface_Send, 0,
ThreeWireInterface_GetRingbuffer(0));
```

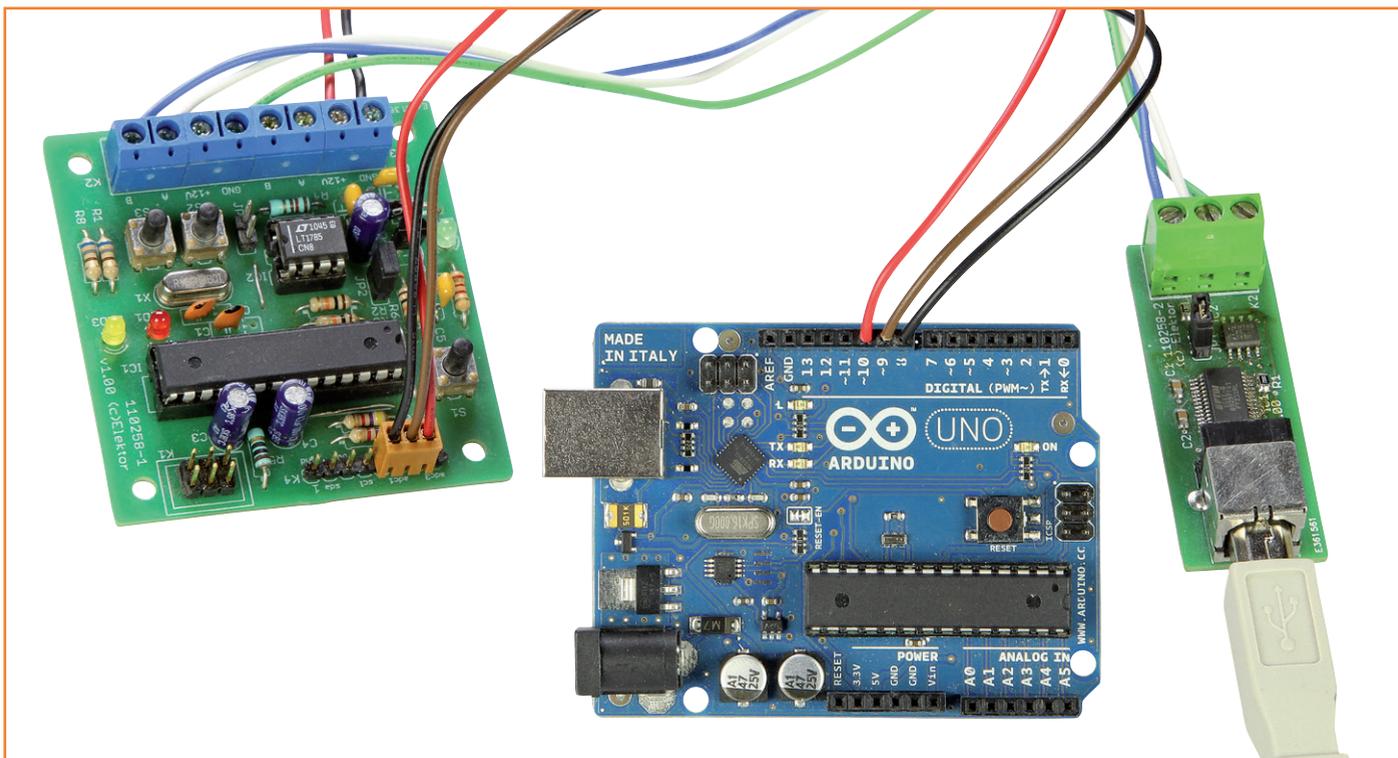


Figure 6.
Le nœud expérimental sert de passerelle entre l'interface à 3 fils et le UART/RS485.

Il ne nous reste plus qu'à raccorder le connecteur de carte à fils au connecteur d'extension du nœud expérimental (broches PC0 à PC2, cf. figure 6) et à flasher le fichier hexadécimal du nœud dans le contrôleur.

On peut encore adapter sur les deux cartes les broches utilisées pour la liaison à 3 fils. Cette interface forme chez chacun des participants un bloc périphérique propre et rappelons que le fichier de la carte encapsule le câblage entre les blocs périphériques et les broches du contrôleur. La configuration de l'interface à 3 fils, avec l'affectation des broches utilisées, se trouve toujours dans la fonction Board_Init dans le fichier BoardEFL.c.

Et ça marche !

Dans le programme de terminal, commençons par donner une commande « x ». Arduino transmet fidèlement par l'interface à 3 fils les variables EFL, ce qui, à 9600 bauds, est visiblement plus lent que précédemment. Avec l'instruction

```
C 0 13 + <CR>
```

on met au niveau haut la broche 13 du connecteur *Digital* et, comme sur la carte

Arduino Uno il y a une LED, on voit tout de suite le résultat.

Dans le prochain épisode, il sera question de nouveaux projets EFL. Si vous avez des suggestions à formuler ou si vous désirez apporter votre contribution, vous pouvez, comme d'habitude, nous les communiquer sur redaction@elektor.de.

Tous les nouveaux codes EFL sont disponibles sur le site Elektor.Labs [5].

(130154 – version française : Robert Grignard)

Liens

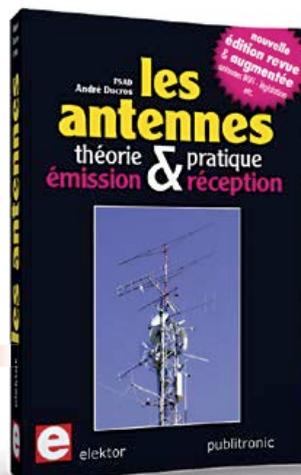
- [1] www.elektor.fr/120668
- [2] www.elektor.fr/100576
- [3] www.elektor.fr/120296
- [4] www.elektor.fr/130154
- [5] www.elektor-labs.com/efl
- [6] www.der-hammer.info/terminal
- [7] www.elektor.fr/110258
- [8] www.elektor.fr/110428
- [9] www.elektor.fr/110405
- [10] www.elektor.fr/120097



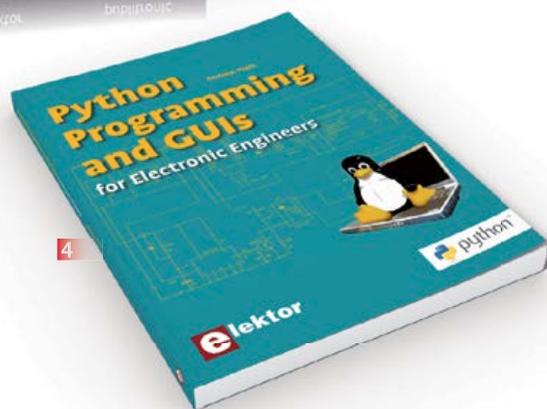
1



2



3



4

Kit (circuits imprimés & composants)

1 Tapir – détecteur ultrasensible d'électrosmog

Fin limier de la pollution électromagnétique qu'il traque et rend audible sur casque, le TAPIR est (aussi) un beau projet à construire, un kit avec tout ce qu'il faut, même le boîtier, qui est fait de quatre circuits imprimés ingénieusement assemblés. Le TAPIR, sigle de Totally Archaic but Practical Interceptor of Radiation (totalement archaïque mais pratique intercepteur de radiations), détecte et localise (en la faisant entendre) toute source de champ électrique E ou, avec l'antenne appropriée, de champ magnétique H.

Réf. : 120354-71 – 14,95 €

Une carte compacte et bon marché qui vous initiera tout en douceur !

2 Embarquez Linux

Linux est partout, même dans certaines machines à café. Souvent, l'électronicien tenté d'adopter ce système d'exploitation est arrêté par sa complexité et par le prix des cartes de développement. Voici Linux pour les électroniciens, sous la forme d'une carte compacte et bon marché qui vous initiera tout en douceur !

Carte Linux Elektor (montée et testée)

Réf. : 120026-91 • 64,95 €

théorie & pratique – émission & réception

3 Les Antennes

Cette bible des antennes devient l'ouvrage de référence pour les radioamateurs, les techniciens et les ingénieurs. La première partie traite de la propagation des ondes dans l'espace et sur les lignes ainsi que des caractéristiques fondamentales des antennes (gain, rayonnement, courant, tension...). Cette étude théorique est suivie de réalisations pratiques, entre autres les antennes filaires, les antennes à gain, et les antennes à large bande et multibandes. La dernière partie est consacrée aux ultimes réglages : adaptation des impédances, appareils de mesure, conseils de sécurité (poussée du vent, résistance des matériaux, pylônes et haubans, foudre...).

470 pages – ISBN 978-2-86661-165-1 – 49,50 €

4 Python Programming and GUIs

for Electronic Engineers

Ce livre s'adresse aux techniciens, scientifiques et amateurs qui souhaitent développer des interfaces graphiques pour faire communiquer entre eux le matériel de leur projet et divers ordinateurs. Vous apprendrez à écrire en Python des applications conçues

aussi bien pour le web que pour les environnements de bureau. Python est un langage de plus haut niveau que le C, mais avant tout un langage de script orienté objet. Sa concision permet d'écrire des programmes courts, donc moins sujets aux erreurs. Sa clarté rend également les codes plus lisibles et plus faciles à modifier après coup. Avec Python, vous pouvez vous concentrer sur le fonctionnement d'ensemble de vos projets. Cette abstraction vaut aussi pour l'écriture des interfaces graphiques. Ici, pas de code de bas niveau pour la gestion d'image ou d'événements comme les clics de souris, Python dispose pour cela de fonctions relativement simples.

223 pages – ISBN 978-0-905705-87-3 – 34,50 €

Informations complémentaires et gamme complète sur :

www.elektor.fr/e-choppe

Elektor/Publitronic SARL

1, rue de la Haye – BP 12910 – 95731 Roissy CDG Cedex

Tél. : +33(0)1.49.19.26.19 – Fax : +33(0)1.49.19.22.37

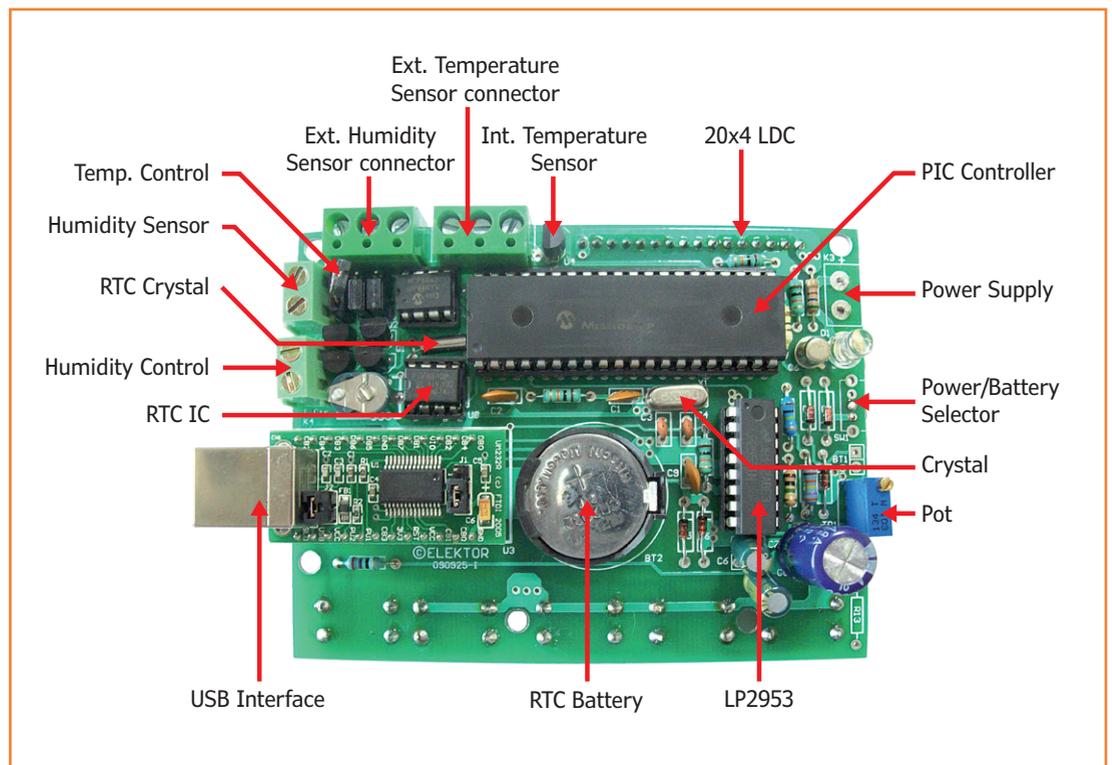
@ : ventes@elektor.fr

thygromètre

capturez le *thygre* avec ce thermomètre hygromètre à mémoire

Manoel Conde de Almeida
(Brésil)

La littérature regorge de thermomètres bricolés et de thermostats numériques de tout poil. Ce projet s'en démarque par la richesse des possibilités qu'il réunit en un seul circuit, dont certaines sont généralement réservées aux instruments professionnels.



L'objectif principal de ce projet est de construire un circuit qui mesure la température et l'humidité relative. Cette version aboutie offre quelques possibilités étonnantes. Pendant qu'elle enregistre les mesures, cette station mémorise les minimums et maximums observés. Elle compare ensuite chaque relevé aux limites introduites par l'utilisateur. Une fois ces limites atteintes, elle peut déclencher des périphériques externes par des sorties dédiées.

La température se mesure en degrés soit Celsius, soit Fahrenheit. Le réglage des limites de tem-

pérature et d'humidité minimales et maximales s'effectue via un PC et un logiciel adapté, disponible en téléchargement gratuit [1].

La station collecte et enregistre les données mesurées dans l'EEPROM embarquée de son microcontrôleur et une horloge numérique y adjoint l'heure, le jour, le mois et l'année pour une horodatation complète. La sonde mesure les données de température et d'hygrométrie toutes les 7 secondes. La mémoire permet d'emmagasiner jusqu'à 240 échantillons combinés sur des périodes qui s'étalent entre 1 et 99 min, par pas d'une minute.

Dompter le thygromètre

L'interface consiste en un écran à cristaux liquides de 20x4 caractères et une rangée de sept boutons (cinq poussoirs et deux interrupteurs marche/arrêt) pour régler tous les paramètres de configuration. On peut aussi contrôler la bête en la raccordant à un PC via le port USB. L'application disponible ici [1] propose, outre une trousse à outils pour paramétrer le montage avec le confort d'un PC complet, d'afficher les mesures récoltées sous la forme d'un petit graphique et de les sauvegarder dans un fichier .csv (données texte séparées par des virgules) afin de les importer dans un tableau pour une analyse plus poussée.

U3 est un module UM232R basé sur la puce FT232R de FTDI [3]. Ce module apporte une interface UART/USB. On le connecte au PORT C du microcontrôleur via les broches C.6 et C.7, configurées en tant que Tx et Rx du contrôleur UART interne. L'alimentation de ce module provient directement de l'interface USB, afin de préserver la batterie pour le reste du circuit.

U8 désigne l'horloge en temps réel, un PCF8583 fabriqué par NXP [4]. Le condensateur C8 et le quartz X2 de 32.768 kHz forment un circuit oscillant. U8 communique avec le microcontrôleur par le biais d'un bus I²C. Les lignes de communication SCL et SDA se connectent au PORT C du

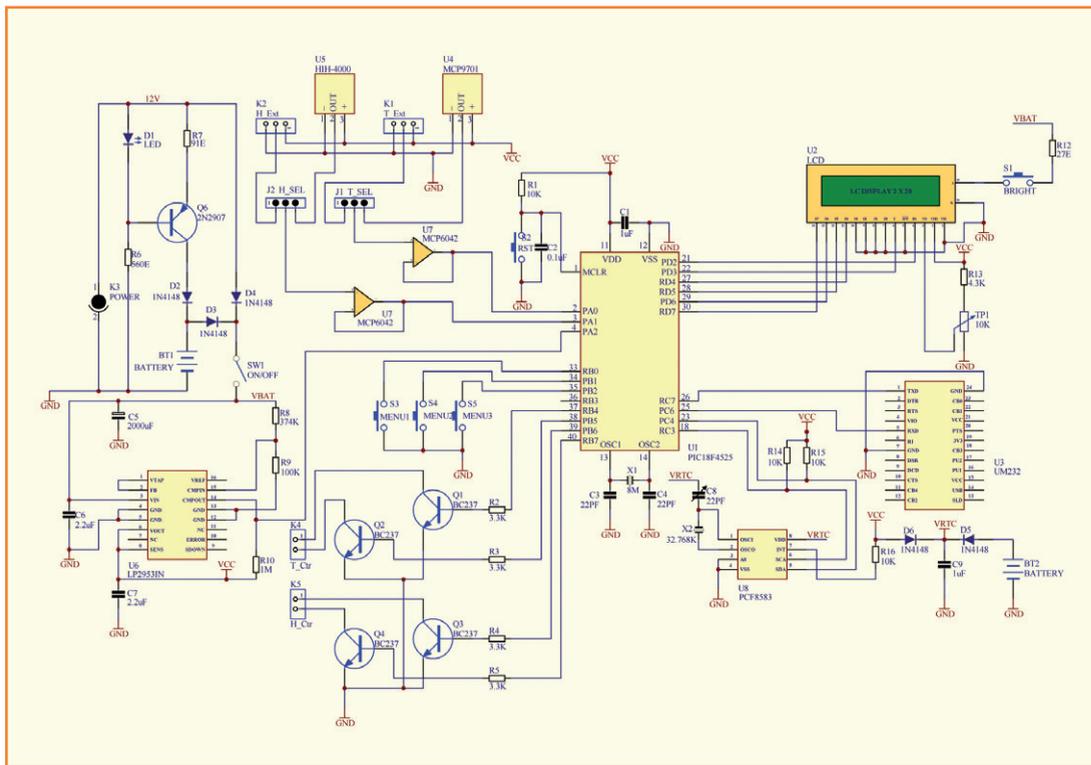


Figure 1. Voici le circuit complet. On remarque tout de suite quelques circuits particuliers comme l'horloge en temps réel et le module de recharge de la batterie.

L'essentiel du matériel

Au centre du circuit de la **figure 1** se trouve un microcontrôleur PIC18F4525 de chez *Microchip* [2]. Il tourne à une fréquence de 8 MHz, dictée par un circuit oscillant composé du quartz X1 et des condensateurs C3 et C4. Le circuit de remise à zéro est constitué de R1, C2 et du bouton-poussoir S2. L'écran LCD 20x4 se connecte au PORT D du contrôleur grâce à un bus de communication sur 4 bits. L'interrupteur S1 allume l'éclairage arrière de l'écran tandis que R13 et le potentiomètre TP1 en ajustent le contraste.

contrôleur par les broches C.18 et C.23. Les poussoirs S3, S4 et S5 donnent accès au menu de fonctions de l'afficheur LCD. Les transistors Q1-Q4 et les résistances R2-R5 constituent des circuits à collecteur ouvert asservis au PORT B par les broches B.4 et B.7. Ces ports s'activent chaque fois qu'une fonction d'alarme est déclenchée parce que la température et/ou l'humidité ont atteint les limites programmées. Le **tableau 1** indique quelles broches du PORT B correspondent aux différentes alarmes. On peut alimenter le circuit par une batterie NiMH

Caractéristiques

- PIC18F4525 Microcontrôleur (DIP-40)
- Écran à cristaux liquides 20x4 (16 broches)
- US232 (convertisseur USB vers série) de FTDI
- MCP9701 sonde de température
- HIH4000 sonde d'humidité
- LP2953 régulateur de puissance
- PC8583 horloge temps réel
- Fonctionne avec une batterie de 9V, circuit de rechargement inclus

Tableau 1. Correspondance entre broche et alarme

Broche	alarme
B.4	limite maximum de température
B.5	limite minimum de température
B.6	limite maximum d'humidité
B.7	limite minimum d'humidité

de 9 V, qui se met en charge chaque fois qu'on rebranche la station à son alimentation de 12 V. Le transistor Q6, la diode D1 et les résistances R6, R7 fournissent un courant d'environ 10 mA, qui recharge la batterie dès qu'on relie l'alimentation à K3. Les diodes D2 à D4 assurent une connexion correcte au circuit. Lorsqu'on branche une alimentation externe, D2 et D4 laissent passer le courant jusqu'à la batterie et au circuit, tandis que D3 bloque la tension excédentaire, ce qui permet à la batterie de se recharger via la source de courant et au circuit d'être alimenté par le secteur. Dès qu'on débranche l'alimentation, D2 et D4 bloquent et le courant de la batterie traverse D3, ce qui débranche la source de courant et passe le circuit en mode autonome d'alimentation.

L'interrupteur SW1 connecte la batterie ou l'alimentation auxiliaire vers U6, un régulateur 5 V à faible perte LP2953A de *National Semiconductor*. Les résistances R8 et R9 forment un diviseur de tension qui donne un échantillon de la tension de batterie. Cette tension est comparée à une tension de référence interne et chaque fois que la batterie descend en dessous des 6 V, la sortie de comparaison (broche 14) inhibe la broche A.2 du PORT A. Le micrologiciel détecte ce changement d'état et affiche le symbole de batterie faible sur l'écran.

La consommation totale de courant est limitée à 10 mA, ce qui correspond, en fonction de la capacité, à environ 15 h de fonctionnement continu

avec une batterie à pleine charge. L'horodateur est alimenté par un circuit de secours qui le maintient en fonctionnement même lorsque l'appareil est éteint. Ce circuit est composé d'une pile BT2 de 3 V au lithium, des diodes D5 et D6 ainsi que d'un condensateur C9. Lorsqu'on coupe l'alimentation auxiliaire, D6 est polarisée en inverse et BT2 alimente l'horloge interne via D5. Dès qu'on rebranche l'alimentation, il se produit l'inverse. D6 transmet le courant, D5 le bloque et l'horloge est alimentée sur V_{CC} .

Au sujet des capteurs

La température est mesurée par une puce *Linear Active Thermistor™* de *Microchip*, le MCP9701, qui la convertit de façon linéaire en une tension proportionnelle selon la formule :

$$V_t = 0,5 + 0,01T,$$

où V_t est la tension de sortie en volts pour une température T en degrés Celsius.

À la sortie du thermomètre se trouve le tampon U7A, l'un des deux amplificateurs opérationnels disponibles sur le MCP6042 (amplificateur opérationnel basse consommation de *Microchip*). Tous deux sont configurés en tampons à gain unitaire. L'humidité relative est mesurée par un hygromètre analogique HIH-4000 de *Honeywell* [5] qui retourne une tension linéaire proportionnelle à l'humidité selon cette formule :

$$V_h = 0,7617 + 0,0297H$$

où V_h est la tension de sortie en volts pour une humidité relative donnée H (%).

À la sortie de l'hygromètre aussi, on retrouve un tampon U7B. Les deux sorties des amplis op. sont reliées aux broches A.0 et A.1 du PORT A du microcontrôleur, configurées en convertisseurs A/N à 10 bits.

Les connecteurs K1 et K2 permettent de brancher des sondes externes de température et d'humidité, à condition d'employer des capteurs identiques à ceux décrits ci-dessus. Les cavaliers J1 et J2 servent à commuter entre capteurs internes ou externes. L'application pour PC autorise un réglage plus fin des paramètres de sensibilité de ces capteurs.

Micrologiciel

Le micrologiciel s'occupe de soulager l'électronique des opérations complexes des fonctions

de mesure, ce qui simplifie le matériel. Il a été écrit en *MikroBasic v7.2* de *Mikroelektronika* [6]. L'ordinogramme (**fig. 2**) décrit sommairement la fonction principale du programme, dont une grande partie se cache derrière une armada de routines dont nous vous ferons grâce ici. Le programme principal commence par quelques routines d'initialisation telles que la configuration du registre interne du microcontrôleur, la mise en service de l'écran LCD, l'écriture en EEPROM des données de configuration standard pour les mesures (si ce n'est pas déjà fait), l'initialisation des variables de service, la mise à zéro de `TIMER0` et la validation de l'interruption qui lui est propre.

Après cette initialisation, le programme se lance dans une boucle infinie (la boucle On) en plusieurs séquences :

- vérifier l'état de la batterie et allumer ou éteindre le témoin de batterie faible ;
- rafraîchir le calendrier à l'écran avec les valeurs délivrées par l'horloge en temps réel ;
- si de nouvelles mesures ont été récoltées, mettre à jour leur affichage ;
- vérifier les boutons du menu pour identifier les éventuelles requêtes utilisateur afin de modifier l'affichage ou lancer une fonction selon le schéma programmé ;
- enregistrer des données si la fonction est active ;
- vérifier la présence d'un PC connecté et exécuter les commandes reçues.

Bien que ce point ne soit pas décrit dans l'ordinogramme, toutes les fonctions dont l'exécution dépend du temps se basent sur les interruptions de dépassement de `TIMER0`. Étudiez le code source ce logiciel, il est dévoilé intégralement dans la documentation du projet [7].

Programme PC

L'application pour PC a été rédigée en *Visual Basic 2008 Express Edition*, disponible gratuitement chez *Microsoft*. Nous allons nous contenter de broser ici ses principales fonctions. La documentation en ligne [1] présente une description

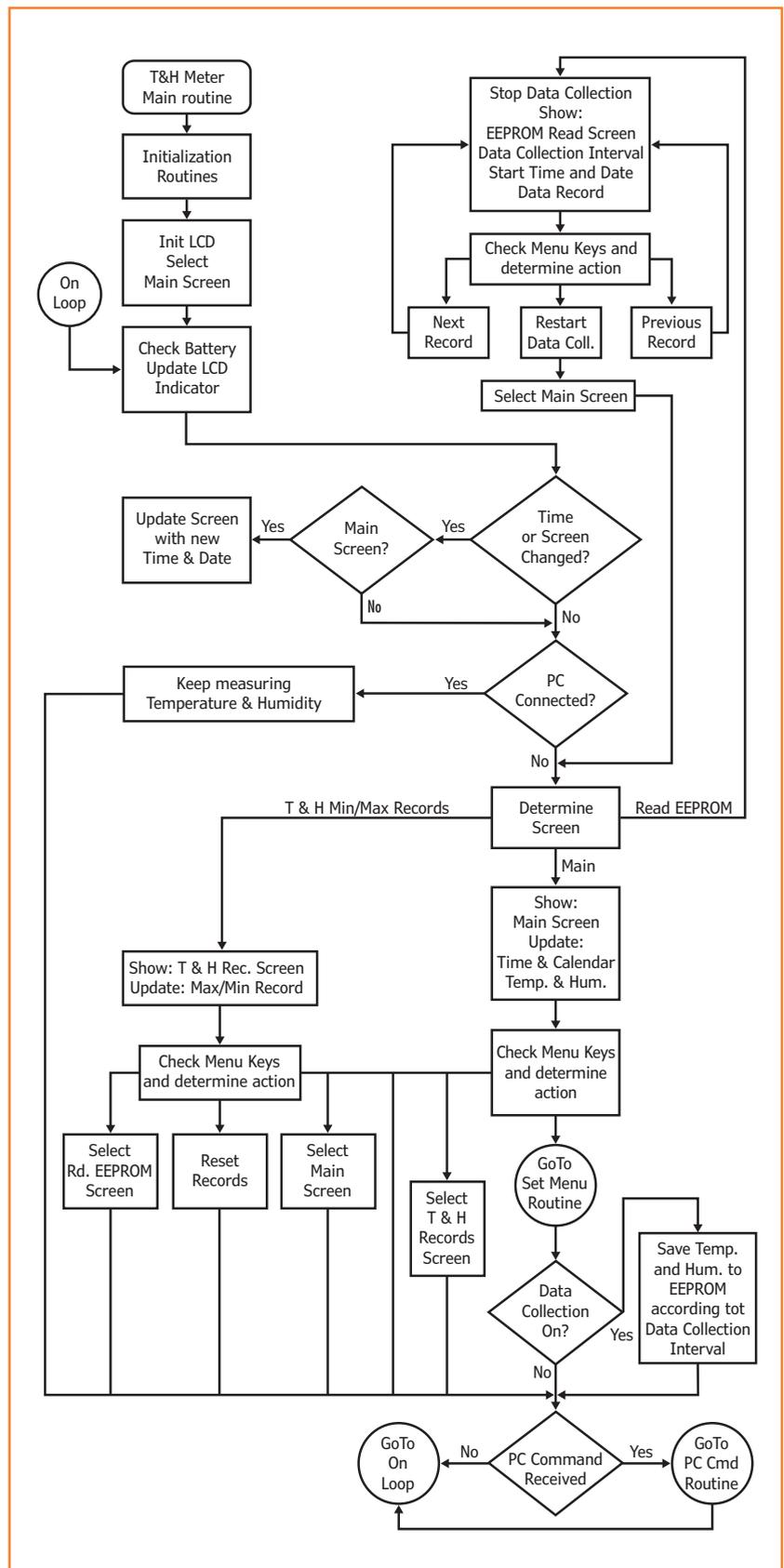


Figure 2.

Cet ordinogramme expose la structure globale de fonctionnement du micrologiciel exécuté par le microcontrôleur.

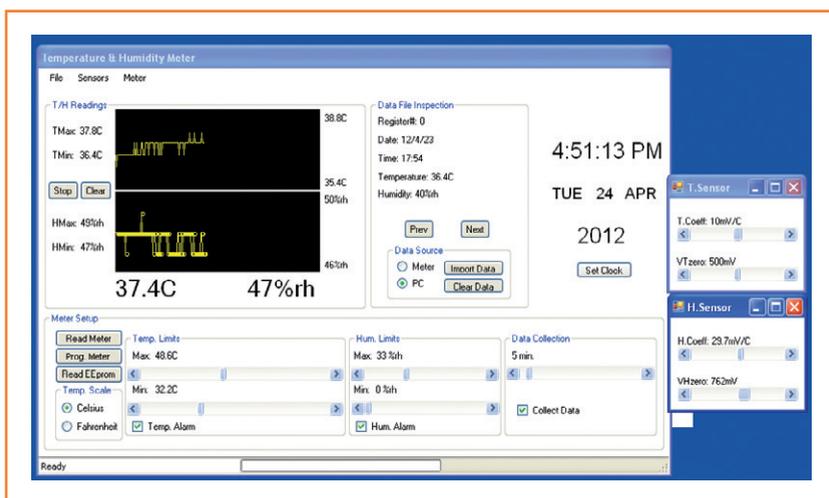


Figure 3. L'interface graphique assure un accès aisé aux différents paramètres de la station thygrométrique.

exhaustive du logiciel, avec tous les détails de son installation et les explications pour en tirer le meilleur.

Afin d'établir une connexion par l'interface USB jusqu'à la station de mesures, le logiciel utilise le pilote gratuit D2XX proposé sur le site du FTDI [8]. Le **figure 3** montre la fenêtre principale du programme. Le menu *File* donne accès à la suppression et à la copie des journaux de données, enregistrés dans des fichiers .csv. Le menu *Sensors* regroupe les fonctions destinées à l'ajustement précis des capteurs de température et d'humidité. On connecte ou déconnecte la station du PC sur lequel tourne l'application via le menu *Meter*.

Le cadre *T/H Readings* affiche la température et l'humidité mesurées dans un graphique de lignes. Sur la gauche, on trouve les minimums et maximums mémorisés et en dessous, en grands caractères gras, les dernières mesures effectuées. Pour synchroniser l'horloge de la station avec celle du PC, on utilisera le bouton *Set Clock*.

Le cadre *Data File Inspection* présente les données récoltées et inscrites dans le fichier .csv. On importe les données dans l'explorateur de fichiers du logiciel après avoir sélectionné la source et cliqué sur le bouton *Import*. Les boutons *Prev* et *Next* servent à se déplacer parmi les données courantes. Le bouton *Clear Data* efface les tableaux de l'explorateur de fichiers, mais conserve intactes les données originales dans les fichiers.

Le cadre *Meter Setup* regroupe toutes les commandes destinées à la configuration de la station et des paramètres d'utilisation du programme. Chaque fois qu'on lance le programme et qu'on

se connecte à la station, il télécharge automatiquement sa configuration. Tous les paramètres prennent alors leur état dans la configuration importée. Les modifications apportées à la configuration courante affecteront immédiatement le fonctionnement du logiciel. Pour appliquer une autre configuration à la station, à l'exception de l'échelle des températures, il faut la lui communiquer d'un clic sur le bouton *Prog. Meter*.

Le programme vérifie sans arrêt les limites de température et d'humidité introduites et prévient, grâce à un repère visuel sur la courbe du graphique, chaque fois qu'elles ont été franchies. Les cases à cocher *Temp. Alarm* et *Hum. Alarm* activent ou désactivent ces fonctions de monitoring. L'ascenseur *Data Collection* détermine l'intervalle de temps entre les échantillons et leur prélèvement démarre dès qu'on coche la case adjacente. Par cette opération, on amorce l'écoute des données en provenance des sondes de température et d'humidité, et l'enregistrement des données dans le fichier pc_cd.csv selon l'intervalle choisi. Sur la station, cela signifie enregistrer les données de mesures dans l'EEPROM. Il est facile d'importer les fichiers csv dans un programme de type *Microsoft Excel* ou *Sun OpenOffice Calc*. Ensuite, vous pouvez leur appliquer tous les traitements voulus, sans limites de complexité comme des rapports, des analyses statistiques, des graphiques, etc.

Comme de coutume, vous trouverez l'intégralité des programmes utilisés, y compris le micrologiciel, en téléchargement sur la page de ce projet [1]. La liste des composants est également en ligne, ainsi que le dessin du circuit imprimé et toutes les informations à propos de l'installation et de l'utilisation du logiciel. N'hésitez pas à nous communiquer vos suggestions et vos commentaires sur le site des projets d'Elektor [7].

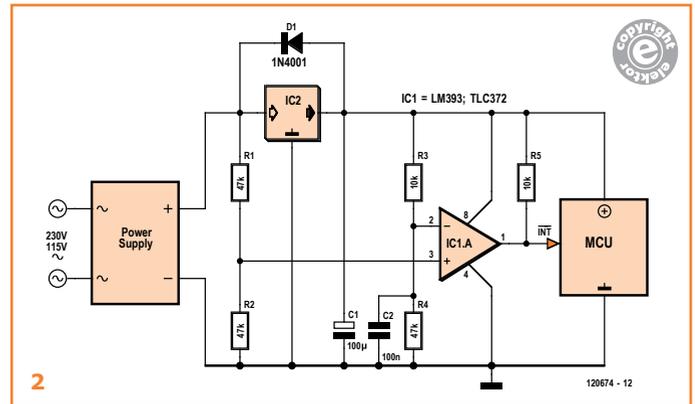
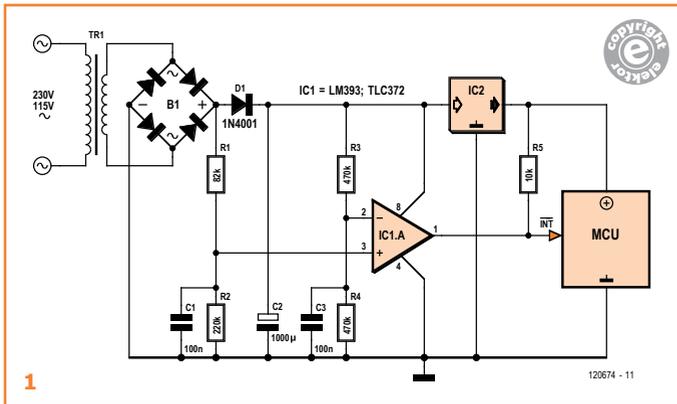
(090925 – version française : Robert Grignard)

Liens

- [1] www.elektor-magazine.fr/090925
- [2] www.microchip.com
- [3] www.ftdichip.com
- [4] www.nxp.com
- [5] sensing.honeywell.com
- [6] www.mikroe.com/mikrobasic
- [7] www.elektor-projects.com/090925
- [8] www.ftdichip.com/FTDrivers.htm

sauve qui peut !

version 2.0



Les concepteurs recourent souvent à l'EEPROM d'un microcontrôleur pour conserver les données de fonctionnement ou les paramètres d'un appareil après sa mise hors tension, ou encore après une coupure de courant, pour y retrouver ces informations à la prochaine mise sous tension. En 2009, dans la rubrique *Secrets du concepteur*, Rainer Reush avait présenté une telle solution de sauvegarde automatique. Dans son montage (fig. 1), IC1 compare la tension en amont de D1 à celle tamponnée par C2. Lorsque le circuit est mis hors tension, cette diode et ce condensateur font que la tension baisse plus vite à l'entrée non-inverseuse d'IC1.A qu'à son entrée inverseuse. Le signal de niveau bas qui en résulte déclenche une interruption du microcontrôleur, signal pour lui du lancement de la sauvegarde des données. Le délai dont il dispose est fonction de l'énergie que contient encore le condensateur électrolytique C2.

Ce circuit est efficace, au moins dans les situations simples. Son problème, ce sont les quelques microsecondes que prennent l'écriture en EEPROM. Et C2 doit être dimensionné au-delà du nécessaire, car il doit aussi fournir le courant de repos du régulateur de tension. Le dimensionnement des autres composants, p. ex. C1, n'est pas non plus des plus simple en raison de l'ondulation présente. Ce n'est pas tout : si vous remplacez le transformateur et son pont redresseur par un bloc secteur moderne qui possède une fonction de stabilisation, voire par une alimentation secteur à découpage, la commande de sauvegarde est mise hors-jeu. Car ces alimentations possèdent un condensateur tampon électrolytique, et avec lui la tension en R1 ne

chute pas suffisamment vite.

Jürgen Werner s'est penché sur ces problèmes. Le fruit de ses réflexions est un circuit amélioré et plus simple (fig. 2). Le comparateur a été déplacé derrière le régulateur de tension, seule la tension d'entrée est comparée à la tension de sortie du régulateur, la diode n'est plus en série avec l'entrée du régulateur, et la capacité du condensateur tampon C1 peut donc maintenant être plus petite. Vient le meilleur : le circuit ne dépend plus de la vitesse à laquelle chute la tension d'entrée. Si en effet l'alimentation est coupée, la tension d'entrée chute tandis que la tension de sortie est maintenue constante durant un certain temps par le régulateur de tension. Avec des diviseurs de tension correctement dimensionnés, la tension de l'entrée non-inverseuse du comparateur est toujours plus petite que celle de l'entrée inverseuse, et le signal de niveau bas qui en résulte déclenche une interruption du µC.

Le circuit est dimensionné pour un bloc secteur de 9 V et un régulateur de tension de 5 V. D1 protège le régulateur contre les inversions de polarité. Avec une capacité de 100 µF pour C1 et une charge de 5 mA, le µC dispose d'au moins 17 ms pour enregistrer les données ou paramètres importants en EEPROM. Il faut ici privilégier une interruption déclenchée sur front. La désactivation de certaines unités du contrôleur comme le convertisseur A/N peut fournir un délai supplémentaire pour la sauvegarde.

(120674 – version française : Hervé Moreau)

Jürgen Werner
(Allemagne)

[1] Sauve qui peut ! Elektor février 2009,
www.elektor.fr/080379

neurosciences cognitives et musique

Tessel Renzenbrink
(Elektor TTF)

Les récents progrès dans l'imagerie médicale par résonance magnétique permettent aux chercheurs de mieux observer le cerveau, la neuroscience cognitive en profite pour se développer à toute vitesse. Artur C. Jaschke est musicien, neuromusicologue et chercheur attaché à la clinique de neuropsychologie de l'université libre d'Amsterdam. Il étudie les effets de la musique sur le cerveau.



Figure 1.
Artur Jaschke
(photo : Elte Rauch)

– Tessel : *Qu'est-ce que la neuroscience cognitive et que fait-elle ?*

– Artur : La neuroscience cognitive explore la relation entre le comportement de l'individu et l'activité de son cerveau. C'est un vaste chantier où se conjuguent les apports de la neurologie, de la biologie, de la psychologie et des sciences humaines.

De nombreuses maladies ont leur origine dans le cerveau, comme Alzheimer, la schizophrénie, la dépression et l'autisme. Notre connaissance du cerveau est éminemment parcellaire, c'est pourquoi il est si difficile de traiter ces maladies. Plus on en saura sur le fonctionnement du cerveau,

mieux on comprendra ses affections. On ne les guérira peut-être pas, mais on améliorera au moins la qualité de vie de ceux qui en souffrent. Pensons par exemple aux interfaces cerveau/machine qui permettent de diriger une machine rien qu'avec son cerveau. Que d'espoir pour les tétraplégiques ! Différentes expériences ont réussi à rendre un peu indépendance à certains d'entre eux en les équipant d'un bras robotisé.

– T : *De quoi provient cet énorme essor de la neuroscience ?*

– A : C'est la suite logique de 20 ans d'innovations technologiques. Pour la première fois, nous sommes arrivés à observer le cerveau en action. Auparavant, on ne pouvait qu'étudier l'anatomie ou capter les signaux émis par l'encéphale.

Maintenant, on a la capacité de se promener littéralement dans le crâne d'un patient conscient et d'observer précisément les zones d'activité en temps réel.

En neuroscience cognitive, nous avons surtout recours à l'imagerie par résonance magnétique fonctionnelle (IRMf) pour étudier la matière grise et à l'imagerie du tenseur de diffusion (ITD) pour la matière blanche. Le traitement de l'information se fait dans la matière grise tandis que la matière blanche véhicule la communication. Imaginez deux maisons dans lesquelles tout se passe, c'est la matière grise. Entre elles, un câble transporte les signaux, c'est la matière blanche. L'IRMf mesure la teneur en oxygène du sang. Quand une zone du cerveau est active, elle a besoin de plus de sang. À l'examen, on voit (**fig. 2**) les parties actives s'éclairer. L'ITD montre le parcours suivi et les signaux qui y transitent (**fig. 3**).

– T : *Que faites-vous en neuropsychologie clinique ?*

– A : La neuropsychologie se situe entre psychologie et neurologie et tâche de créer des liens entre elles. Nous observons par exemple des patients déments que nous comparons à un groupe de contrôle sain. À l'aide de tests neuropsychologiques, nous notons les différences de comportement. Nous leur faisons placer des cartes, par exemple, dans une configuration particulière. Les personnes atteintes ont beaucoup plus de difficultés que les individus sains. Nous leur faisons refaire le test pendant l'examen de l'activité cérébrale sous IRMf. Nous constatons alors une moindre complexité des réseaux de connexions dans certaines parties du cerveau des déments. Nous obtenons ainsi une image de la composante physique de la maladie. Nous consacrons ensuite plus de temps à des interventions comme

la thérapie de mouvement chez les déments. Nous intercalons différents moments de mesure tant avec des tests neuropsychologiques que des explorations du cerveau. Cela nous fournit deux ensembles de données qui permettent de cartographier l'efficacité de l'intervention.

Ma spécialité, c'est la thérapie par la musique, parce que tout un chacun y a accès. Il y a un large consensus sur le fait que la musique a un effet positif sur les enfants autistes. Vous pouvez observer les progrès dans leur comportement social et leurs compétences en communication. Cela ne guérit pas l'enfant, mais améliore sa qualité de vie. C'est malaisé à quantifier scientifiquement, mais j'essaie.

La musique est un système dynamique. Écoutez un morceau que vous n'avez jamais entendu, vous y reconnaîtrez plein de choses : il y a une mélodie, du rythme, une harmonie, mais c'est toujours une composition différente. Par sa dynamique, la musique est un défi intellectuel, un stimulus sain, non invasif, qui pousse le cerveau à créer de nouvelles connexions.

Toutes nos expériences ont une base physique. Pour enregistrer une nouvelle expérience, le cerveau fait de nouvelles synapses, ce sont des liaisons entre neurones. Le changement continu du réseau neural s'appelle la plasticité neurale. Si vous faites souvent quelque chose de nouveau, le cerveau apprécie. Il reste donc actif.

– T : *Comment la musique peut-elle changer le comportement ?*

– A : Cela concerne les fonctions exécutives qui sont des processus cognitifs tels que la planification, la prise de décision, la résolution de problèmes et le multitâche. Des activités très faciles pour les personnes saines, mais pas pour les autistes. Or, ces processus sont ceux que vous exécutez constamment au quotidien, ils ont une grande influence sur le comportement.

Si vous écoutez activement de la musique ou, mieux encore, si vous en jouez, c'est entre autres votre cortex préfrontal qui est extrêmement actif, ce qui stimule le développement de nouvelles connexions. Pendant les fonctions exécutives, le cortex préfrontal est fort impliqué. L'amélioration de la connectivité du réseau a donc aussi un effet sur les fonctions exécutives. Vous pouvez utiliser les nouvelles liaisons pour planifier ou faire plusieurs choses en même temps. C'est la théorie et nous essayons empiriquement d'en apporter la preuve.

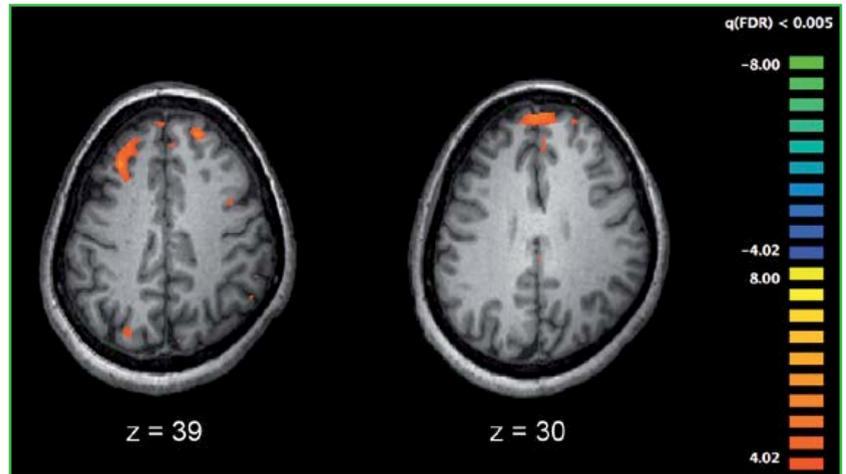


Figure 2. Exemple d'une IRMf lors d'une tâche en mémoire de travail.

(photo : J. Kim, N.L. Matthews & S. Park CC BY 2.5)



Figure 3. Image d'une mesure d'ITD du cerveau humain.

(photo : Thomas Schultz CC BY-SA 3.0)

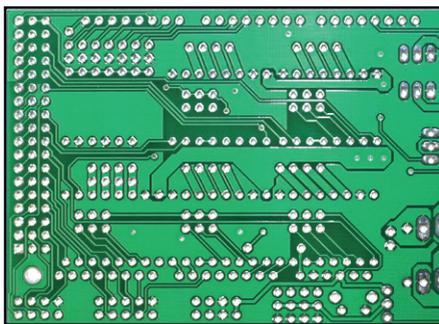
Chacun sait que la musique a un effet positif sur l'humain, ce ne sont pas les parents d'autistes ni les scientifiques qui diront le contraire. Mais personne ne sait comment ça marche. On connaît le résultat, on veut savoir comment il arrive.

(130056 – version française : Robert Grignard)

Lecture recommandée :

Sur le sujet des rapports entre la musique et certaines pathologies du cerveau :

Musicophilia : la musique, le cerveau et nous, Oliver Sacks, Ed. du Seuil, 2012



gravure de CI professionnels

Le processus de fabrication de votre CI à quatre couches



La technologie de fabrication des circuits imprimés professionnels a évolué énormément depuis l'époque où Elektor publiait ses premières « platines », il y a cinquante ans. Dans cet article, nous allons explorer le laboratoire de notre partenaire *Eurocircuits* et voir comment ils cuisinent leurs gâteaux multicouches.



Comprendre le mode de fabrication vous aidera sans aucun doute à concevoir des circuits plus faciles et moins coûteux à produire, tout en améliorant la fiabilité à long terme. Voyons donc le processus de fabrication d'une carte à quatre couches. Habituellement, les fabricants de circuits imprimés professionnels ne produisent pas de pièces uniques. Au lieu de cela, ils combinent plusieurs circuits sur une grande planche de production, pour une manipulation plus efficace d'un bout à l'autre du processus de fabrication. C'est ce qu'on appelle *order pooling*, ou groupement des commandes. C'est ainsi que pratique Eurocircuits. Les illustrations vous montrent quatre dessins différents sur une même planche.

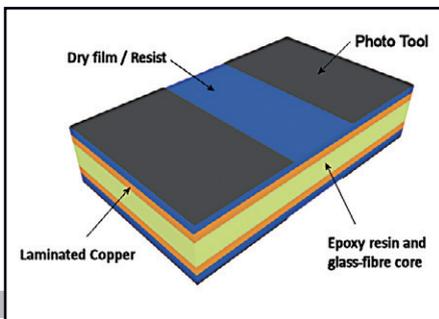
(1) D Gerber aux données de fabrication

Le concepteur du circuit imprimé a préparé le dessin des pistes avec un système de CAO (Conception Assistée par Ordinateur). Face aux formats de données propriétaires de chaque système, l'industrie du CI a créé une structure standard de fichiers de sortie, de façon à uniformiser le format de description des propriétés physiques d'une carte. Ce format s'appelle Gerber, la version actuelle est « Extended Gerber » ou « RS274X ». Les fichiers Gerber définissent les couches de pistes en cuivre, de même que les masques de soudage, la position des composants et leur repérage. On commence par vérifier si les données remises par un client répondent aux exigences de la fabrication. C'est fait automatiquement pour la plus grande partie. Les largeurs de piste, l'isolement entre elles, les collerettes, le plus petit diamètre de perçage, toutes ces caractéristiques sont comparées aux possibilités de la fabrication. Une fois le dessin accepté, un technicien préparera les outils nécessaires au fonctionnement des machines qui produisent et testent le circuit imprimé.



(2) Outillage pour le transfert de l'image du CI

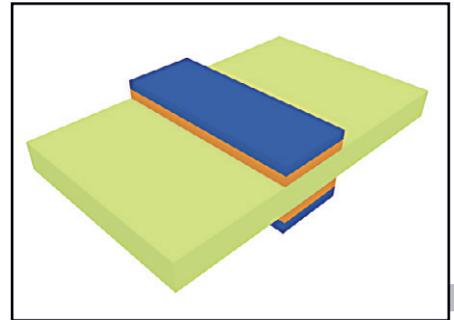
Un traceur à laser insole les films photographiques nécessaires à la production. Ils sont développés automatiquement et prêts à entrer dans le processus de fabrication du CI. Un film, ou outil photographique, est produit pour chaque couche. Les films sont repérés l'un par rapport à l'autre par des perforations placées précisément dans chaque feuille. Ces trous correspondent aux pions de centrage des matériels d'exposition, de telle façon que les différentes couches de la carte soient parfaitement alignées.



(3) Copie des couches internes

Pour fabriquer les couches internes d'un circuit multicouche, les façonniers comme Eurocircuits démarrent typiquement avec une planche d'isolant cuivré, un sandwich de fibre de verre-résine époxy entre deux feuilles de cuivre. Le cuivre est d'abord nettoyé et la planche passe dans une pièce hors poussière pour garantir qu'aucune poussière ne se dépose sur la surface, où elle risquerait de provoquer un court-circuit ou une coupure. La planche nettoyée reçoit une couche de film photosensible, dit *photo-resist*. Ensuite, l'image est reportée sur le circuit imprimé par une insoleuse, dans laquelle de puissantes lampes à ultra-violettes polymérisent le *photo-resist* à travers les zones transparentes du film, pour former une image du cuivre. Le châssis de l'insoleuse porte des pions de repérage correspondant aux perforations des films et de la planche. L'opérateur positionne le premier film sur les pions, ensuite la planche, ensuite le deuxième film. Les pions assurent l'alignement précis des couches supérieure et inférieure.

Une fois insolée, la planche reçoit une pulvérisation d'une solution fortement alcaline qui dépouille le *photo-resist* qui n'a pas été complètement polymérisé. La planche est lavée sous pression et séchée. Le dessin du cuivre est maintenant recouvert par le *photo-resist* durci. L'opérateur vérifie le panneau pour s'assurer que la surface du cuivre est propre et tout le *photo-resist* superflu a été enlevé.



4

(4) Gravure des couches internes

Le dessin du cuivre est maintenant gravé avec une solution fortement alcaline pour dissoudre le cuivre à nu. Le processus est suivi attentivement pour garantir que la largeur finale des conducteurs est celle qui était prévue.

Ensuite, le *photo-resist* bleu qui protège l'image de cuivre est pelé. L'opérateur vérifie que tout a été enlevé.



4

(5) Alignement et contrôle des couches internes

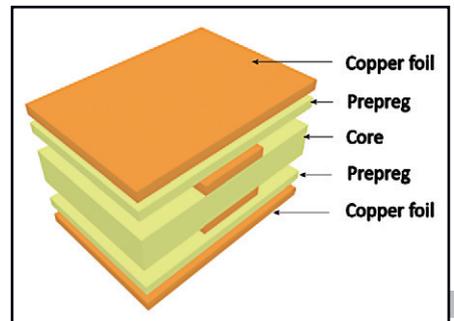
L'âme de notre carte multicouche est maintenant prête. L'opérateur poinçonne les trous de repérage pour l'alignement des couches internes et externes. Comme il n'y a pas de moyen de corriger un défaut des couches internes une fois les couches externes collées, la planche est soumise à un contrôle complet. Un système automatique de contrôle optique scanne la planche et la compare à l'image numérique tirée des données de conception originales.



5

(6) Collage des couches internes et externes

Chaque couche externe est constituée de feuilles de tissu de verre pré-imprégnées de résine époxy non polymérisée (*prepreg*) et d'une mince feuille de cuivre. D'abord, une feuille de cuivre, puis deux feuilles de *prepreg* sont placées sur la lourde table en acier. Ensuite, l'âme préparée est positionnée soigneusement à l'aide des pions de repérage. Enfin, deux autres feuilles de *prepreg*, une autre feuille de cuivre et une plaque de pression en aluminium sont disposées par-dessus. Cet empilage est placé dans la presse à coller, qui met en œuvre la chaleur et la pression pour coller entre elles les couches du circuit imprimé. La chaleur fond et durcit la résine du *prepreg*, pendant que la pression lie les couches ensemble. Le processus est commandé par ordinateur, pour assurer que le collage tienne la durée de vie du CI.

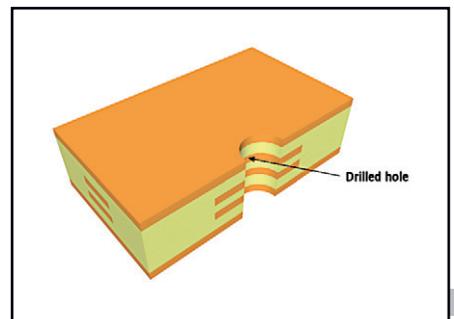


6

(7) Perçage

Perçage aux rayons X des trous de référence

Avant la gravure des couches de cuivre externes, tous les trous des composants traversants et des vias sont percés. Une broche équipée de rayons X sert à repérer la position des perçages dans le cuivre des couches internes. La machine perce les trous de repérage pour garantir le perçage précis au centre des pastilles internes.



7



Préparation des empilements pour le perçage

L'opérateur installe d'abord une contre-plaque en fibre sur la table pour empêcher le foret de déchirer la feuille de cuivre quand il débouche en dessous. Ensuite il installe la planche et une feuille d'entrée en aluminium.

Perçage des trous

La perceuse est à commande numérique. L'opérateur charge le programme de perçage adéquat (« bande de perçage »). Le programme indique à la machine quel foret utiliser et à quelles coordonnées XY. La broche est entraînée à l'air comprimé et peut tourner à 150 000 tours par minute. Le perçage à grande vitesse assure que la paroi des trous est lisse et offre une base sûre pour la métallisation.

Le perçage est un processus lent puisque chaque trou doit être percé individuellement. Aussi, suivant le diamètre des forets, on peut percer jusqu'à trois planches superposées. La machine choisit le foret à utiliser dans la tourelle, vérifie si c'est le diamètre correct et le charge dans la tête de perçage.

Élimination de l'excédent de résine

Pendant le processus de collage, l'excédent de résine du *prepreg* est chassé vers les bords de la planche, en dehors de la zone de l'image. Cet excédent est éliminé par une fraiseuse à commande numérique. La planche percée est prête pour la métallisation.

(8) Métallisation – première phase

Une couche conductrice est d'abord déposée sur les parois des trous. L'opérateur fixe la planche dans un cadre qui permettra de la transporter à travers une série de bains chimiques et de rinçages dans lesquels les parois des trous reçoivent une amorce de microparticules de palladium et une couche de un micron d'épaisseur de cuivre chimique. Le reste du cuivre est prêt pour la galvanoplastie, un procédé qui dépose du métal par électrolyse sur une surface conductrice. Mais d'abord...

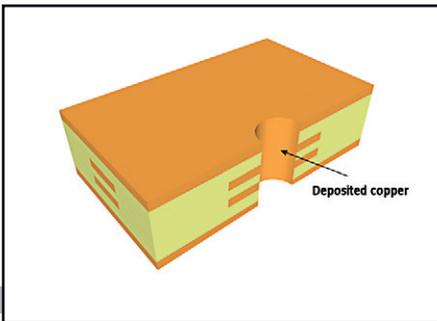
(9) Report des images des couches externes

... la planche revient en salle hors poussière pour recevoir une couche de *photo-resist*, laminée à chaud sur le cuivre dans une machine à feuilles. L'opérateur positionne le premier film sur les pions de repérage, puis la planche préparée et enfin le deuxième film. Le sandwich est insolé à travers les films des couches extérieures, développé, puis le *photo-resist* qui n'a pas été durci est éliminé. Le cuivre du dessin des couches externes est à nu.

(10) Métallisation – deuxième phase

Maintenant, la planche reçoit le dépôt de cuivre électrolytique. L'opérateur démarre la chaîne de métallisation automatique, où la surface du cuivre est nettoyée et activée dans plusieurs bains et ensuite métallisée. Tout le processus est piloté par ordinateur pour garantir que chaque planche reste dans chaque bain le temps exact.

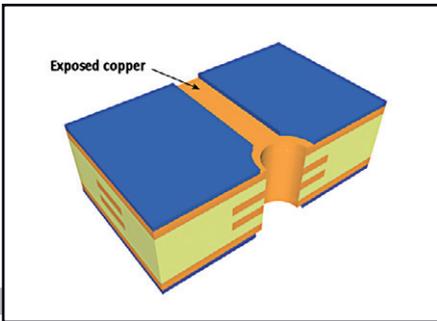
7



8



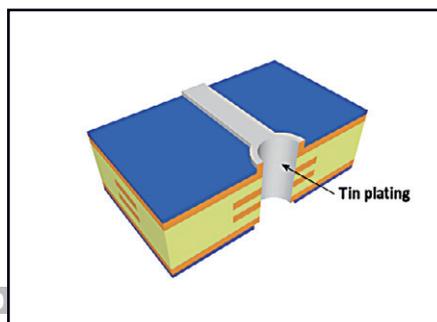
8



9



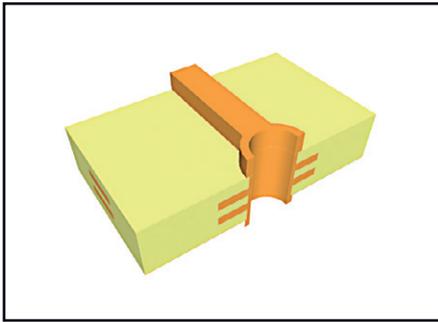
9



10



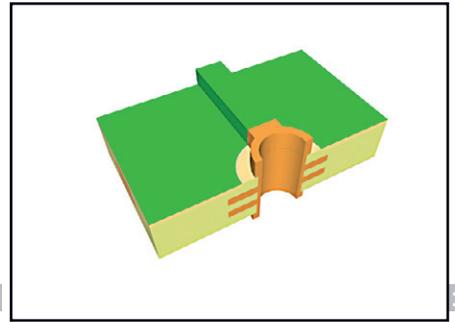
10



11



11

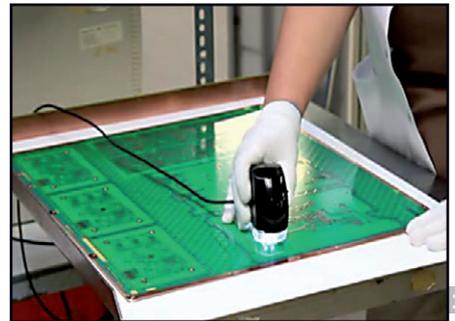


12

Pour assurer une bonne conductivité dans les trous, environ 25 microns de cuivre sont nécessaires sur les parois des trous. Du fait de la galvanoplastie, 25 à 30 μm sont déposés aussi sur le cuivre des pistes. Ainsi, en démarrant avec une feuille de cuivre de 17,5 μm , on trouve une couche de cuivre de 40 à 42 μm après le traitement. Pour l'étape suivante – élimination du reste de la feuille de cuivre –, une fine couche d'étain est déposée sur le cuivre. Quand la métallisation est terminée, l'opérateur vérifie par une méthode non destructive l'épaisseur des couches de cuivre et d'étain.

(11) Gravure des couches extérieures

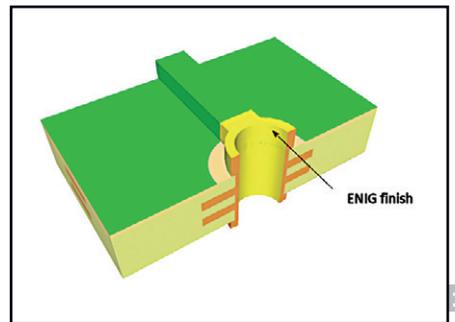
Les couches externes sont gravées maintenant. D'abord, le photo-resist qui recouvre le cuivre indésirable est dissous et évacué par lavage. Ensuite une solution alcaline forte attaque tout le cuivre nu. Le processus est contrôlé soigneusement pour éviter la sous-gravure, afin que les pistes aient exactement la largeur prévue. Pour finir, le placage d'étain qui protégeait l'image du cuivre est éliminé.



12

(12) Vernis épargne soudure

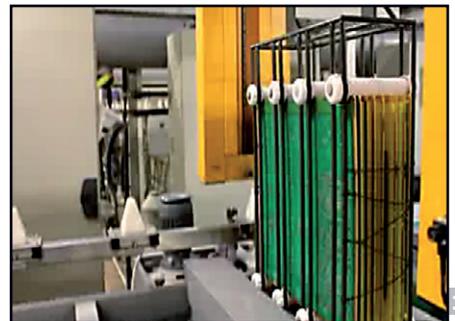
Avant l'application du vernis épargne soudure, les planches sont nettoyées et brossées pour supprimer toute zone ternie. Elles sont ensuite chargées dans une machine verticale qui enduit simultanément les deux faces de la planche avec l'encre époxy du masque de soudure. La planche est encadrée et transportée à travers un séchoir qui durcit l'encre juste assez pour l'insolation (« sec au toucher »). L'opérateur vérifie si la couverture est complète et uniforme.



13

Ensuite les planches enduites sont insolées aux ultra-violet. L'opérateur monte les films dans la machine et installe la planche sur les pions de repérage. Comme pour les *photo-resists* utilisés précédemment pour la gravure et la galvanoplastie, la lumière ultra-violette polymérise l'encre où le film est transparent. C'est là que se trouvera le masque de soudure sur la carte terminée.

Les panneaux insolés sont évacués par une bande transporteuse de la salle hors poussière vers le révélateur qui élimine l'encre non insolée indésirable. L'opérateur vérifie l'alignement du masque avec la planche et s'assure qu'il n'y a pas de trace d'encre sur les pastilles ou dans les trous. Pour assurer une protection forte et permanente, le masque est durci au four.

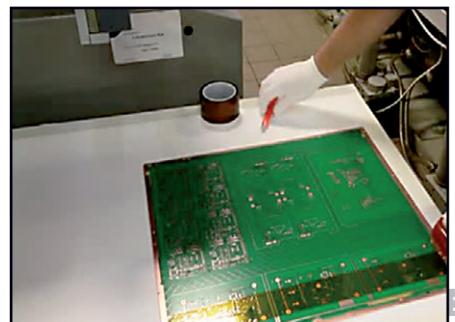


13

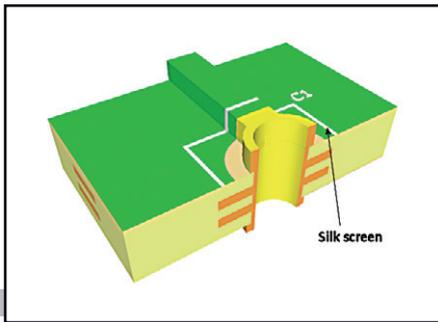
(13) Finition des pastilles et des trous

Les îlots de cuivre et les trous ne sont pas masqués par le vernis épargne. Une finition de surface soudable leur est appliquée pour protéger le cuivre jusqu'au soudage des composants. L'image affiche un fini doré obtenu par un dépôt chimique de 5 μm de nickel recouvert par un dépôt de 0,1 μm d'or.

Du fait de la législation européenne *Reduction of Hazardous Substances* (RoHS), le plomb ne peut plus être utilisé pour la finition, c'est pourquoi l'or est proposé, de même qu'une finition en argent fin ou un lissage sans plomb à l'air chaud. Pour cette dernière option, la planche est trempée dans un bain d'étain fondu. Au sortir



14



du bain, des jets d'air chaud chassent l'excédent de métal fondu pour laisser une couche d'étain d'environ 2 μm .

(14) Dorure solide par galvanoplastie

L'or par galvanoplastie est nécessaire pour les languettes au bord des cartes qui sont régulièrement enfichées dans des connecteurs de bus et retirées. L'opérateur commence par coller une bande de protection au-dessus des languettes. Il monte ensuite la planche dans un bain de galvanoplastie horizontal. Le dépôt de 1 à 1,5 μm d'or vient par-dessus un dépôt de 4 à 5 μm de nickel.

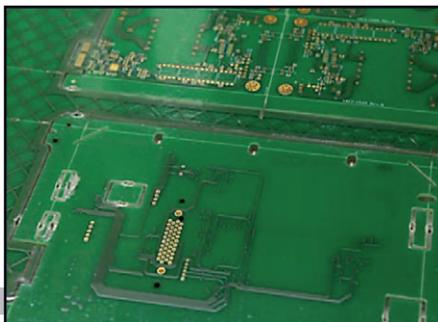
(15) « Sérigraphie »

Le contour et le repérage des composants sont imprimés directement par une imprimante à jet d'encre spéciale. On continue à parler de sérigraphie, comme du temps où les « écrans de soie » étaient en fil d'acier. L'imprimante fonctionne exactement comme une imprimante ordinaire qui projette des gouttelettes d'encre sur la planche pour produire l'image. Ensuite l'encre époxy du masque de soudure et celle de la « sérigraphie » sont soumises au durcissement final. Cela prend environ dix minutes dans un four à cinq étages avec transporteur.



(16) Découpage des planches

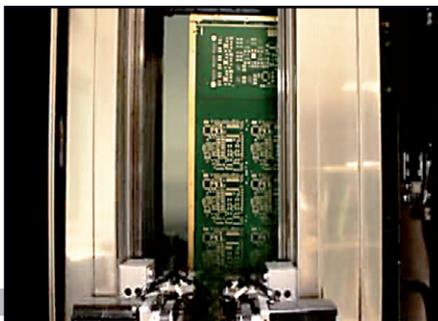
La planche est prête pour la séparation des différents circuits imprimés. Une fraiseuse à commande numérique exécute d'abord toutes les petites encoches et découpes internes. Ensuite la broche charge automatiquement une fraise de 2 mm, vérifie le diamètre et découpe le contour de chaque circuit imprimé.



(17) Test électrique

À la fin, chaque carte multicouche est testée électriquement par comparaison avec les données originales. Une sonde volante vérifie que chaque ligne est complète et n'est pas en court-circuit avec une autre.

Une méthode plus rapide, en option, utilise une machine *Acceler8* qui comporte 4000 pointes minuscules, comme une brosse. Elle construit une carte électrique d'un circuit testé précédemment. Ensuite elle compare chaque circuit à sa carte. Le procédé réduit la durée des tests de 90 %.



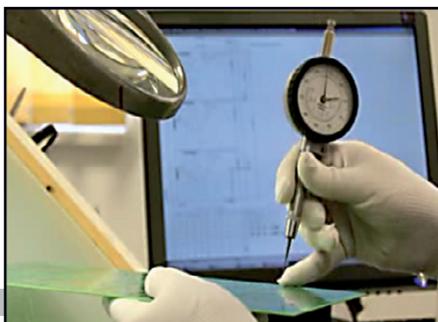
(18) Inspection finale

Pour finir, une équipe d'inspecteurs à l'œil de lynx examine soigneusement chaque circuit imprimé. Si tout est correct, une marque de sortie est apposée. Les circuits sont emballés sous vide pour éviter la poussière et l'humidité, puis dans du film à bulles et expédiés aux clients dans une boîte solide et scellée.

Dans la prochaine livraison, nous nous intéresserons aux règles de conception qui résultent de propriétés physiques du processus de fabrication décrit ici.

(130061 – version française : Jean-Paul Brodier)

Illustrations Eurocircuits



Lien Internet : www.elektorpcbsevice.com

Eurocircuits est un façonnier européen de circuits imprimés en technologie standard. Son siège est situé dans la ville belge pittoresque de Malines, alors que les centres de production se trouvent près d'Aix-la-Chapelle en Allemagne et à Eger en Hongrie. Eurocircuits se spécialise dans les prototypes et les petites séries de circuits pour les concepteurs, les bureaux d'études, les marchés de niche, les universités et les centres de recherches.

Professional Quality
Trusted Service
Secure Ordering



Elektor PCB Service at a glance:

- 4 Targeted pooling services and 1 non-pooling service
- Free online PCB data verification service
- Online price calculator available
- No minimum order value
- No film charges or start-up charges

Delivery
from 2
working
days

retour sur des particularités du LCR-mètre 0,05 %

Le luxe de la précision accessible à tous



Jean-Jacques Aubry,
Ollioules

Le succès du LCR-mètre

Les réactions enthousiastes et attentives de nombreux lecteurs à la publication de ce projet montrent que les appareils de mesure de qualité restent parmi leurs réalisations favorites. L'auteur, Jean-Jacques Aubry, a donné sur le forum français [4] une rectification — la seule erreur significative relevée à ce jour — sur le plan de découpe pour le connecteur USB et SW1. Pour la découpe de SW1, la cote 1,16 mm doit être 0,96 mm. Pour la découpe de J19, la cote 8,03 mm doit être 5,39 mm et la cote 3,97 mm doit être 6,61 mm. Un dessin corrigé a été incorporé au téléchargement sur le site, de même qu'une nouvelle version du schéma qui rectifie quelques points de détail. Dans la liste des composants du circuit principal, J17, listé avec J7, J8... est en fait J9. On y trouve deux fois R81 ; la bonne valeur est 7,5 kΩ, il faut supprimer R81 = 10 kΩ. Sur le schéma du circuit principal, R31 = 750 Ω et sur le schéma de l'extension, la résistance en série avec la LED D5 est R8 = 1 kΩ. Et enfin C30 = 1n5 5% NPO.

Ces rectifications n'ont aucune incidence sur le fonctionnement du circuit. Le circuit imprimé et

le module prêt à l'emploi restent inchangés, et comme tout fonctionne bien, il n'y a pas eu à ce jour (21 mai 2013) de mise à jour depuis la parution du 3^e article. Et puisqu'on en est aux détails, au début du document *Mise en route (First time Setup* dans l'anglaise version) à télécharger depuis notre site, le fichier du *firmware* est appelé à tort *LCR3A_update_Vxxx.hex*. Le nom exact est *LCR3A_firmware_Vxxx.hex*.

L'auteur sera heureux de répondre directement à d'autres questions sur le forum et engagera volontiers le dialogue avec les utilisateurs de son LCR-mètre de précision.

Dans les trois articles de la description du LCR-mètre [1], certains aspects ont été passés sous silence pour ne pas en allonger démesurément la publication. Nous vous proposons ici de revenir sur des détails qui peuvent être intéressants aussi en dehors du cadre de l'appareil décrit.

C'est le cas d'une routine particulière pour l'affichage sur un écran graphique d'icônes qui demande une certaine gymnastique.

Il sera question également de la précision des mesures, des facteurs d'imprécision et des erreurs. Pour que la lecture de cet article soit utile, il est évidemment préférable d'avoir lu aussi les trois précédents, avec notamment la description du schéma et des principes de mesure.

Bibliothèque pour l'afficheur graphique

Le *bootloader* et le *firmware* utilisent, si nécessaire, l'afficheur graphique (GLCD) pour afficher des messages et des images. La RAM de l'afficheur est organisée en 128 colonnes et 8 lignes d'octets (64 bits ou pixels). Or les éléments à afficher (icônes) dépassent les 8 bits en hauteur et sont décrits sur plusieurs octets : la hauteur des deux polices de caractères utilisées est p. ex. de 11 et 16 bits. Tous les icônes sont définis dans le fichier *glcd_bitmaps.c* (ou *bootloader_glcd-bitmaps.c* pour sa version minimaliste, sans certaines des icônes et polices utilisées par le *firmware*). Si l'on veut pouvoir écrire à n'importe

quelle position, p. ex. à cheval sur les frontières d'octets, et que l'écriture dans la RAM du GLCD se fait par octets entiers à des adresses bien définies, il faut :

connaître le contenu de la RAM avant écriture, faire de savants calculs entre les anciens et nouveaux octets pour ne modifier que les pixels (bits) nécessaires.

Le nombre de lignes de port disponibles pour l'afficheur rend obligatoire l'utilisation du mode *série*. Malheureusement, ce mode ne permet pas la lecture de la RAM de l'afficheur. Il faut donc

créer, dans la RAM du MCU, un miroir de celle du GLCD :

```
uchar xdata GLCD_Array[LCD_COLS]
[LCD_ROWS];
```

Ces calculs entre anciens et nouveaux octets apparaissent vite inextricables tant le nombre de cas différents à résoudre est grand. La solution que j'ai adoptée consiste, pour une colonne, à passer d'une représentation par 8 octets à une représentation par 64 bits ! Cette solution nécessite :

1. La création dans la RAM du MCU de l'image d'une colonne où chaque octet est représentatif d'un seul pixel de la colonne de la RAM du GLCD, donc valeur 0 ou 1.

```
uchar xdata Column_Array[(LCD_ROWS + 1) * 8]; // + 1 for 2nd byte of char in last
line
```

2. La création d'une routine de lecture de la colonne de l'afficheur (en fait du miroir) et écriture, après transformation en équivalent bits (0 ou 1), dans `Column_Array[]`.

```
void GLCD_read_column(uchar col)
{
    uchar rows, pix, i, j;
    i = 0;
    // read column col, byte after byte
    for (rows = 0; rows < LCD_ROWS; rows++)
    {
        pix = GLCD_Array[col][rows];
        // write pix, bit after bit
        for (j = 0; j < 8; j++)
        {
            Column_Array[i] = pix & 0x01;
            pix >>= 1;
            i++;
        }
    }
}
```

3. La création d'une routine de lecture du contenu de `Column_Array[]`, écriture dans la RAM du GLCD et mise à jour du miroir :

```
void GLCD_write_column(uchar col)
{
    uchar rows, pix, i, j;
    i = 0;
    // write column col, byte after byte
    for (rows = 0; rows < LCD_ROWS; rows++)
    {
        pix = 0;
        // read pix, bit after bit
        for (j = 0; j < 8; j++)
        {
            pix += Column_Array[i] << j;
            i++;
        }
        if (GLCD_Array[col][rows] != pix) // only if GLCD RAM byte modified
        {
```

```

        GLCD_set_pos(rows, col);
        GLCD_WriteData(pix);           // draw in GLCD RAM
        GLCD_Array[col][rows] = pix;  // update RAM mirror
    }
}

```

4. La création d'une routine pour l'affichage d'une icône définie en *bitmap*. On peut le faire n'importe où, quelles que soient ses dimensions (dans les limites de l'afficheur !).

```

void GLCD_show_icon(uchar code *bitmap, uchar width, uchar height, uchar x, uchar
y, uchar mode)
{
    uchar tx, ty, pix, hb, i, j, k;
    hb = (height - 1) / 8 + 1; // character height in bytes
    for (tx = 0; tx < width; tx++) // loop for width columns
    {
        GLCD_read_column(tx + x);
        i = y;
        k = 0;
        for (ty = 0; ty < hb; ty++) // read hb bytes of icon
        {
            pix = *(bitmap + ty * width + tx); // read one byte
            if (mode == GLCD_PIXEL_OFF)
                pix = ~pix;
            for (j = 0; j < 8; j++) // write 8 bits of pix to Column_Array
            {
                if (mode != GLCD_PIXEL_INV)
                    Column_Array[i] = pix & 0x01;
                else
                    Column_Array[i] ^= pix & 0x01;
                pix >>= 1;
                if (k == height)
                    break;
                i++;
                k++;
            }
        }
        GLCD_write_column(tx + x);
    }
}

```

5. Et enfin, la création d'une routine d'écriture d'un texte qui ne sera que l'écriture d'une succession d'icônes définies dans une police.

```

void GLCD_draw_text( uchar x, uchar y, uchar *text , uchar mode )
{
    uchar i, posx, posy;
    uchar *pt;
    posy = y - font_height + 1;
    for( pt = text, i = 0; *pt; i++, pt++ )
    {
        posx = x + i * font_width;
        if( posx + font_width > LCD_COLS )
        {
            i = 0;
            posx = x;
            posy += font_height;
        }
        GLCD_show_icon( font + (*pt - font_offset) * font_charsize, font_
width, font_height, posx, posy, mode );
    }
}

```

Précision des mesures

L'impédance à mesurer peut s'écrire :

$$Z_x = \frac{V_p + jV_q}{I_p + jI_q} \times \frac{G_i R_{sense}}{G_v} = \left[\frac{V_p I_p + V_q I_q}{I_p^2 + I_q^2} + j \frac{V_q I_p - V_p I_q}{I_p^2 + I_q^2} \right] \times \frac{G_i R_{sense}}{G_v}$$

expression dans laquelle G_i et G_v sont les gains en courant et en tension de la chaîne d'amplification, et R_{sense} la résistance du *convertisseur_IU*.

$$G_i = G_{INA128} \times G_{BUFFER} \times G_{PGA^i} \times G_{DAC^i} \quad G_v = G_{INA128} \times G_{BUFFER} \times G_{PGA^v} \times G_{DAC^v}$$

Soit encore, en utilisant la représentation *série* d'une impédance $Z = R_S + j X_S$

$$R_S = \frac{V_p I_p + V_q I_q}{I_p^2 + I_q^2} \times \frac{G_{PGA^i} G_{DAC^i} R_{sense}}{G_{PGA^v} G_{DAC^v}} \quad X_S = \frac{V_q I_p - V_p I_q}{I_p^2 + I_q^2} \times \frac{G_{PGA^i} G_{DAC^i} R_{sense}}{G_{PGA^v} G_{DAC^v}}$$

Si on néglige les erreurs de numérisation comme expliqué au paragraphe *Il faut veiller au gain* du premier article [1], alors :

$$\frac{\Delta R_S}{R_S} = \frac{\Delta X_S}{X_S} = \left[\frac{\Delta G_{PGA^i}}{G_{PGA^i}} + \frac{\Delta G_{PGA^v}}{G_{PGA^v}} \right]_{i \neq v} + \left[\frac{\Delta G_{DAC^i}}{G_{DAC^i}} + \frac{\Delta G_{DAC^v}}{G_{DAC^v}} \right]_{i \neq v} + \frac{\Delta R_{sense}}{R_{sense}}$$

On peut séparer l'erreur globale en deux parties principales:

- l'erreur due à l'imprécision sur les gains du PGA103 et du DAC8811
- l'erreur sur la vraie valeur des résistances du *convertisseur_IU* (erreur de calibrage).

À cela s'ajoutera une erreur due :

- à l'affichage (± 1 bit du dernier chiffre)
- à l'erreur de phase résiduelle (après compensation de l'erreur de phase)
- aux fluctuations créées par le bruit d'amplification et les signaux parasites captés par les cordons de mesure (secteur, etc.).

Et puis l'erreur de numérisation apparaîtra sur les gammes 1 et 8, quand les amplitudes des signaux tension ou courant deviendront trop différentes par manque de gain.

Facteur d'imprécision

Erreur de gain

D'après la documentation *BURR-BROWN (TI)* du **PGA103**

Gain	1	10	100
Typical gain error	$\pm 0,005\%$	$\pm 0,02\%$	$\pm 0,05\%$
Max gain error	$\pm 0,02\%$	$\pm 0,05\%$	$\pm 0,2\%$

Pour les gammes 3 à 6, l'amplificateur programmable PGA103 a toujours un gain unitaire et le terme

$$\left[\frac{\Delta G_{PGA^i}}{G_{PGA^i}} + \frac{\Delta G_{PGA^v}}{G_{PGA^v}} \right] \text{ est nul.}$$

Pour les gammes 2 et 7, ce terme introduit une erreur maximum de $\pm 0,07\%$ ($\pm 0,025\%$ typique).

Pour les gammes 1 et 8, ce terme introduit une erreur maximum de $\pm 0,22\%$ ($\pm 0,055\%$ typique).

D'après la documentation *TI* du **DAC8811C**, sa précision relative max. est de ± 1 LSB. L'erreur maximale sur le gain sera alors de $\pm 1/N$, où N est le code qui détermine le gain de l'amplificateur final.

palier post_Ampli.	0	1	2	3	4	5	6	7
N	7 500	8 700	10 000	11 600	13 500	15 500	18 000	20 700
palier post_Ampli.	8	9	A	B	C	D	E	F
N	24 000	27 600	32 000	36 900	42 600	49 100	56 700	65 500

Quand les paliers de *post_amplification_U* et *post_amplification_I* sont égaux, le terme $\left[\frac{\Delta G_{DAC^i}}{G_{DAC^i}} + \frac{\Delta G_{DAC^v}}{G_{DAC^v}} \right]$ est nul.

Sinon, il est maximum quand l'un est 0 et l'autre 1 ; il vaut alors $\frac{1}{7500} + \frac{1}{8700}$ soit 0,025%

Note : L'utilisation de la version B (**DAC8811B**) double cette erreur.

- Le *convertisseur_IU*, dont le gain en boucle ouverte n'est pas infini, introduit aussi une erreur de mesure. Le gain en boucle fermée étant ≤ 1 sur les gammes 3 à 6, son gain en boucle ouverte d'environ 80 dB (10000) à 10 kHz introduit une erreur supplémentaire d'environ 0,01 %. Celle-ci sera négligeable aux fréquences inférieures.

Note : Sur l'ordinateur, le programme *AU2011* permet de visualiser la valeur de l'erreur de gain.

Erreur de phase

L'utilisation dans le *convertisseur_IU* d'un amplificateur à très large bande passante (50 MHz) et les différentes compensations de phase (décrites dans le document téléchargeable *Mise en route* [3]) permettent de réduire au minimum les rotations de phase parasites. Cependant, elles ne sont pas totalement éliminées. De plus, la phase de l'amplificateur final est supposée indépendante du gain, ce qui n'est vrai qu'au premier degré, car dans la plage d'utilisation du DAC8811, avec des données entre 0x2000 et 0xFFFF, sa bande passante est assez constante (environ 8 MHz).

Cette erreur résiduelle aura une incidence sur la valeur du paramètre secondaire, d'autant plus marquée que la phase du composant en test sera proche de 0 ° ou ± 90 °.

Erreur de calibrage

L'erreur initiale est celle des résistances de précision implantées sur le circuit imprimé, c'est-à-dire $\pm 0,05$ %. Il est possible de l'améliorer en suivant les indications du paragraphe 7 *Calibrage des résistances* du document téléchargeable *Mise en route* [3].

(130174)

Liens & références

- [1] LCR-mètre 0,05 % 1re partie www.elektor-magazine.fr/110758
- [2] LCR-mètre 0,05 % 2e partie www.elektor-magazine.fr/130022
- [3] LCR-mètre 0,05 % 3e partie www.elektor-magazine.fr/130093
- [4] www.elektor.fr/forumLCR

e explorez

le monde de l'électronique



avec les cinq nouveaux
vaisseaux de la flotte Elektor

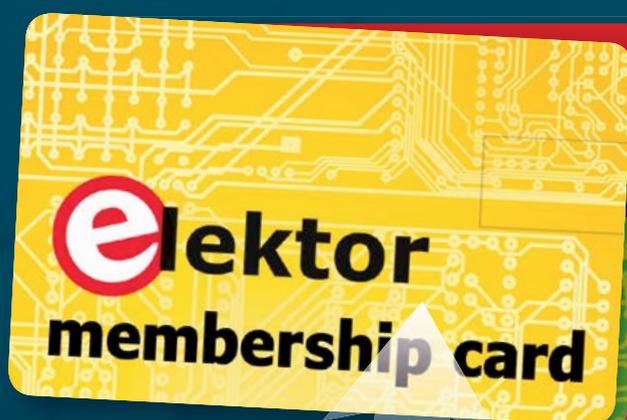
elektor.magazine : **le vaisseau amiral**

elektor.labs : **le drakkar**

elektor.community : **le paquebot**

elektor.post : **le catamaran**

elektor.store : **le cargo**



Rejoignez la communauté Elektor :
embarquement immédiat !



avec puce NFC Mifare programmable

Découvrez nos formules d'abonnement sur www.elektor.fr/abo

les artisans de l'électronique

quand les électrons nous font voyager

David Cuartielles
(Espagne)

Photos :
Laura Balboa (Mexique)



Quand il s'agit d'électronique, deux choses m'intéressent : les objets connectés et l'enseignement de la technologie aux plus jeunes. Les objets connectés en sont à leurs balbutiements, il reste beaucoup à faire mais il y a de belles réussites en perspective. Le monde de l'enseignement, quant à lui, est bien plus complexe, souvent perçu comme rigide, et aurait bien besoin que l'on s'y intéresse.



Figure 1.
L'église de Zegache au Mexique. Merci aux ateliers communautaires – Talleres Comunitarios – pour les images.

D'habitude je vous présente ici en à peu près 2500 mots un bricolage amusant avec pour mission d'intéresser les gens à l'électronique, mais, ce mois-ci c'est différent. J'aimerais vous raconter l'histoire d'atomes et d'octets qui m'ont mené bien au-delà de mes horizons habituels. Je parlerai également des différentes manières qu'a l'électronique de nous connecter à travers le monde. Le titre de cet article aurait dû être « **de l'art de l'électronique à celui de la menuiserie** ».

Liste des courses

Pour reproduire les projets présentés ici, il vous faudra :

- le kit de démarrage Arduino officiel
- un micro servomoteur tout ce qu'il y a de plus standard
- un haut-parleur 8 Ω à membrane en papier
- un capteur de distance à infrarouges Sharp

Expérience n° 1 : le voyage

Santa Ana Zegache est un village au sud de l'état d'Oaxaca, au Mexique. Pour s'y rendre, il faut huit heures de voiture à partir de Mexico. Bien sûr, on peut prendre un vol jusqu'à l'aéroport d'Oaxaca et réduire le trajet de moitié, mais quelle expérience que de s'y rendre en voiture (fig. 1) !

Santa Ana est un village typique. Les chiffres « officiels » parlent de 5.000 habitants, difficile à dire. Nous y sommes arrivés le 27 décembre 2011 pour la première fois. Je n'étais pas plus surpris par l'absence de revêtement sur les routes que par le style pour le moins singulier des bâtiments dits « achevés » mais dont le ferrailage du béton reste apparent.

Il n'y a qu'une seule place dans le village autour de laquelle se trouvent la mairie, l'école, l'église et les deux seuls cybercafés du coin. L'église a été rénovée grâce à l'aide d'une fondation créée par Rodolfo Morales et, je cite la page internet du projet qui m'a amené au Mexique :

« Le maître peintre Rodolfo Morales considérait Zegache comme inestimable. Il a dédié la fin de sa vie à la restauration de son église. Cependant, il ne vit jamais l'achèvement des idées ayant motivé sa vision philanthrope : préserver son héritage artistique et raviver les vocations premières de la région. »

Afin de rénover cette église, un atelier (fig. 2) a été créé et un certain nombre de locaux formés aux arts de la charpenterie et de la menuiserie. Grâce à la générosité de la fondation Alfredo Harp Helú Foundation, de la fondation Rockefeller, de la Curtiduría, et de M. et Mme. Sandretto, les villageois ont pu continuer à travailler pendant plus d'une décennie. Ils ont appris par eux-mêmes le travail et la restauration du bois sous toutes ses formes.

Aujourd'hui ils continuent de restaurer les églises environnantes ainsi que l'iconographie religieuse dans toute la région. Leur atelier est rempli de mains sculptées à la main, d'images du Christ et de miroirs, beaucoup de miroirs (fig. 3).

Depuis un bon moment, la production de miroirs est devenue leur principale source de revenus en plus de la rénovation. Ils invitent des artistes du monde entier à participer à des stages pendant lesquels ces derniers apprennent à connaître le village, l'atelier et leurs miroirs. C'est sur un type traditionnel de miroirs que l'on demande aux artistes de *bricoler*. L'atelier aura ensuite le droit de fabriquer cinq répliques identiques de



Figure 2.
Vue d'ensemble de l'atelier à Zegache.

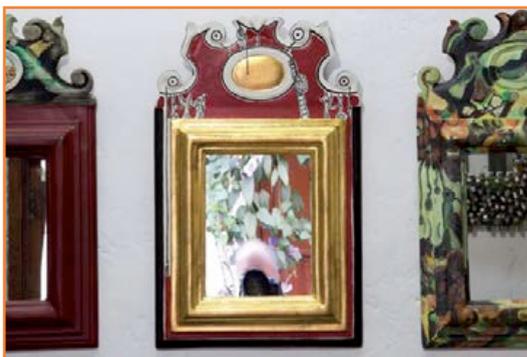


Figure 3.
Les miroirs fabriqués à Zegache.



Figure 4.
Notre atelier Arduino
improvisé dans l'atelier de
Zegache.

l'œuvre créée par l'artiste et de les vendre à un prix donné. Ces pièces en série limitée ont une double fonction : faire rentrer un peu d'argent, mais surtout faire connaître au monde l'atelier, le village et ses habitants.

Notre premier contact avec le village pourrait se résumer ainsi : nous nous y sommes rendus, avons passé quelques heures à visiter l'atelier et avons essayé de nous y imaginer avec un atelier

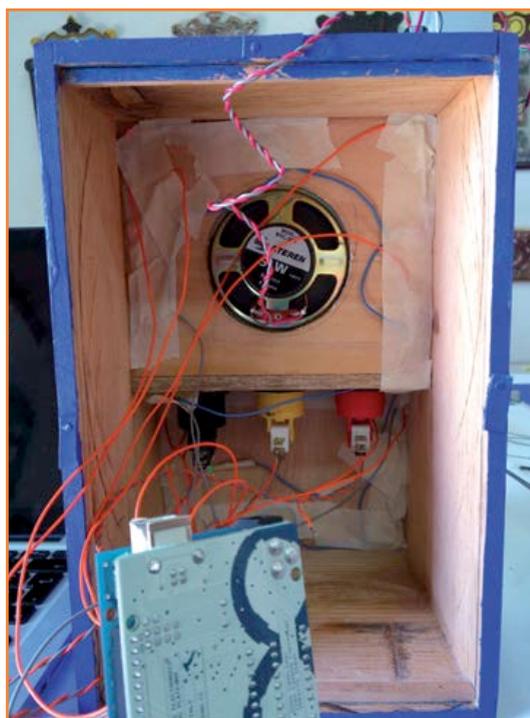


Figure 5.
À Zegache, tout le monde
est à l'œuvre pour fabriquer
miroirs et jouets interactifs.

pour enseigner Arduino : pas beaucoup d'électricité, aucun magasin à 40 km à la ronde, pas de connexion à internet sur place et très peu d'ordinateurs.

Expérience n° 2 : enseigner

C'est en février 2012 que nous nous sommes rendus à Zegache pour la deuxième fois. Nous savions à quoi nous attendre et avons fait nos valises en conséquence. En plus de mon PC j'avais un fer à souder, huit kits de démarrage Arduino, quelques capteurs, des *shields* de prototypage, des HP à membrane en papier et enfin quelques clés USB et CD remplis de logiciels (il n'était même pas imaginable de télécharger quoi que ce soit depuis les cybercafés du coin). Lors de notre précédent voyage, j'avais établi une liste des magasins d'électronique d'Oaxaca, la capitale de l'État située à une heure de route de Zegache, et avais donc une idée précise de ce que je pourrais ou ne pourrais pas y trouver : boutons, fils, LED ou batteries ne seraient pas un problème. Ne pouvant compter sur la présence de l'habituel tableau blanc ou projecteur, nous avons acheté de grandes feuilles de papier et des feutres. Notre atelier Arduino était prêt (**fig. 4**).

Les participants ont été groupés en équipes de trois et ont commencé à apprendre les bases d'Arduino. Chaque équipe (cinq au total) avait son propre ordinateur, kit de démarrage et tous les logiciels dont elle aurait besoin.

Nous avons passé une semaine à présenter les bases de l'électronique : entrées et sorties numériques, conversion analogique/numérique, ce qu'est la programmation ou encore un EDI... Un jour, j'ai observé qu'une des femmes s'amusaient à copier-coller du texte (Ctrl+C, Ctrl+V) apparemment sans raison. J'ai fini par comprendre que c'était pour elle comme de la magie, une expérience jubilatoire... et une bonne leçon pour nous qui, blasés, oublions parfois à quel point l'électronique numérique peut faire de nous des magiciens. Après trois jours passés à parler de bits et d'octets, nous avons commencé la pratique. Nous avons invité les villageois à réfléchir sur la manière dont ils pourraient créer des miroirs interactifs en leur ajoutant par exemple des LED et du son. Nous avons passé le reste des deux semaines à fabriquer des boîtes en bois, les sculpter, les peindre à la main, souder des LED et bien sûr programmer. Tout le monde participait (**fig. 5**). Quelle expérience enrichissante !

Les gens là-bas coopèrent. Peu importait qu'ils aient



Figure 6.
Certains des projets
fabriqués lors du premier
atelier.

chacun un projet à finir, ils trouvaient le temps de s'entraider. Inutile de dire que les cinq projets ont été menés à bien malgré le temps limité.

Nous n'avons pas travaillé tard, ne serait-ce qu'une seule fois : les journées se terminaient à 18 h et tout le monde devait rentrer chez lui pour s'occuper de sa famille. De toute façon, tous les locaux de l'église sont fermés à cette heure-ci et il faut donc obligatoirement quitter les lieux.

Ils recyclent tout. Vous avez sans doute entendu que l'utilisation d'outils numériques tels que les machines de découpe au laser rendent les gens économes avec les matériaux... et bien écoutez ça : nous n'avons pas acheté le moindre morceau de bois pour réaliser les projets. Quand ils avaient besoin d'un morceau de 30 cm ils collaient deux morceaux de 15 cm.

Ils ont appris non seulement les bases de l'électronique, mais également comment souder et assembler les choses à la main. Tous les projets étaient fonctionnels et, comme j'ai pu le vérifier un an après, à l'épreuve du temps.

Notre premier atelier s'est terminé avec une bonne liste de projets (**fig. 6**) :

- un miroir magique qui enchaîne effets sonores et lumineux commandés par des boutons

- un jukebox capable d'ajouter des instruments à une boucle sonore
- un plateau pour jouer à des jeux de boisson (*no comment*)
- un miroir à comméragés qui vous complimente quand vous passez devant
- une boîte à bijoux qui joue un air différent et produit des effets lumineux différents à chaque fois que vous l'ouvrez

Expérience n° 3 : apprendre autant que possible

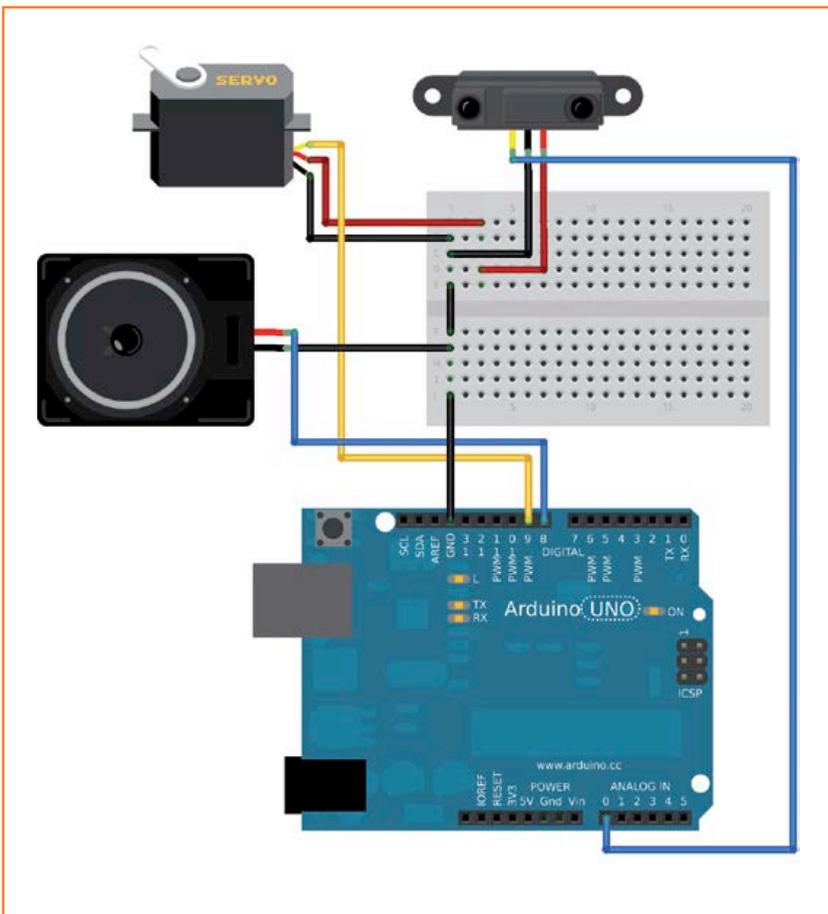
Ce projet s'inscrivait dans le cadre d'un programme artistique de l'Union européenne appelé Euroaxca et au sein duquel différents artistes et artisans des deux côtés de l'Atlantique collaborent et partagent leurs expériences de domaines différents. J'avais planifié une deuxième expérience pour ce projet et, au vu des excellents résultats de la première, hésitais sur les modalités de la suivante.

À mes yeux, il devrait être possible pour les menuisiers de Zegache d'apprendre les bases de la construction de jouets interactifs et de les reproduire de la même manière qu'ils le font avec les miroirs. Un peu plus tard, ils seraient capables d'y apporter leurs modifications et d'en créer de nouveaux.



Figure 7. Parties de notre oiseau robotique sculptées à la main.

Figure 8. L'électronique de notre oiseau robotique.



Or ce but était déjà atteint. Nous avons vu qu'il est possible, en deux semaines de travail sur leurs propres projets, de donner aux gens des rudiments de la pratique de l'électronique. Que faire après ?

La réponse est venue de la coordinatrice du projet européen qui nous a suggéré la fabrication de personnages animés. « Un oiseau, ça serait l'idéal », disait-elle. Convaincu, j'ai proposé à mes collègues, alors que nous quittions la Suède pour nous rendre au Mexique, un an après notre premier voyage : « fabriquons un robot oiseau ! ». Peu de choses avaient changé, que ce soit à

Listing 1. Voici à quoi ressemble le code d'un oiseau robotique

```

/**
 * Zegache Robot
 *
 * (c) 2013 D. Cuartielles
 */

#include <Servo.h>

Servo myservo;

int sensorPin = 0;
int speakerPin = 8;
int servoPin = 9;
int val;

void setup()
{
  myservo.attach(servoPin);
  pinMode(speakerPin, OUTPUT);
}

void loop()
{
  val = analogRead(sensorPin);

  if ( val < 250 && val > 50) {
    myservo.write(50);
    tone(speakerPin, val);
    delay(200);
    myservo.write(100);
    noTone(speakerPin);
    delay(100);
    myservo.write(120);
    tone(speakerPin, val);
    delay(200);
    myservo.write(0);
    noTone(speakerPin);
    delay(200);
  }
}

```

Zegache ou dans l'atelier. Certains étaient partis, d'autres venaient d'arriver. Ils avaient construit une immense structure pour ombrager un peu la place (car là-bas le soleil tape), et deux buts, pour que les enfants puissent jouer au football, et enfin trouvé un moyen de relier l'atelier à l'internet (ou plutôt une sorte d'internet) : quelqu'un dans le village avait un accès à haut débit (apparemment tout le monde n'y a pas droit) partagé à travers de multiples répéteurs sans-fil répartis dans le village. Que deux ordinateurs puissent se connecter en même temps relevait plus de la chance que de la technique !

L'interaction avec les gens a été simple. J'ai approché quelques gars de l'atelier et nous avons parlé de fabriquer des oiseaux robotiques. Jamais en manque d'idées, ils se sont mis à sculpter différents types d'oiseaux capables de bouger les ailes, la queue, ouvrir le bec ou encore secouer la tête (**fig. 7**). Il aura fallu deux jours pour fabriquer le premier oiseau, les suivants ont été faits en un clin d'œil.

Côté circuit et logiciel, tout était très simple (**fig. 8**). Tous les oiseaux possédaient un micro servomoteur tirant une ficelle afin de faire bouger leurs parties mobiles. Rajoutez à ça un haut-parleur pour jouer quelques sons et un capteur à infrarouges pour mesurer la distance à laquelle se trouvent les spectateurs. L'idée était que les oiseaux aient l'apparence d'une simple sculpture de bois (**fig. 9**) jusqu'à ce que quelqu'un s'en approche. Il est également possible d'interagir avec les oiseaux : agiter votre main en face du capteur vous permettra de changer la mélodie. Un simple coup d'œil au **listing 1** vous montrera à quel point il est facile de fabriquer votre propre oiseau robotique. Du côté de la production de son, il est bien entendu possible d'améliorer considérablement les choses, mais je me devais de faire quelque chose que les gens de Zegache pourraient facilement modifier par eux-mêmes. Nous repasserons les voir pour nous concentrer sur la fabrication de jouets sonores encore plus perfectionnés, mais ceci est une autre histoire.

Le mot de la fin

L'électronique sert à faire fonctionner les choses, mais elle peut également être un moyen d'apprendre sur le monde qui vous entoure. Avec un esprit ouvert, il devient possible de fabriquer des projets intéressants en travaillant avec des gens tout autour du monde. Prenez votre carte Arduino



Figure 9.
L'oiseau interactif terminé.

et vos astuces, parlez à vos amis et voisins, et encouragez-les à fabriquer des jouets amusants et des objets interactifs.

Le partage est l'essence de la création !

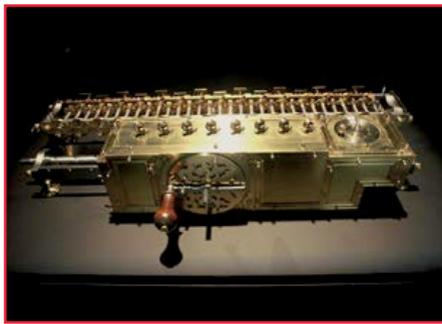
(120751 – version française : Kévin PETIT)

Références

- [1] Le projet de l'atelier de Zegache : http://proyectozegache.com/index_en.php
- [2] Le projet européen Euroaxaca : <http://euroaxaca.org/>

Remerciements

Merci à Geska et Robert pour les photos et surtout pour m'avoir invité à rejoindre le projet Euroaxaca qu'ils ont initié. Merci également aux habitants de Zegache pour leur hospitalité et pour m'avoir rappelé que les objets électroniques ne sont pas faits que d'octets.



Konrad Zuse, ses Z1 à Z4 et au-delà

Tout a commencé avec des zéros, des uns et de l'algèbre



Depuis les premiers jours des mathématiques et de la pensée logique, les hommes ont cherché à simplifier le travail répétitif. Différentes machines à calculer ont été développées, plus ou moins perfectionnées selon la compréhension des fonctions logiques. Cet article évoque la contribution impressionnante de Konrad Zuse au développement de l'ordinateur.



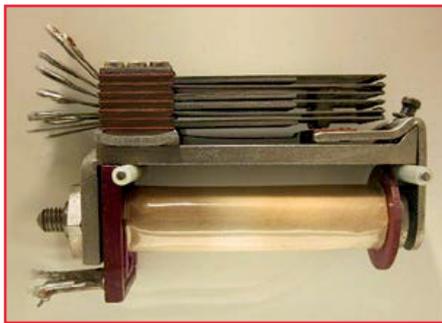
Peter Beil (Allemagne)

Parmi les premières machines à calculer figurent celles de Blaise Pascal (1646), puis de Gottfried Leibnitz (**fig. 1**) environ 50 ans après. Il a fallu deux siècles avant que des machines à calculer soient capables d'exécuter directement des opérations logiques. Konrad Zuse, penseur prolifique et inventeur (**fig. 2**), a construit son Z1 en 1938. C'était un véritable ordinateur (mécanique), programmable librement, qui calculait sur des valeurs binaires. L'original a été détruit pendant la guerre, mais une réplique a été construite, exposée au *Deutsches Technik Museum* de Berlin (**fig. 3**).

nateur les opérations nécessaires était d'utiliser le système binaire. C'était directement à l'opposé des chercheurs et pionniers aux États-Unis et en Grande-Bretagne, qui préféraient le système décimal.

Z2, 16 bits à 10 Hz

Les problèmes mécaniques du Z1 entraînaient Zuse à tenter une expérience : il conçut en 1939 le Z2, qui mettait en œuvre plusieurs centaines de relais de téléphonie (**fig. 4**). Il fonctionnait à une fréquence approximative de 10 Hz et effectuait les quatre opérations arithmétiques sur des nombres binaires en virgule flottante. Il possédait une mémoire à 16 bits et pesait 300 kg.



Z1 : des débuts perturbés

L'ordinateur Z1 était basé sur des fonctions logiques, telles que ET, OU etc. effectuées mécaniquement ! La machine a connu bien des vicissitudes du fait des tolérances mécaniques, frottements etc. Les contacts se bloquaient souvent. Un des avantages du Z1 par rapport aux ordinateurs plus récents est son stockage entièrement non volatile : les machines modernes ne peuvent reproduire cette propriété qu'indirectement. Konrad Zuse avait vu que la seule façon de rendre techniquement réalisables dans son ordi-

Z3 : des données sur film

En mai 1941, Zuse présenta son Z3, le premier ordinateur numérique viable (**fig. 5**). Il utilisait 600 relais dans l'unité arithmétique et 1400 de plus dans l'unité de mémoire. Comme le Z1, il utilisait l'arithmétique binaire à virgule flottante, et il était le premier ordinateur programmable universel. On raconte qu'un ami de Zuse qui travaillait au studio de cinéma UFA lui avait donné l'idée d'utiliser du film perforé comme support des données d'entrée.



Le poinçonnage du film se faisait directement sur l'ordinateur (fig. 6). Naturellement, il y avait aussi un lecteur adapté : les perforations du film permettaient un alignement précis des trous qui représentaient les données (fig. 7).

L'exemplaire original du Z3 a été détruit aussi pendant la guerre, mais une réplique toujours fonctionnelle est exposée au *Deutsches Museum* (musée allemand des sciences et techniques) à Munich. La réplique a été construite par la firme Zuse KG (aujourd'hui dissoute). Un regard sur ses entrailles révèle une quantité phénoménale de câblage, un mécanisme rotatif de commutation pour produire les signaux d'horloge et des relais pas-à-pas comme ceux qui furent utilisés en téléphonie par la suite (fig. 8, 9 et 10). La machine traitait des mots de 22 bits, composés d'un exposant de 7 bits, d'une mantisse de 14 bits et d'un bit de signe. Elle disposait d'un stockage par relais pour 64 mots et les programmes tournaient toujours en boucle. Les nombres étaient saisis au moyen d'un clavier (fig. 11) et les résultats apparaissaient sur de petites lampes dans une zone d'affichage (fig. 12). En plus des quatre opérations arithmétiques, le Z3 offrait une fonction racine carrée.

Z4 : toujours tout mécanique

En 1942, Zuse commença à travailler sur le Z4, prêt à être utilisé en 1945 (fig. 13). Cette machine aussi était basée sur des relais et se trouva en compétition avec les américains Mark I (1944) et ENIAC (1946). Ceux-ci fonctionnaient suivant des principes différents, p. ex. en utilisant des tubes et des représentations décimales (fig. 14, ENIAC 1946).

Comme le Z3, le Z4 était très semblable aux ordinateurs d'aujourd'hui. Il pouvait exécuter des boucles, l'arithmétique en virgule flottante avec une mantisse à 24 bits, un exposant à 7 bits, un bit de signe (soit un mot de 32 bits ordinaire aujourd'hui) et deux unités arithmétiques fonctionnaient simultanément. En plus des opérations de base, il pouvait calculer les carrés et extraire les racines. Le jeu d'instructions du Z4 se composait de 29

instructions. Un organe électrique d'impression était connecté, mais ce n'était pas ce que nous appellerions une imprimante. Il servait à enregistrer une trace des calculs de la machine (fig. 15).

Les programmes étaient encore chargés par les bandes de film perforées (fig. 16). Le « stockage » au sens moderne n'était pas possible et la coupure d'alimentation entraînait la perte de toutes les données. Les mémoires à tore de ferrite ne devaient pas apparaître avant le milieu des années cinquante.

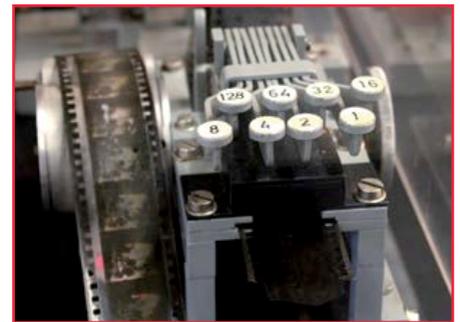
Le Z4 était capable d'exécuter environ 30 opérations par minute : une addition prenait une demi-seconde, une multiplication environ trois secondes et demie. Différents raffinements étaient à l'étude, tels que les branchements dans le programme et les registres d'index pour le calcul des adresses, mais ils furent perdus dans le chaos de la guerre.

Langage de programmation et développements après-guerre

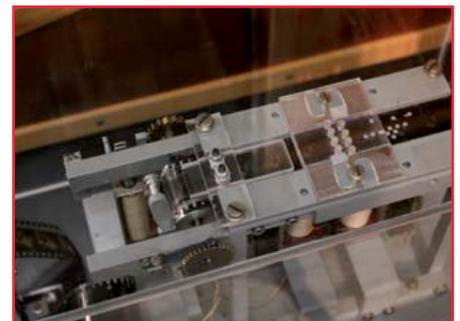
Le fait n'est pas très connu, mais Konrad Zuse était déjà conscient de la nécessité d'un langage de programmation de haut niveau. En 1945 et 1946, il développa *Plankalkül* (système formel de planification) mais ne put jamais publier ses résultats. Ces travaux préfiguraient les langages plus modernes comme Fortran, ALGOL et COBOL.

Dans les années de l'immédiat après-guerre, le Z4 connut une véritable odyssée à travers l'Allemagne, la Suisse et la France. IBM (entre autres) était intéressé par l'acquisition de la propriété intellectuelle, à seule fin d'empêcher les développements ultérieurs de la machine. Au cours des années cinquante, elle se trouva à l'Institut Technique fédéral suisse, où elle participait à la solution de problèmes de recherche scientifique : en 1950, c'était le seul ordinateur en service en Europe. En 1960, la machine termina son voyage au *Deutsches Museum* de Munich.

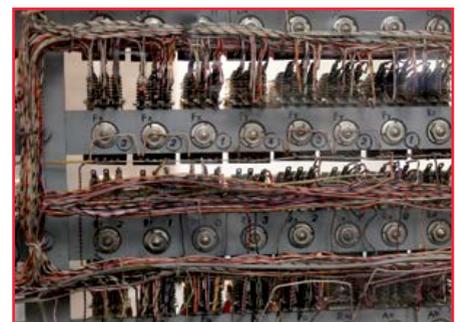
La firme Zuse continua la construction de gros systèmes scientifiques pendant plusieurs années et produisit en 1961 le tout premier traceur parfaitement



6



7



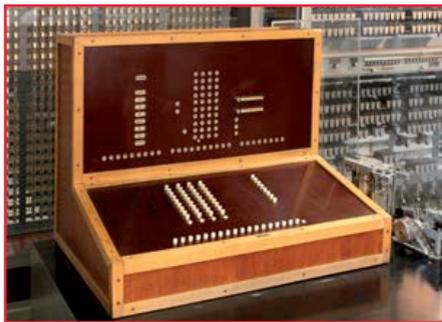
8



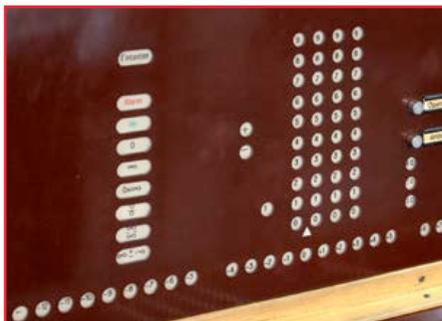
9



10



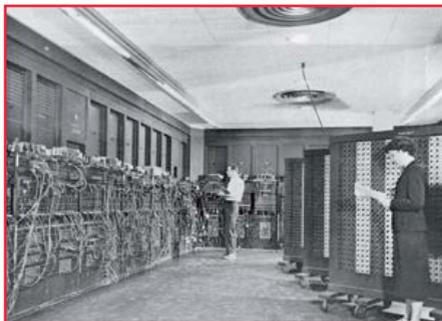
11



12



13



14



15

Comment « pense » un ordinateur comme le Z3 ?

Chaque opération revient finalement à l'addition ou à la soustraction de deux entiers. L'addition de deux nombres en virgule flottante s'effectue comme suit. On calcule d'abord la différence entre les deux exposants ; ensuite on utilise cette valeur pour décaler une des mantisses de manière à aligner les points binaires ; enfin on additionne les mantisses alignées.



K. Zuse (l) and H. Nixdorf

La soustraction s'effectue de façon similaire, avec une étape supplémentaire où l'on prend le complément à 2 de la deuxième mantisse.

La multiplication s'effectue par l'addition des deux exposants, puis la multiplication des mantisses par additions itératives.

La division est semblable à l'addition : on soustrait les deux exposants, puis on divise les mantisses par soustractions itératives.

Les racines carrées sont calculées par une méthode itérative semblable à celle de la division.

À haut niveau, l'unité arithmétique est composée de deux parties, l'une traite les opérations sur les exposants, l'autre les calculs sur les mantisses.

Les instructions effectuées par un algorithme itératif nécessitent un séquenceur pour le pilotage des différentes parties de la machine : cela correspond en gros à l'utilisation de micro-code dans les processeurs modernes.

fonctionnel, le *Graphomat*. Toutefois, la petite entreprise fut incapable de supporter la concurrence écrasante des États-Unis, elle fut rachetée en 1964 pas la firme suisse *Brown Boveri & Cie*, puis en 1967 par *Siemens AG*.

(130040 version française : Jean-Paul Brodier)

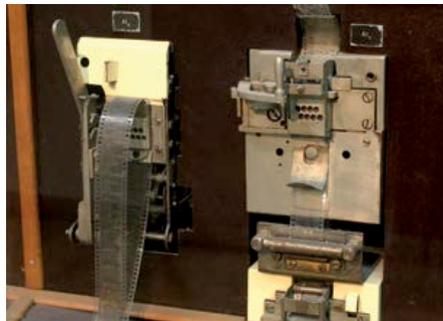
Crédits

Photos 1, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 15, 16 par l'auteur, avec l'autorisation du *Deutsches Museum* à Munich.

Photo 14 : wikipedia.de

Photo 3 : *Deutsches Technik Museum* de Berlin

Photo 2 : Professeur Horst Zuse



EST^D 2004

Retronics is a monthly section covering vintage electronics including legendary Elektor designs. Contributions, suggestions and requests are welcome; please telegraph editor@elektor.com

Hexadoku casse-tête pour électroniciens

Ce sous-titre est trompeur, car il n'est pas nécessaire d'être électronicien pour s'attaquer à une grille hexadoku, il suffit de savoir compter de 0 à 15. On se prend vite au jeu et on y prend souvent son pied. Essayez de la remplir en respectant les règles, et envoyez-nous votre solution. Vous gagnerez peut-être l'un des quatre chèques-cadeaux Elektor mis en jeu. En tout cas, même si vous ne gagnez pas, vous vous serez agréablement détendu.

Une grille hexadoku est composée de chiffres du système hexadécimal, de 0 à F. Remplissez le diagramme de 16 x 16 cases de telle façon que tous les chiffres hexadécimaux de 0 à F (0 à 9 et A à F) n'apparaissent qu'une seule et unique fois dans chaque ran-

gée, colonne et carré de 4 x 4 cases (délimités par un filet gras). Certains chiffres, déjà placés dans la grille, en définissent la situation de départ. Pour participer, inutile de nous envoyer toute la grille, il suffit de nous envoyer la série de chiffres sur fond grisé.

Participez et gagnez !

Nous tirons au sort l'une des réponses internationales correctes reçues dans les délais ; son auteur recevra un chèque-cadeau d'une valeur de **100 €** à valoir sur des **circuits imprimés Eurocircuits**. Nous offrons en outre 3 chèques-cadeaux à valoir sur des livres d'Elektor d'une valeur de **50 €** chacun. À vos crayons !

Où envoyer ?

Envoyez votre réponse (les chiffres sur fond grisé) avec vos coordonnées par courriel, télécopie ou courrier avant le **1^{er} juillet 2013** : Elektor c/o Regus Roissy CDG – Le Dôme – 1, rue de La Haye BP 12910 – 95731 Roissy CDG
Courriel : hexadoku@elektor.fr | www.elektor.fr/hexadoku

Les gagnants

La solution de la grille du numéro de mai (419) est : **3D1AE**

Le gagnant des **100€** à valoir sur des circuits imprimés **Eurocircuits** est **Dirk Neerijse** (Belgique).

Les 3 chèques-cadeaux Elektor d'une valeur de 50 € chacun vont à :

József Nagy (Baracs, Hongrie) | Sigrid Scheel (Braunschweig, Allemagne) | G. D. Joe Young (Victoria, Canada).

Bravo à tous et félicitations aux gagnants !

		6					2	4	9	C	E	8			
E		C	A		8	B								0	
2		A					F	5	8		1				
5				2		7		B			4			6	
8					0		D	9	7			F	6		
D	7			E		1	2	C	6			0		B	3
		6	0				8	F			4	7		C	
					7			B			2	E	8	4	5
3	4	9	5	2			D			A					
	6		D	4			1	2				8	A		
7	A		8			B	E	4	C		6			5	9
	0	1			6	A	9		7						2
0		8			9	6			2						E
		D		8	A	2						5			C
1								9	D		7	6			8
	3	F	7	1	B	5						9			

9	4	E	7	F	5	6	D	C	1	3	8	0	2	B	A
B	A	0	F	1	7	E	2	4	9	5	D	C	8	3	6
5	1	2	3	0	9	C	8	6	A	E	B	4	D	F	7
C	D	6	8	3	A	4	B	2	F	0	7	E	9	5	1
6	9	4	E	2	8	7	F	1	B	C	0	3	A	D	5
8	7	3	A	9	1	0	C	5	D	4	E	2	B	6	F
D	2	F	1	4	B	5	E	3	6	9	A	8	C	7	0
0	5	B	C	6	D	A	3	7	2	8	F	1	E	4	9
7	6	9	B	5	0	F	4	8	C	2	3	D	1	A	E
1	8	5	0	A	2	3	9	D	E	7	4	6	F	C	B
E	F	A	4	7	C	D	1	9	0	B	6	5	3	2	8
2	3	C	D	8	E	B	6	F	5	A	1	7	0	9	4
A	C	1	5	B	6	8	7	E	3	F	2	9	4	0	D
F	E	8	6	C	4	2	0	A	7	D	9	B	5	1	3
3	0	D	9	E	F	1	5	B	4	6	C	A	7	8	2
4	B	7	2	D	3	9	A	0	8	1	5	F	6	E	C

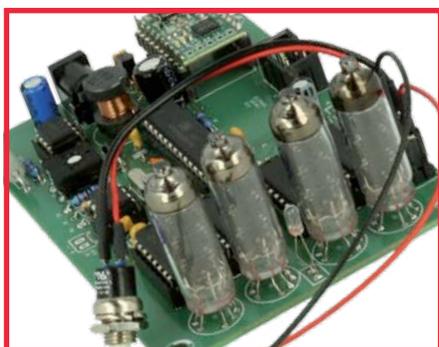
Tout recours est exclu de même que le sont, de ce jeu, les personnels d'Elektor International Media et leur famille. Un seul gagnant par foyer.

•bientôt dans Elektor



étage de puissance audio compact 200 W dans 4 Ω

L'électronique audio s'est faite rare dans les pages d'Elektor. C'est paradoxal, car le sujet intéresse de très nombreux lecteurs. Messieurs les Auteurs Audiophiles et Audiomanes, faites-nous donc des propositions intéressantes, nous les publierons. Comme ce sera le cas bientôt pour cet ampli construit autour d'un circuit intégré spécial.



horloge à tubes

Autant l'audio s'est fait discret dans Elektor, autant Arduino est envahissant, mais qui s'en plaindrait ? Ici il s'agira d'un système à µC compatible avec Arduino, complété par un circuit capable d'attaquer des afficheurs à tubes (une autre mode difficile à contenir) pour en faire une horloge-thermomètre.



serveur web Xmega

Carte polyvalente autour d'un AVR plutôt balaise. Pour communiquer : 4 LED, 4 poussoirs, un afficheur (opt.) et le choix entre RS485 et plusieurs connecteurs UART/TTL (pour connecter p. ex. la passerelle BOB USB-TTL d'Elektor. Sans oublier un connecteur d'extension, un socle micro-SD et la place pour un module TCP/IP.

Informations préliminaires non contractuelles
Parution du numéro de septembre : 21 août

Publicité

Schaeffer
AG

FACES AVANT ET BOÎTIERS

Pièces unitaires et petites séries à prix avantageux.

A l'aide de notre logiciel – Designer de Faces Avant * – vous pouvez réaliser facilement votre face avant individuelle. GRATUIT: essayez-le! Pour plus de renseignements, n'hésitez pas à nous contacter, des interlocuteurs français attendent vos questions.

*Vous en trouverez la dernière version sur notre site internet.

- Calcul des prix automatique
- Délai de livraison: entre 5 et 8 jours
- Si besoin est, service 24/24



Exemple de prix: 34,93€
majoré de la TVA/
des frais d'envoi

Schaeffer AG · Nehmitzer Damm 32 · D-12277 Berlin · Tel +49 (0)30 805 86 95-0
Fax +49 (0)30 805 86 95-33 · Web info@schaeffer-ag.de · www.schaeffer-ag.de

elektor  **labs**
Sharing Electronics Projects

Home News Proposals In Progress Finished



get started
with the
LPC800

About Elektor.LABS

Proposals In Progress Finished

Create a Project

elektor  **labs**

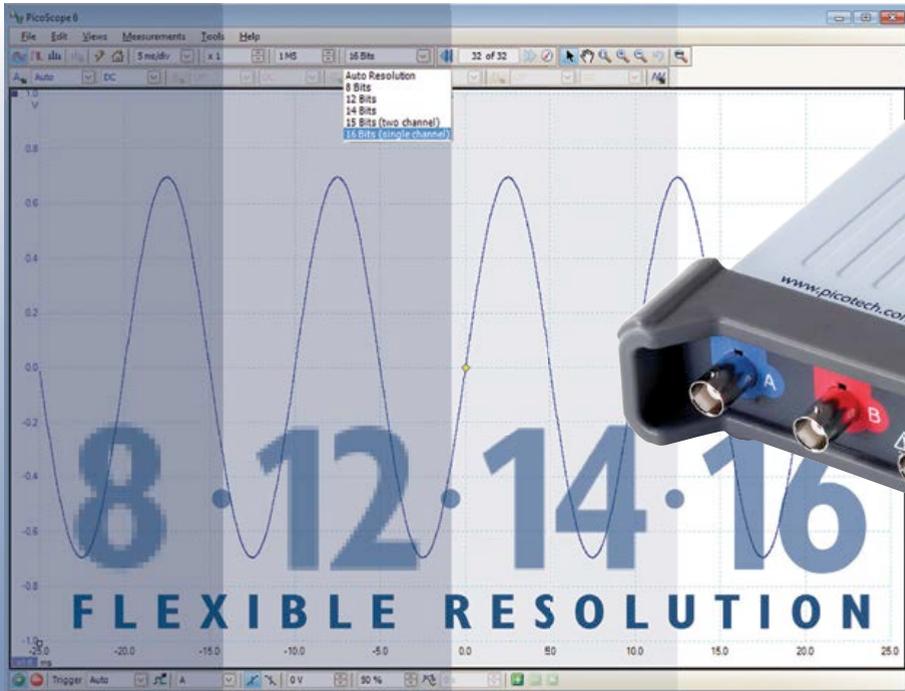
est ouvert

24 heures sur 24

7 jours sur 7

Participez à l'élaboration des
projets sur
www.elektor-labs.com

POURQUOI FAIRE UN COMPROMIS PRÉCISION ou VITESSE QUAND VOUS POUVEZ TOUT AVOIR ?



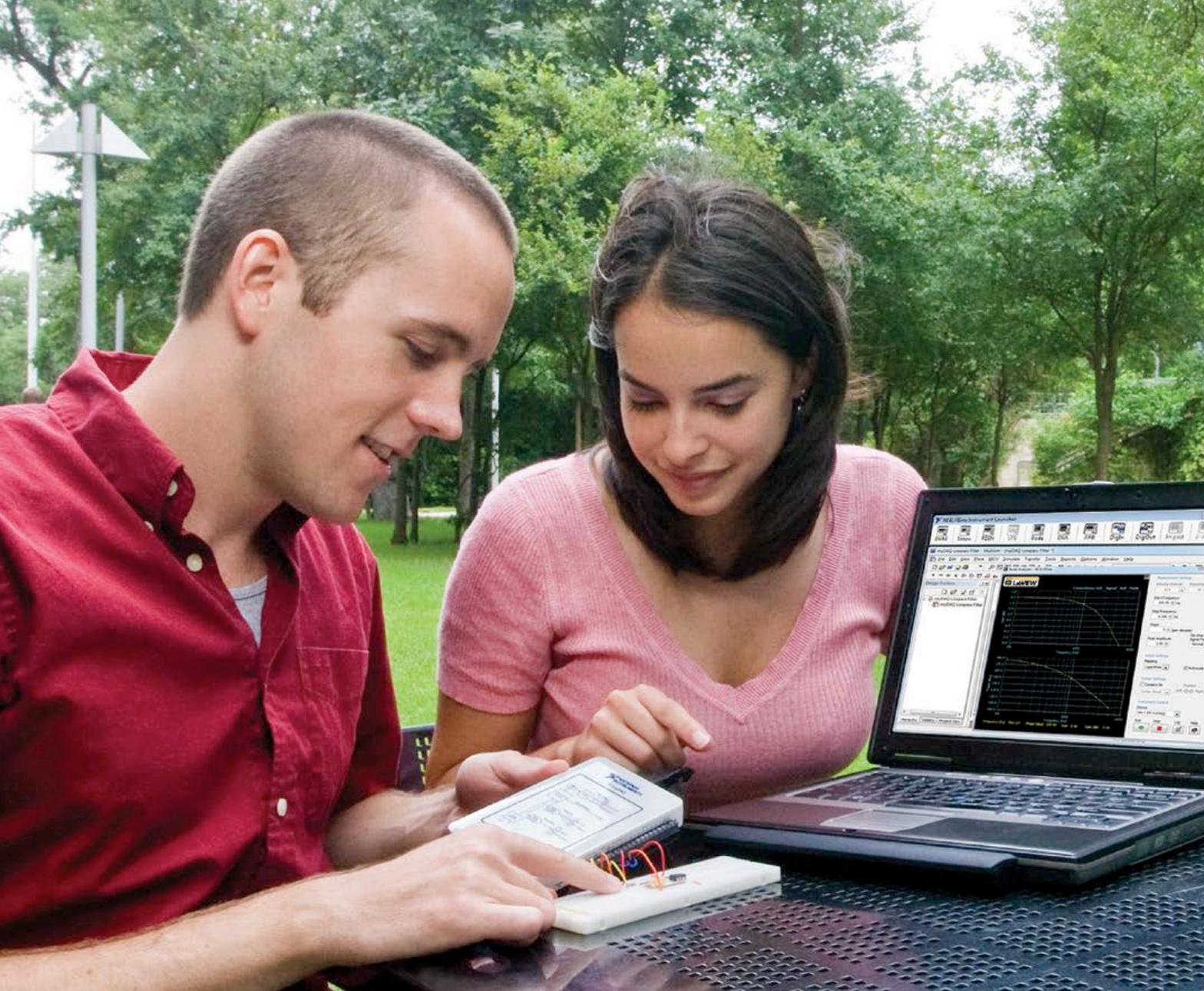
PicoScope	PicoScope 5442A	PicoScope 5442B	PicoScope 5443A	PicoScope 5443B	PicoScope 5444A	PicoScope 5444B
Canaux	4					
Bande passante	tous les modes: 60 MHz		modes de 8 à 15 bits: 100 MHz modes à 16 bits: 60 MHz		modes de 8 à 15 bits: 200 MHz modes à 16 bits: 60 MHz	
Taux d'échantillonnage	2.5 GS/s		5 GS/s		10 GS/s	
Mémoire (8 bits)*	16 MS	32 MS	64 MS	128 MS	256 MS	512 MS
Mémoire (≥ 12-bit)*	8 MS	16 MS	32 MS	64 MS	128 MS	256 MS
Résolution (amélioré)**	8 bits, 12 bits, 14 bits, 15 bits, 16 bits (resolution du matériel + 4 bits)					
Générateur de signal	Générateur de fonctions	AWG	Générateur de fonctions	AWG	Générateur de fonctions	AWG

Canal 2 modèles aussi disponibles * partagée entre les canaux actifs ** La résolution maximale est limitée sur les plages les plus bas de tension: $\pm 10 \text{ mV} = 8 \text{ bits}$ • $\pm 20 \text{ mV} = 12 \text{ bits}$. Toutes les autres plages peuvent utiliser la résolution maximale.

L'OSCILLOSCOPE À RÉOLUTION SOUPLE



TOUS LES MODÈLES INCLUENT SONDAS, LE LOGICIEL COMPREND: AINSI QUE 5 ANS DE GARANTIE. LE LOGICIEL COMPREND DES: MESURES, ANALYSEUR DE SPECTRE, FULL SDK, DÉCLENCHEMENTS AVANCÉS, PERSISTANCE DE COULEURS, DÉCODAGE SÉRIE (CAN, LIN, RS232, I²C, I²S, SPI), MASQUES, MATHS, LE TOUT EN STANDARD. MISES À JOUR GRATUITES.



Au service du génie

National Instruments met à la disposition des étudiants le matériel et le logiciel dont ils ont besoin pour développer leur expérience, aller au-delà de la théorie, et réaliser l'importance du rôle de l'ingénieur dans la société.

>> Découvrez comment NI supporte la prochaine génération d'innovateurs, en visitant ni.com/academic/f

01 57 66 24 24

NATIONAL INSTRUMENTS France ■ 2 rue Hennape - 92735 Nanterre Cedex, France ■ Tél. : 01 57 66 24 24 ■ Fax : (0)1 57 66 24 14 ■ Société de droit américain - capital social 1 000 dollars ■ US ■ 11500 N Mopac Expwy, Austin-Texas USA - 10056236 - 344 497 649 ■ RCS Nanterre - SIRET B 344 497 649 00048 - APE 516J - N.J.I. FR 57344497649

©2013 National Instruments. Tous droits réservés. National Instruments, NI, et ni.com sont des marques de National Instruments. Les autres noms de produits et de sociétés mentionnés sont les marques ou les noms de leurs propriétaires respectifs. Pour plus d'informations concernant les marques de National Instruments, veuillez vous référer à la partie Terms of Use sur le site ni.com/legal. 09680

 **NATIONAL
INSTRUMENTS**