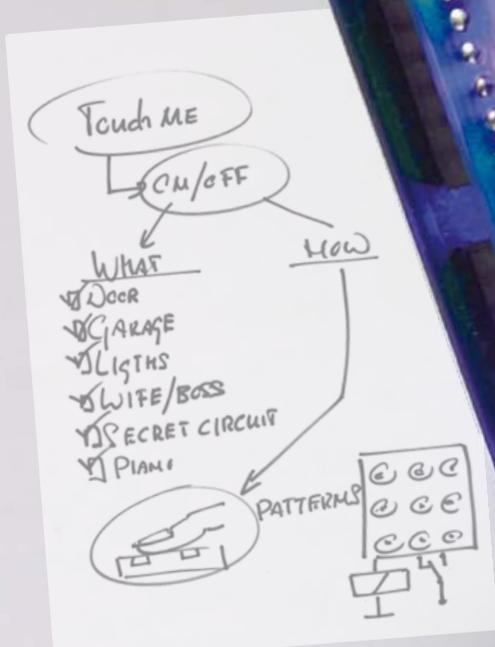
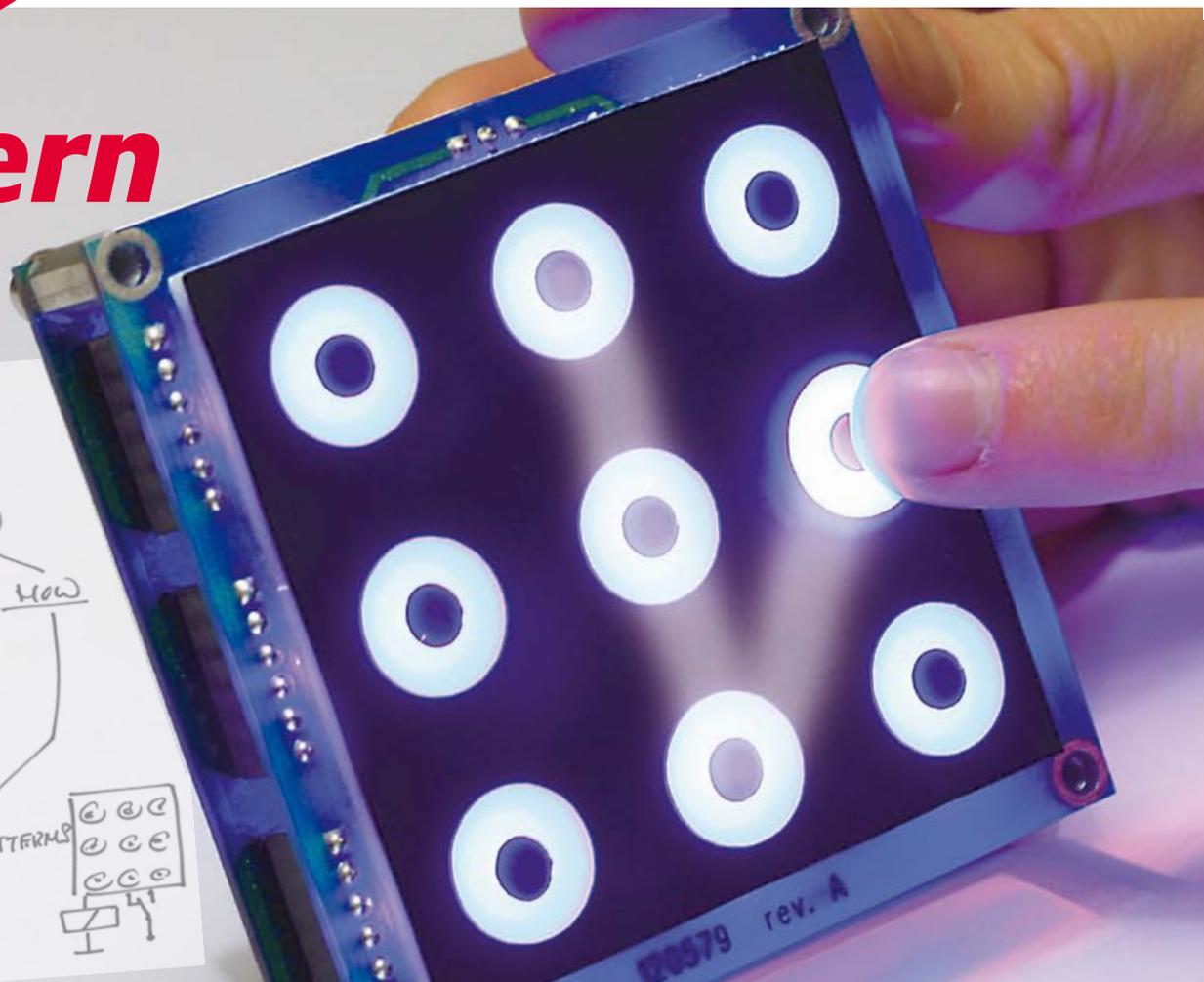


elektor

Pattern Lock



verrou tactile à reconnaissance de motif

- **carte d'extension Linux** | guidon de moto chauffant
- LCR-mètre 2^e partie | **Arduino commande une webcam**
- PIColino : prototypage éclair | **diagrammes d'états-transitions** ● kit Audion
- Radioman 1956 ● des nouvelles du chaudron



e explorez

le monde de l'électronique



avec les cinq nouveaux
vaisseaux de la flotte Elektor

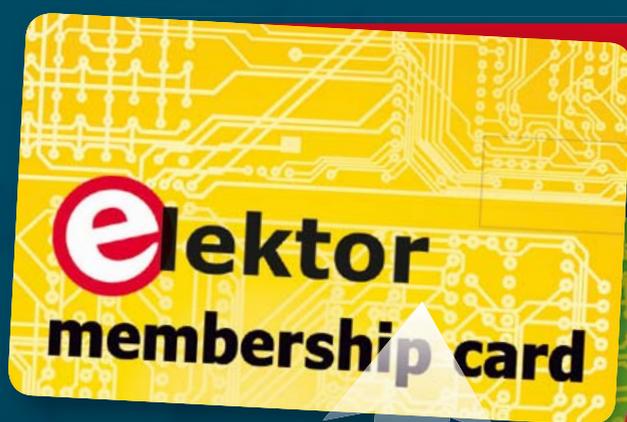
elektor.magazine : **le vaisseau amiral**

elektor.labs : **le drakkar**

elektor.community : **le paquebot**

elektor.post : **le catamaran**

elektor.store : **le cargo**

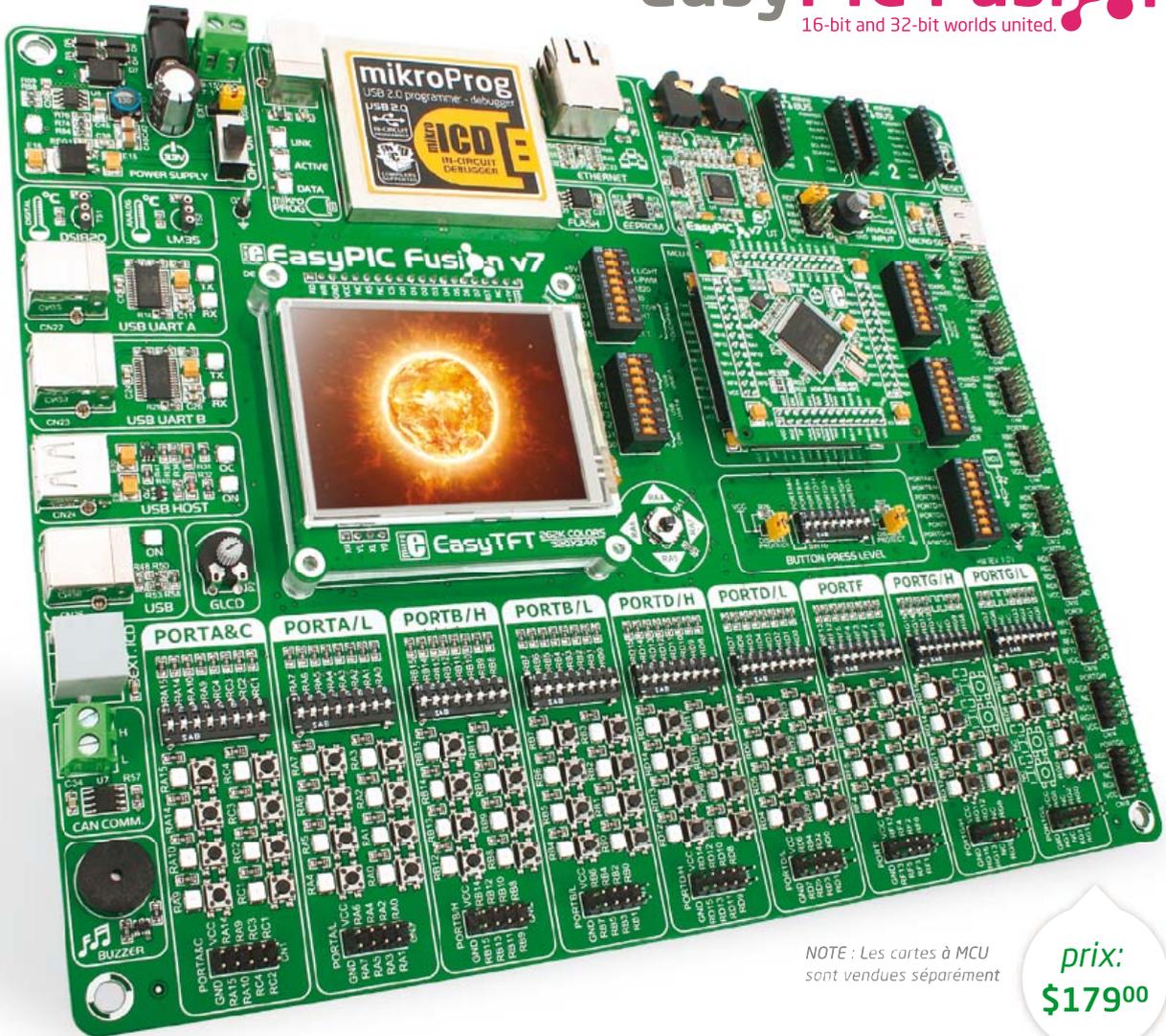


Rejoignez la communauté Elektor :
embarquement immédiat !



avec puce NFC Mifare programmable

Découvrez nos formules d'abonnement sur www.elektor.fr/abo



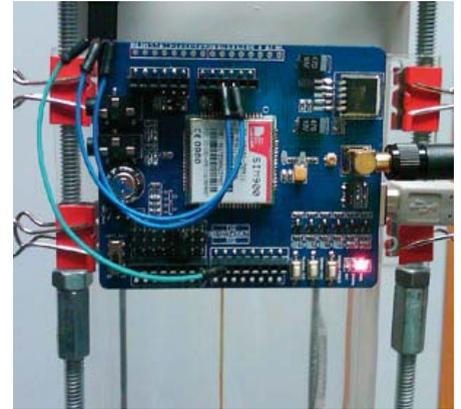
NOTE : Les cartes à MCU
sont vendues séparément

prix:
\$179⁰⁰



EN COMPÉTITION POUR LE **embedded AWARD 2013** Dans la catégorie Meilleurs Outils

Nous sommes honorés et fiers de compter parmi l'élite des compagnies dont le succès est assuré par des produits reconnus comme les plus innovants de l'année. EasyPIC Fusion v7 mérite assurément d'être reconnu comme la seule carte de développement qui supporte trois architectures de microcontrôleurs et réunisse des micros à 16 et 32 bits sur la même station de travail.



● communauté

8 le monde d'Elektor

- Elektor à l'école
- Elektor en Thaïlande
- des bons points pour Elektor
- sous la caboche des lecteurs de CC



● industrie

10 nouveaux produits

12 redresseur à faible fuite
avec un AOP auto-alimenté

● Projects

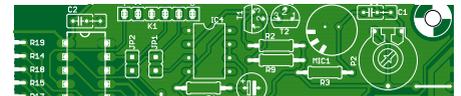
14 Piccolino :
prototypage ultrarapide
Développer des circuits à PIC et les tester aussitôt.

18 carte d'extension Linux
Afficheur, boutons, horloge RTC, buzzer et 16 GPIO pour saisir rapidement un paramètre ou connaître l'état du processus.

24 Pattern Lock :
clavier tactile à reconnaissance de motif
Élégante serrure tactile rétro-éclairée, à partir de la méthode de détection QTouch d'Atmel.

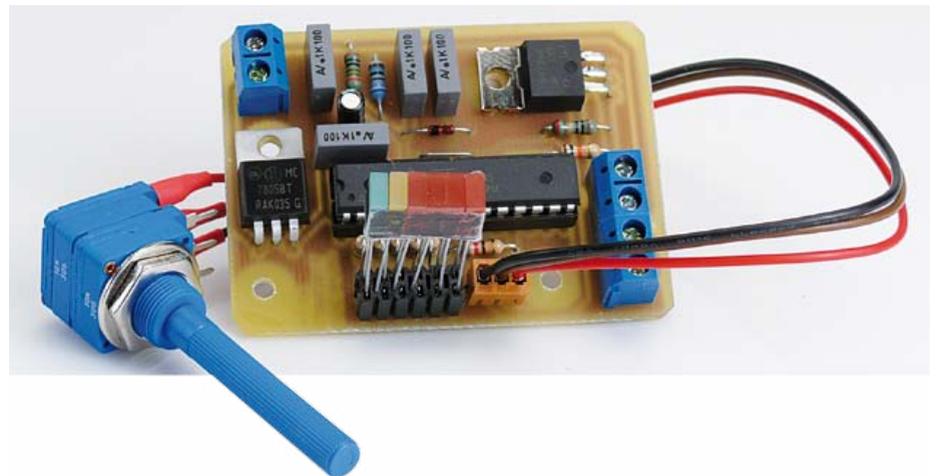
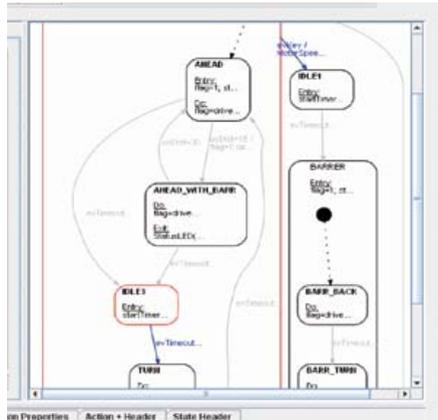
36 mettez le cap sur Arduino (5)
Une application ludique pour prendre des photos automatiquement.

46 commande de guidon chauffant
pour motos et gyropodes
Les µC sont partout, jusque dans les poignées chauffantes de moto.



48 suiveur de tension parcimonieux
Pour ne pas charger la source qu'il surveille, un circuit de suivi de tension doit consommer le moins possible.

50 mon disque bosse dur dix LED pour en témoigner
Un afficheur à dix LED donne une image précise de l'activité de votre disque dur.



52 du code de pros avec les diagrammes d'états-transitions

De plus en plus utilisés en génie logiciel, ils favorisent une programmation robuste.

56 LCR-mètre 0,05 % 2e partie

L'extension afficheur et clavier pour utiliser l'appareil sans PC, les fonctions de l'appareil par ses menus et l'afficheur et le logiciel.

Tech the Future

66 données portes ouvertes

Des hackers, des programmeurs, des chercheurs et des décideurs hollandais réunis en vue de bidouiller les données de leur Parlement.

labs

42 quartz muet

Il n'y a pas que les CMS qui peuvent poser des problèmes d'implantation.

42 qu'est-ce qui mijote au labo ?

Trois prototypes du LCR-mètre

43 régulateurs de tension à découpage avec composants traversants

Une version meilleur marché des régulateurs à découpage ?

44 flux continu audio

Test du Streamer Audio de Future Electronics

70 des nouvelles du chaudron

Quelques questions fréquentes et leurs réponses

Magazine

6 de nous à vous

le réseau international d'Elektor

72 changez de loisirs, devenez dompteur de puces

Entretien avec l'auteur du nouveau livre « maîtrisez les microcontrôleurs avec Arduino ».



76 kit Audion Radiomann (vers 1956)

La radio, objet de fascination absolue au début des années 1950.

79 hexadoku

Casse-tête pour électroniciens

82 bientôt dans Elektor

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

du mardi au 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.

Boîtes de pralines

La matière micro-électronique s'enrichit, sa maîtrise devient ardue et il se dessine pourtant un mouvement diamétralement opposé, qu'on pourrait appeler de la *facilitation* : jamais l'adjectif *facile* n'a été plus utilisé à propos des μ C en général et des 32 bits en particulier, des FPGA et de Linux. Désormais, tout est facile, et même *rapide et facile* : programmation, conception, prototypage... Les composants les plus complexes sont censés s'assembler comme du Lego : *sans connaissances ni études approfondies*



promet la pub. Les cartes interconnectables, rapetissées, se présentent sous forme d'ensembles cohérents, alignées comme des pralines alléchantes dans leur boîte, ludiques comme des boîtes de construction. Leurs interfaces, comme des ports accueillants, ouvrent ces circuits sur une mer d'autres circuits et de logiciels compatibles. Sur ce point, Elektor devrait faire des efforts et rendre compatibles entre eux les projets présentés au fil des numéros. Tâche difficile du fait de la diversité des auteurs et de leurs préoccupations. Heureusement, les nouvelles modalités communautaires de gestation et de production de nos réalisations, autour et dans le chaudron d'**Elektor.LABS**, favoriseront cette intégration et cette homogénéisation. Et ça, c'est aussi et surtout **VOTRE** affaire, car **Elektor.LABS, c'est vous**. Dans la composition de la revue, en revanche, une bonne dose d'hétérogénéité ne saurait nuire, comme en témoignent les diverses *pralines* de ce numéro. Divergence du format des articles : de la modeste demi-page aux fleuves d'une dizaine de pages. Divergence des sujets : ici les subtilités d'un redresseur auto-alimenté, là les finesses du LCR-mètre (déjà minutieusement présenté le mois dernier), ou encore (p.66) les idées novatrices sur l'*ouverture des données* du parlement d'une démocratie occidentale ; une forme de civisme à imiter ?

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, Raymond Vermeulen, 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
+31 46 4389417
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
+27 78 2330 694
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 ELECTRONICS EDUCATION

audio X PRESS



vous connecte à



Nos annonceurs



Beta Layout
www.pcb-pool.com13



National Instruments84
www.ni.com/measurements-platform/f



Eurocircuits
www.elektorpcbservice.com65



Reichelt
www.reichelt.fr35



MikroElektronika
www.mikroe.com 3

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

Elektor World

compilé par
Wisse Hettinga

Chaque heure du jour et de la nuit, quelque part dans le monde, Dame Électronique accouche d'une nouvelle trouvaille, aidée par les mains expertes d'un Obstétrix amateur ou professionnel, en tout cas fier et passionné. Certaines de ces trouvailles changeront le cours du monde, d'autres procurent juste quelques frissons de plaisir. Ainsi va le monde de l'électronique, le monde d'Elektor.



Bons points certifiés

Elektor propose des formations dans certains pays (Allemagne, Pays-Bas...) pour lesquelles nous avons obtenu le label CEDEO. Nous en sommes très honorés, mais ce qui nous a touchés bien davantage, ce sont les avis satisfaits de 80 % des participants aux ateliers et autres séances de formation proposés par Elektor. L'institut indépendant EDEO certifie les formations professionnelles payantes aux Pays-Bas. Avec ce certificat dont Chantalle est très fière, à juste titre, Elektor figure maintenant sur la liste des 800 prestataires bataves reconnus officiellement !

24e Semaine de la presse et des médias dans les écoles

700 exemplaires du numéro de mars d'Elektor ont été diffusés gratuitement dans les écoles de France et de Navarre qui en avaient fait la demande dans le cadre de la Semaine de la presse (du 25 au 30 mars). Chaque année, au printemps, les enseignants de tous niveaux et de toutes disciplines sont invités à participer à cette activité d'éducation civique. Le but est d'aider les élèves, de la maternelle aux classes préparatoires, à comprendre le système des médias, à former leur jugement critique, à développer leur goût pour l'actualité et à forger leur identité de citoyen. Elektor y participe régulièrement.

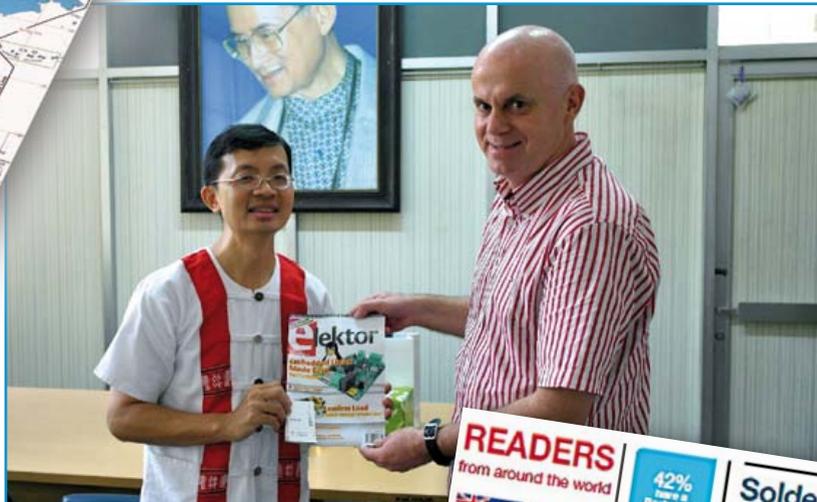


24^e SEMAINE DE LA PRESSE ET DES MÉDIAS DANS L'ÉCOLE®



Elektor à Bangkok

Un des secrets les mieux gardés d'Elektor est son vaste réseau de correspondants dans le monde entier et dans les professions les plus variées, même si la plupart sont étudiants, concepteurs, techniciens ou professeurs, comme le professeur David Banjerdpongchai, chef du département de génie électrique (*Electrical Engineering*) de l'Université Chulalongkorn de Bangkok, en Thaïlande, avec la première édition thaïe d'Elektor. Bienvenue à bord, David !



Dans la caboche des lecteurs de CC

E. HARTFORD, CT, États-Unis— L'équipe de *Circuit Cellar* a mené récemment une enquête auprès de 1000 électroniciens et universitaires sur leurs préférences. Combien soudez-vous ? Combien connaissez-vous de langages de programmation ? Utilisez-vous des FPGA ? Quels sont les meilleurs fournisseurs de micro-informatique embarquée ? Et d'autres questions du même ampérage !

Les résultats sont fascinants :

<http://bit.ly/11Qzflp>.

Les lecteurs de *Circuit Cellar* résident aux quatre coins de la planète : États-Unis, bien sûr, Brésil, Inde, Allemagne, Royaume-Uni, Japon... et France, bien sûr. La barrière de la langue ne résiste pas à l'intérêt pour la micro-informatique.



Mindstorms Education EV3

Fort du succès du kit de robotique *Mindstorms NXT*, *LEGO Education* propose maintenant la plate-forme *LEGO MINDSTORMS Education EV3*, conçue pour la salle de classe à partir des réactions de plus 800 enseignants du monde entier suite à l'utilisation du kit NXT.

Destinée à offrir aux élèves une expérience pratique de la programmation d'applications intégrant des capteurs et des actionneurs, la plate-forme EV3 est basée, comme NXT, sur l'environnement de programmation LabVIEW de *National Instruments*, bien connu des ingénieurs et scientifiques.

Le corps humain comme canal de communication sécurisé et à basse consommation

Points clés :

- Plateforme fournissant une solution de communication de courte portée et à faible débit pour une connexion sécurisée dans une vaste gamme d'applications sans fil
- Supporte l'authentification bidirectionnelle pour les technologies de cryptages avancées
- Augmente l'autonomie des batteries en éliminant la présence d'un émetteur/récepteur sans fil ou de champs inductifs haute puissance
- Le kit de développement BodyCom et sa plateforme de développement gratuite V1.0 sont disponibles



La **technologie BodyCom** de *Microchip* fait appel au couplage capacitif avec le corps humain et permet aux concepteurs d'utiliser ce corps comme canal de communication sécurisé. Comparée aux méthodes sans fil actuelles, la technologie *BodyCom* a une très faible consommation tout en renforçant la sécurité grâce à une authentification bidirectionnelle. L'absence d'antenne RF simplifie la conception du circuit. La plateforme de développement *BodyCom V1.0* est fournie avec une bibliothèque gratuite de logiciels qui fonctionnent avec tous les microcontrôleurs PIC de 8, 16 et 32 bits (plus de 900 !).

Le système établit les communications bidirectionnelles entre un contrôleur centralisé et une ou plusieurs unités sans fil. Pour de nombreuses applications, une communication sans fil sécurisée est vitale, or il n'y a pas de canal plus sécurisé que celui du corps humain, surtout avec l'authentification bidirectionnelle comme support de cryptage avancé (Keeloq ou AES).

La technologie *BodyCom* augmente considérablement l'autonomie de la batterie en éliminant la présence d'un émetteur/récepteur sans fil ou de champs inductifs de forte puissance. Elle permet l'utilisation d'un microcontrôleur ordinaire et des fréquences AFE standards (125 kHz et 8 MHz), sans quartz externe. Le coût et la complexité de la certification sont éliminés en raison de la conformité aux recommandations du FCC (Part 15-B) sur les émissions par rayonnement.

Voici quelques exemples d'applications :

Le contrôle d'accès : systèmes de sécurité, verrouillage des portes pour les humains ou les animaux ;

La sécurité et la protection corporelle : mise en/hors service d'appareils, d'outils, d'armes à feu et de systèmes informatiques ;

L'équipement médical : la surveillance de patient, l'accès aux chambres d'un hôpital ou la traçabilité d'un appareil ;

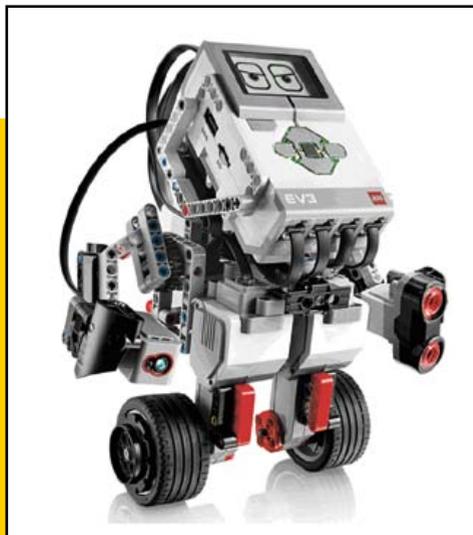
L'électronique grand public : la gestion de profil pour les consoles de jeu et d'appareils d'entraînement.

www.microchip.com/get/9TMM www.microchip.com/get/GA5E

L'environnement logiciel, toujours aussi simple d'utilisation, se distingue de la génération précédente notamment par l'adjonction de nouvelles capacités d'enregistrement de données qui permettent aux élèves de recueillir et traiter leurs données et d'en faire des graphiques.

Le kit est livré avec un programme d'apprentissage personnalisable, un cahier numérique pour prendre des notes et suivre les progrès des élèves, un programme d'étude comportant 30 h d'enseignement en salle de classe. Un kit d'expansion est également prévu pour permettre aux élèves de construire des robots plus complexes. « Prévu », car si cette plate-forme peut être pré-commandée, il faudra attendre le seconde semestre 2013 pour la disponibilité effective.

www.LEGOeducation.com/MINDSTORMS
poweredby.labview.com/



reichelt : sécurité et confort en un

En domotique, les fonctions de sécurité et de confort sont souvent commandées par des systèmes différents. Avec le système d'automatisation domestique et de radio alarme HIS20S d'ELRO, *reichelt elektronik* propose désormais un système qui combine les deux champs d'application.

Le cœur de l'installation est la centrale de commande avec écran LCD et les touches éclairées Soft-Touch. L'appareil peut prendre en charge jusqu'à 32 récepteurs, par exemple des capteurs, des commutateurs et des détecteurs. La commande intelligente permet de programmer des scénarios particuliers, p. ex. l'absence nocturne avec alarme activée, l'éclairage à commande temporelle ainsi que la fermeture ou l'ouverture automatique des volets roulants. *reichelt elektronik* propose, avec le HIS20S, un kit de démarrage qui comprend la centrale de commande, un module radio-USB pour la gestion sans fil du système depuis le PC, un récepteur à brancher ainsi qu'un bouton mural pour allumer l'éclairage, une radiocommande, un détecteur magnétique pour la sécurité d'une porte ou d'une fenêtre, un détecteur de mouvements (PIR=infrarouge passif) ainsi qu'un logiciel convivial pour régler, configurer et gérer le système. Il est possible d'ajouter d'autres interrupteurs d'éclairage pour l'intérieur et l'extérieur, des capteurs de mouvements et des capteurs magnétiques, des barrières photo-électriques à infrarouge, des détecteurs de fumée et d'eau, des interrupteurs encastrés pour les volets roulants ainsi qu'un radio-clavier. Tous les composants communiquent entre eux sans fil par radio.



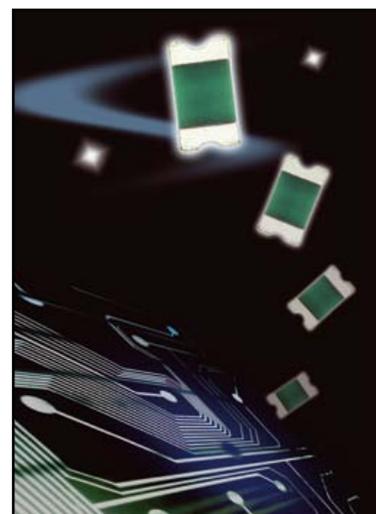
AVX : Condensateur RF multicouches organique à facteur Q élevé et faible ESR

Cette série de condensateurs multicouches organiques (MLO) de haute qualité est idéale pour les amplificateurs de puissance RF, les réseaux de filtres et les systèmes IRM.

AVX Corporation a développé une série de condensateurs en boîtier CMS de taille 0603, de tension nominale 500 Vdc et 250 Vdc, supportant de 1 MHz à plus de 5 GHz. Les condensateurs RF avancés à électrolyte organique de la série MLO3 conservent un facteur Q élevé, une résonance propre élevée et une faible ESR en haute fréquence.

La possibilité de fabriquer ces composants sur des substrats de grande surface, grâce à une technologie sophistiquée d'imagerie directe par laser, améliore les coûts et le contrôle des tolérances. Le résultat final est un condensateur RF bas de profil et à faible ESR. De plus, le coefficient d'expansion thermique du condensateur MLO est adapté à celui des cartes à circuits imprimés, pour plus de fiabilité.

Ces condensateurs RF conformes RoHS de tension nominale de 50 Vdc à 500 Vdc offrent des capacités de 0,1 pF à 2,5 pF avec une tolérance atteignant +/- 0,02 pF. Disponible en boîtiers 0603 avec une finition 100% étain, la série MLO3 présente des caractéristiques de diélectrique NPO stables pour une plage de température de fonctionnement de -55°C à +125°C.



redresseur à faible fuite avec un AOP auto-alimenté

Martin Tomasz,
chercheur chez Touchstone
Semiconductor, Inc.

Ce circuit redresseur possède une tension de seuil moindre que celle d'une diode (**fig. 1**) et comprend un ampli op à très faible consommation, un MOSFET canal P à faible seuil et deux résistances de retour. La tension redressée alimente la partie active du circuit et aucune alimentation supplémentaire ne sera nécessaire ; le courant de repos du circuit est inférieur au

courant de fuite en polarisation inverse de la plupart des diodes Schottky. Ce circuit permet un redressement actif pour des tensions aussi basses que 0,8 V. Aux tensions plus basses, la diode de corps du MOSFET prend le relais telle une diode ordinaire.

Lorsqu'une tension directe règne entre l'entrée et la sortie, l'ampli op rend le MOSFET passant, selon l'équation suivante :

$$V_{GATE} = V_{OUT} - (R_2 / R_1) \times (V_{IN} - V_{OUT})$$

où V_{GATE} est la tension de grille du MOSFET, V_{IN} la tension d'entrée, et V_{OUT} celle de sortie. Les relations entre tensions d'entrée et de sortie et tensions drain-source et grille-source du MOSFET sont décrites par les équations suivantes :

$$V_{DS} = V_{IN} - V_{OUT} \text{ et } V_{GS} = V_{GATE} - V_{OUT}$$

où V_{DS} est la tension drain-source et V_{GS} la tension grille-source. Algébriquement, ces équations relient la commande de la grille du MOSFET à une fonction de la tension drain-source :

$$V_{GS} = -(R_2 / R_1) \times V_{DS}$$

Un bon choix est de prendre pour R_2 une valeur 12 fois plus élevée que R_1 , cela donne une chute de tension de 40 mV entre le drain et la source du MOSFET, suffisante pour le rendre passant aux faibles courants de drain (**fig. 2**). Un rapport plus élevé réduit davantage la chute de tension jusqu'à la valeur limite de la tension de décalage de l'ampli op : 6 mV.

L'alimentation de l'ampli-op (*Touchstone Semiconductor TS1001*) vient du condensateur réservoir C_1 en sortie du circuit. L'ampli possède des entrées et sorties de type rail à rail et ne souffre pas d'inversion de phase lorsqu'il fonctionne près des rails d'alimentation, c'est pour cela que l'on peut relier sans encombre sa sortie à son entrée d'alimentation. De plus, comme il se contente d'une tension d'alimentation de seulement 0,8 V, le redressement actif fonctionnera jusqu'à cette limite. Son entrée non-inverseuse est reliée au rail V_{DD} et sa sortie à la grille du MOSFET. Le circuit

Figure 1.
Ce circuit simule un redresseur, mais possédant une tension de déchet de seulement 40 mV ou moins. Le circuit possède même un courant de fuite en polarisation inverse moindre que pour une diode Schottky.

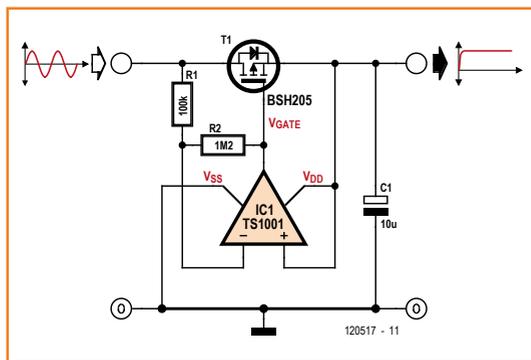


Figure 2.
La sortie du circuit (violet) avec une tension sinusoïdale en entrée (bleu) montre que la tension de grille du FET (jaune) chute seulement lorsque la différence de tension entre entrée et sortie est inférieure à 40 mV.

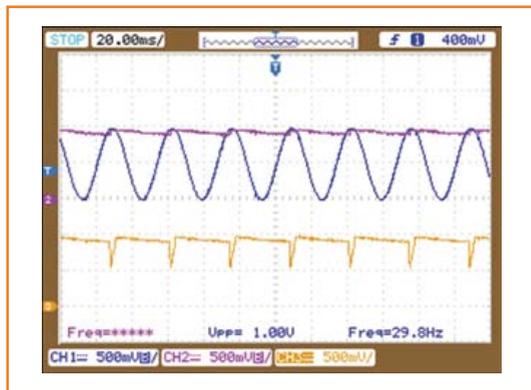
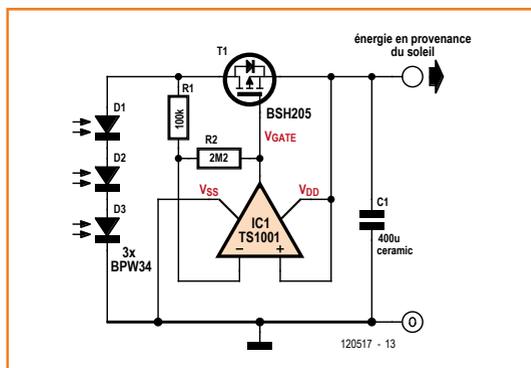


Figure 3.
Vous pourrez utiliser le redresseur actif pour charger un condensateur à partir de photodiodes. Celui-ci possède une faible tension de déchet et protège les diodes des courants inverses lorsqu'il n'y a pas de lumière.



consomme un peu plus de 1 μA lors du redressement actif d'un signal sinusoïdal à 100 Hz, moins que le courant de fuite de la plupart des diodes Schottky. Le seuil du BSH205 est suffisamment bas pour supporter des courants de l'ordre du milliampère sous une tension grille-source de 0,8 V. La bande passante de l'ampli op le cantonne aux signaux de faible fréquence ; aux fréquences supérieures à 500 Hz, le gain de l'ampli commence à décliner. Cependant, lorsque la fréquence augmente, la diode de corps du MOSFET se charge du redressement et le MOSFET reste bloqué. Des signaux d'entrée possédant un temps de descente faible pourraient potentiellement entraîner le passage d'un courant inverse à travers le MOSFET, avant que l'ampli op ne puisse commander son blocage. Pour les faibles courants, le MOSFET fonctionne en dessous de sa tension de seuil et l'ampli se coupe rapidement à cause de la relation exponentielle entre la tension grille-source et le courant drain-source. La vitesse de balayage de 1,5 V/ms de l'ampli est le

A propos de l'auteur

Martin Tomasz est un ingénieur chevronné en électronique analogique, radio et à signaux mixtes, riche de 22 ans d'expérience de la conception de circuits et de systèmes. Chercheur chez *Touchstone Semiconductor*, Martin a 17 brevets au compteur, et deux hivernages en Antarctique pour l'*US Antarctic Program*, afin de travailler sur la surveillance d'appareils conçus pour des environnements extrêmes.

facteur limitant. Tant que la charge du circuit est suffisamment faible pour que le MOSFET ne fonctionne pas dans sa zone de linéarité, les courants inverses n'excéderont pas les courants directs. Une application de récupération d'énergie solaire de très faible puissance est présentée en **figure 3**. Les BPW34 produisent de 10 à 30 μA sous 0,8 à 1,5 V, suivant leur illumination. Ce circuit redresse le pic de la tension produite lorsque l'illumination varie rapidement et minimise les fuites inverses vers les photodiodes lorsque leur illumination est faible.

(120517 - version française : Kévin PETIT)

Publicité

PCB-POOL
L'ORIGINAL DEPUIS 1994
Beta LAYOUT

create : electronics

Pochoir gratuit
avec chaque commande
"Prototype"

Service Assemblage
A partir d'un composant

Embedded RFID
authentifiez, suivez et
protégez votre produit
www.magic-pcb.com

NOUVEAU!

Appel Gratuit : FR 0800 90 33 30
sales@pcb-pool.com

PROTEUS | Foot 2000 | VERBENA | Easy-PCB | cadence | EDWIN | GreenGate | PULSONIX | DIPTRACE

Beta
LAYOUT
create : electronics

PCB-POOL est la marque déposée de Beta LAYOUT GmbH. Toute marque déposée appartient à son propriétaire respectif.

eSTORE
Beta LAYOUT

Développer, assembler, souder

€ 18,60

Iteaduno Leonardo V1.0 compatible

Reflow Controller
€ 129,00

Starter Kit UHF RFID Basic
€ 279,00

Le grand Beta-Reflow-Kit
€ 129,00

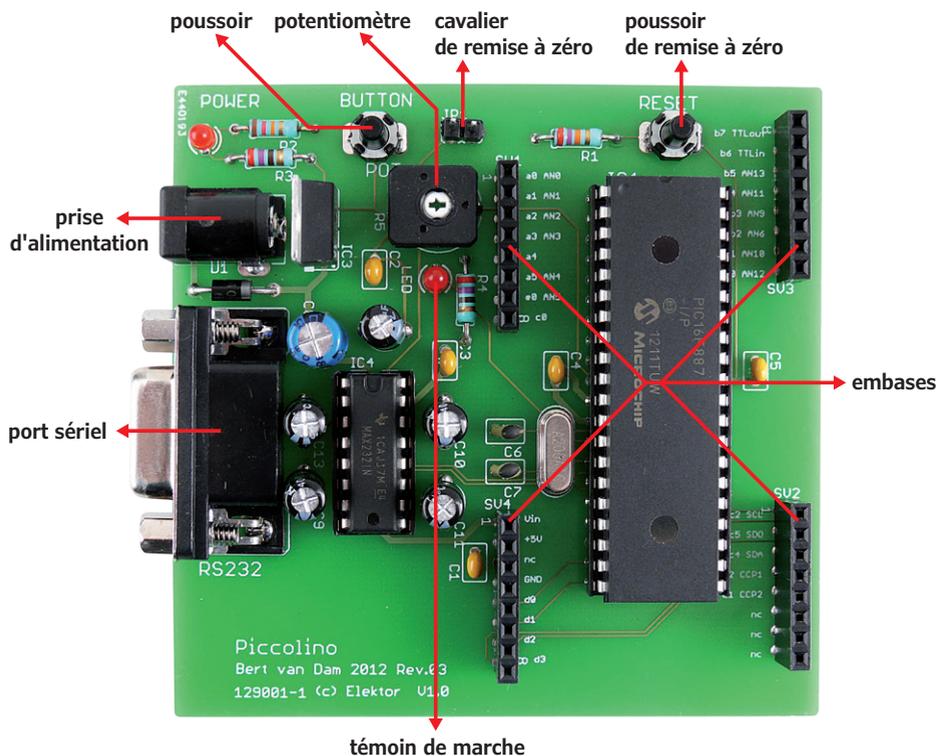
www.beta-eSTORE.com

Beta
LAYOUT
create : electronics

eSTORE est la marque déposée de Beta LAYOUT GmbH.

PICcolino

carte de prototypage ultrarapide



Bert van Dam (Pays-Bas)

PICcolino est une nouvelle plateforme de prototypage utile pour développer des circuits à microcontrôleur PIC et les tester aussitôt. Il y a tout ce qu'il faut comme matériel de base sur la carte, avec des connecteurs prévus pour y brancher aisément des composants additionnels.

Figure 1.
Les différents composants sur la carte Piccolino.

L'objectif de la carte PICcolino est de mettre à disposition des embases sur lesquelles on peut simplement enficher des modules d'extension. Si vous utilisez fréquemment des composants déterminés sur vos prototypes, faites-en directement un module d'extension. Et si vous prévoyez sur ce module d'autres connecteurs, vous pourrez continuer d'empiler comme bon vous semble. Sans radiateur sur le régulateur, Piccolo peut fournir à peu près 100 mA sur les embases. Le microcontrôleur 16F887 de Microchip est installé dans un support, on peut donc en principe en changer. Ce n'est pas tout à fait aussi simple, parce que ce PIC contient un chargeur d'amorce spécial qui lui permet de se programmer lui-même. Si vous voulez utiliser un autre PIC, il faudra le doter aussi d'un chargeur d'amorce, sinon PICcolino ne marchera plus ! Mais vous pouvez vous procurer auprès d'Elektor des sucédanés de ce PIC.

Simple et complet

La carte PICcolino de la **figure 1** est disponible assemblée, prête à l'emploi, chez Elektor. Vous pouvez en outre télécharger gratuitement une suite logicielle avec environnement de développement, le langage de programmation JAL ainsi que ses bibliothèques, les instructions pour l'installation et un programme d'exemple. Vous disposerez ainsi gratuitement d'un environnement de développement complet pour PIC dans le puissant langage JAL (*Just Another Language*). Même avec peu, voire pas d'expérience des microcontrôleurs, chacun peut se lancer sans peine dans la programmation de la carte PICcolino, avec l'aide de la suite logicielle. Il suffit de suivre soigneusement les instructions d'installation reprises dans un PDF joint à la carte et d'essayer le programme de test qui l'accompagne. Vous aurez ainsi la preuve que tout est bien installé.

Exemple de projet : un voltmètre simple

Rien de tel qu'un petit projet pour montrer la simplicité du travail avec PICcolino : avec une entrée analogique de la carte, fabriquons un voltmètre pour des tensions entre 0 et 5 V. Pour éliminer l'effet d'éventuelles variations de tension d'alimentation, aidons-nous d'un petit circuit, une référence fixe basée sur un LM336.

La feuille de caractéristiques de la puce (**fig. 2a**) nous montre qu'elle se raccorde à la tension d'alimentation par une résistance R_S . La valeur de cette résistance, on la calcule en fonction du courant que le LM336 peut supporter : entre 400 μ A et 10 mA. Commençons avec 5 V et le courant maximum de 10 mA. Le LM336 règle lui-même la sortie pour atteindre 2,5 V, donc sous une alimentation de 5 V, la chute de tension sur la résistance vaudra $5 - 2,5 = 2,5$ V. Pour faire circuler un courant de 10 mA, la résistance série doit être $R = U / I = 2,5 / 10 \times 10^{-3} = 250 \Omega$. C'est la valeur minimum.

La tension minimale à laquelle le PIC peut encore fonctionner est de 2 V. Le courant doit alors être d'au moins 400 μ A et la résistance valoir $2 / 400 \times 10^{-6} = 5 \text{ k}\Omega$. C'est la valeur maximum. Mais dans ce cas, le LM336 ne donnerait plus 2,5 V et le voltmètre ne serait pas juste. Pas de souci malgré tout, nous allons voir qu'en pratique la tension minimale de service avoisine 3,4 V. Choisissons une valeur intermédiaire, p. ex. 1 k Ω , comme à la figure 2 de la feuille de caractéristiques.

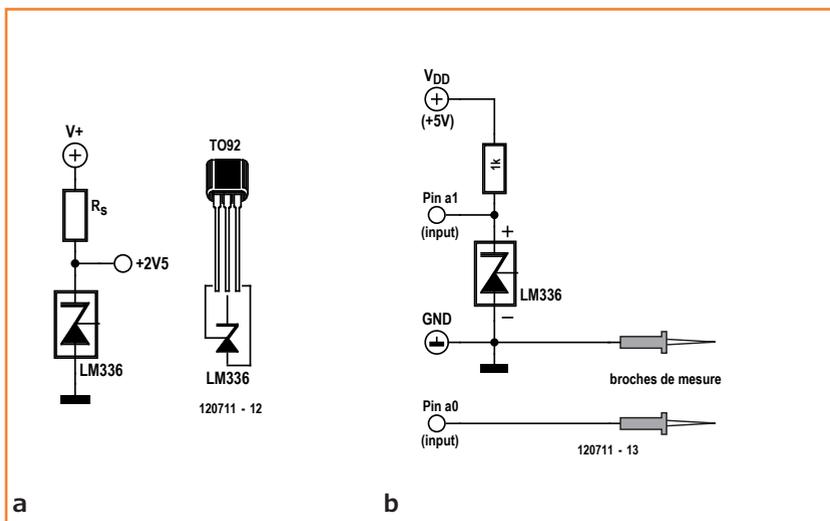
Outre le PICcolino, nous n'avons besoin que d'un LM336 et d'une résistance de 1 k Ω . Ces composants sont compris dans le kit complet PICcolino, disponible chez Elektor. La référence de 2,5 V se branche sur une entrée analogique de la carte. Nous avons choisi a1. Il faut encore deux broches de mesure, ou des fils, pour se relier à la tension à mesurer (entre 0 et max. 5 V). L'un des fils est raccordé à la masse, la broche GND de PICcolino, et l'autre à l'entrée analogique a0 (**fig. 2b**).

Le logiciel

Nous mesurons directement les unes après les autres les valeurs analogiques sur la broche a0, la référence, et a1, la tension à mesurer, avec l'instruction `adc_read_low_res`. Si vous n'êtes pas familiarisé avec le langage de programmation JAL utilisé, nous vous conseillons la lecture du livre *50 applications des microcontrôleurs PIC*, publié par Elektor (www.elektor.fr/PIC50). Le résultat des

Caractéristiques techniques

- programmeur inutile, PICcolino a son chargeur d'amorce
- alimentation stabilisée de 5 V sur la carte avec témoin à LED
- communication avec le PC par RS232 (ou USB par adaptateur)
- LED, interrupteur et potentiomètre pour démarrer d'emblée
- mise à zéro par poussoir ou automatique par cavalier
- espace pour modules d'extension enfichables
- microcontrôleur moderne PIC16F887 de Microchip :
 - 20 MHz (5 MIPS)
 - mémoire flash de 8 192 mots
 - 768 octets de RAM
 - 256 octets d'EEPROM
 - 35 broches d'E/S dont 25 sur embases
 - 14 entrées analogiques
 - 2 sorties MLI
 - 2 comparateurs
- communication par RS232 TTL, RS232, I²C, SPI, 1-Wire
- masse, 5 V et V_{in} disponibles sur embase.



deux mesures est un nombre compris entre 0 et 255, le 0 correspond à 0 V et 255 à la tension d'alimentation du microcontrôleur, en principe 5 V donc. La référence est 2,5 V et doit fournir une valeur de $255 / 2 = 127,5$. Si la tension qui alimente le contrôleur s'écarte de 5 V, la mesure sera différente. Appelons cette valeur x et la mesure sur

Figure 2.
Raccordement du LM336 (a)
et schéma du voltmètre (b).

L'auteur

Bert van Dam est l'auteur de livres, de matériel didactique et d'articles sur les microcontrôleurs PIC et ARM, sur l'Intelligence artificielle et sur les langages de programmation JAL, C, assembleur et Flowcode.

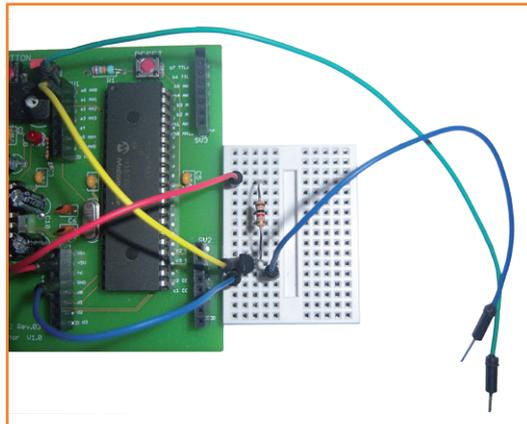


Figure 3. Assemblage sur une petite platine d'expérimentation.

Listage 1. Exemple de voltmètre

```
-- voltmètre simple (0 à 5 V) avec tension de référence
include 16f887_bert_piccolino

-- définition des entées
pin_a0_direction = input
pin_a1_direction = input

-- déclaration des variables
var byte ref, met, volt
const byte text[]=" volt "

forever loop
  -- tension de référence
  ref = adc_read_low_res(1)
  -- tension inconnue
  met = adc_read_low_res(0)
  -- déterminer la tension vraie
  volt = word(met)*25/ref
  --seulement par appui sur le bouton
  if pButton then
    -- aller à la bonne position
    serial_hw_write(27)
    serial_hw_write(«H»)
    -- valeur de mesure
    format_byte_dec(serial_hw_data,volt,6,1)
    -- texte
    serial_hw_printf(text)
    -- accorder au PC le temps de répondre
    delay_100ms(1)
  end if
end loop
```

l'autre broche y, nous pourrons écrire pour la tension réelle :

$$(y/x) * 2,5 V$$

Exemple : la tension est de 2,5 V. La mesure donne pour les deux broches 125. La bonne tension est alors de $125/125 * 2,5 V = 2,5 V$. Autre exemple : la tension est de 5 V. La mesure donne 255, la référence 127. La bonne tension est alors $255/127 * 2,5 V = 5 V$.

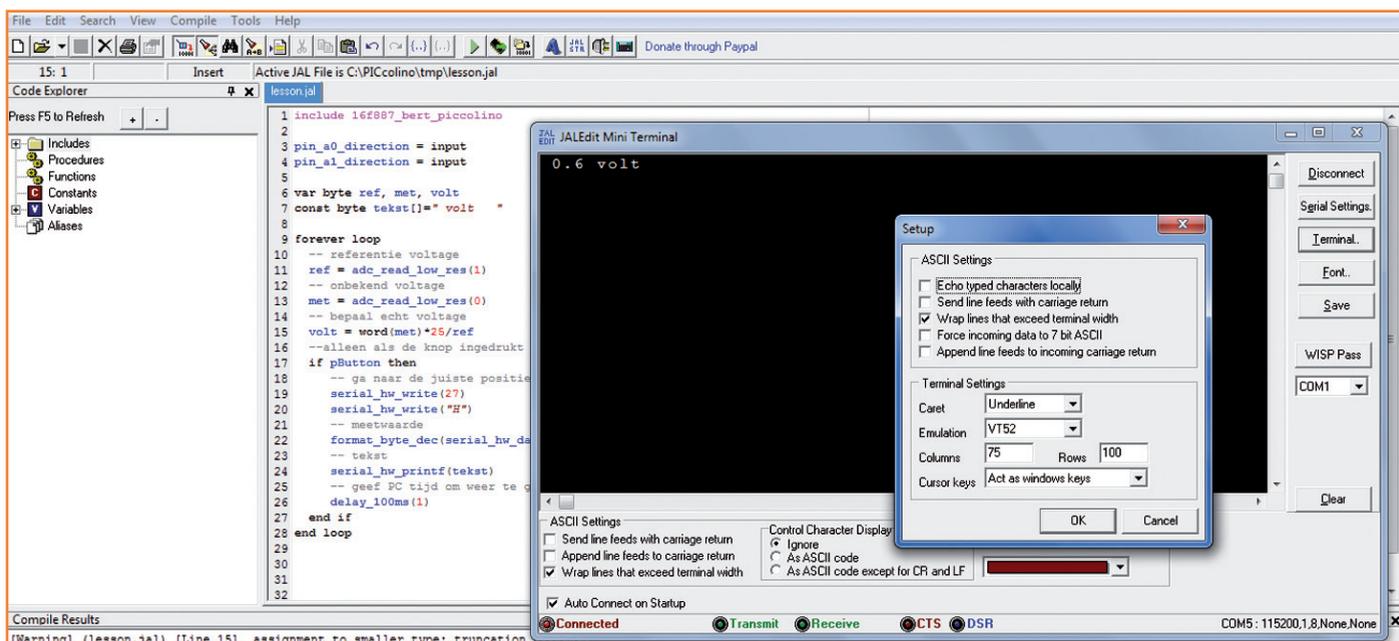
Comme JAL ne connaît pas les décimales, nous n'allons pas multiplier par 2,5, mais par 25. Les variables employées sont des octets, ils ne peuvent donc pas dépasser 255. Multiplier par 25 peut produire un résultat de $255 * 25 = 6\ 375$, ce qu'un octet ne peut pas contenir. Pour ne pas manquer de chiffres, faisons appel à l'instruction `word()` parce qu'un mot peut atteindre au maximum la valeur 65 535. Le résultat final peut s'exprimer par un octet, parce que le plus grand nombre est $255 * 25 / 127$, soit 50. Nous affichons le résultat avec un chiffre derrière la virgule avec l'instruction `format_byte_dec`, ce qui revient effectivement à le diviser par 10.

Le programme du voltmètre est transcrit dans le **listage 1**. Il envoie la mesure au PC quand on appuie sur le bouton-poussoir du PICcolino. Côté PC, le résultat est affiché dans le terminal de l'environnement de développement JALedit joint à la carte. Lancez le terminal d'un clic sur le symbole de la fiche, en haut à droite. Il se peut que vous receviez un message d'erreur si ce n'est pas le port COM par défaut qui est en service sur votre PC. Cliquez alors sur *Continue Application*. On peut éviter ce message en ne cochant pas *Autoconnect on Startup* dans l'écran du terminal. Cliquez sur *Serial Settings*, choisissez le bon port COM et introduisez les réglages suivants :

COM	le port sur lequel PICcolino est branché
Baudrate	115200
Databits	8
Stopbits	1
Parity	none
Flow control	none

Cliquez alors sur Ok et choisissez Terminal. Sélectionnez-y VT52 comme émulation (**fig. 4**) et cliquez sur Ok, puis sur Connect.

Connectez les sondes de mesure sur une tension appropriée entre 0 et 5 V. La masse doit se raccorder à celle de PICcolino, repérez donc le + et



le -, puis poussez sur le bouton de PICcolino. La valeur de la tension s'affiche à l'écran (fig. 5).

Tension trop basse

Docile aux instructions, j'ai branché mon PICcolino sur une alimentation de 7 V ou un peu plus, le microcontrôleur travaille donc sous 5 V. Il est parfaitement capable de mesures correctes de tension sans LM336. Mais avec le LM336, le circuit fonctionnera-t-il encore même si l'alimentation ne lui donne plus 5 V ? Essayons, pour voir ! Le PICcolino est maintenant branché sur une alimentation réglable qui lui délivre au moins 7 V, mais pas plus de 12 V. Entre les pointes de test, une pile de 1,5 V en respectant le + et le -. Je branche un voltmètre entre masse et broche +5 V du PICcolino. C'est la tension de fonctionnement du microcontrôleur. Je lance le programme : l'écran affiche une valeur mesurée de 1,5 V, pour autant que la pile soit bonne. Nous allons maintenant diminuer lentement la tension de l'alimentation de laboratoire. Mais attention : sur la broche +5 V, elle ne peut pas descendre sous celle des pointes de mesure, 1,5 V dans ce cas-ci, sous peine d'endommager le microcontrôleur. Surveillons donc le voltmètre ! En descente jusqu'à 7 V, rien à signaler. Plus loin, le régulateur de la tension de 5 V ne sera plus capable de fournir une tension stable, elle va décroître aussi. Quand il ne reste plus que 3,4 V environ sur la broche +5 V, la liaison RS232/

USB s'évanouit. Mais PICcolino continue vaillamment son travail, il ne s'arrêtera d'ailleurs qu'aux alentours de 2 V. Et à l'écran, la tension de mesure est restée fixe à 1,5 V tout le temps, jusqu'aux 3,4 V fatidiques. Le circuit fonctionne donc impeccablement.

Évidemment, ceci n'est qu'un petit projet pour faire connaissance avec PICcolino, mais avec cet environnement de développement, vous pourrez réaliser très vite et tester des circuits plus vastes et plus complexes.

(120711 - version française : Robert Grignard)

Figure 4. L'environnement de développement gratuit JALedit avec l'installation de l'émulation de terminal.

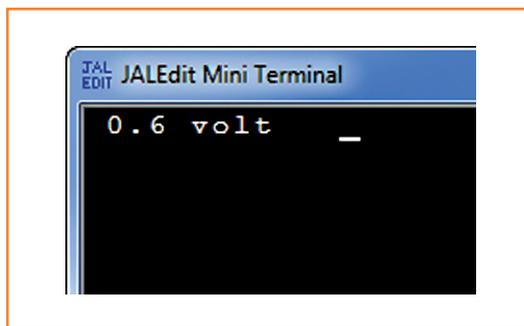


Figure 5. Il ne reste qu'à lire la tension sur l'écran du PC.

Tout le matériel nécessaire pour ce projet est disponible auprès d'Elektor. La carte PICcolino complète est aussi disponible accompagnée d'un lot de composants et d'une platine d'expérimentation.

carte d'extension Linux

Afficheur, boutons, horloge RTC, *buzzer* et 16 GPIO

Benedikt Sauter [1]

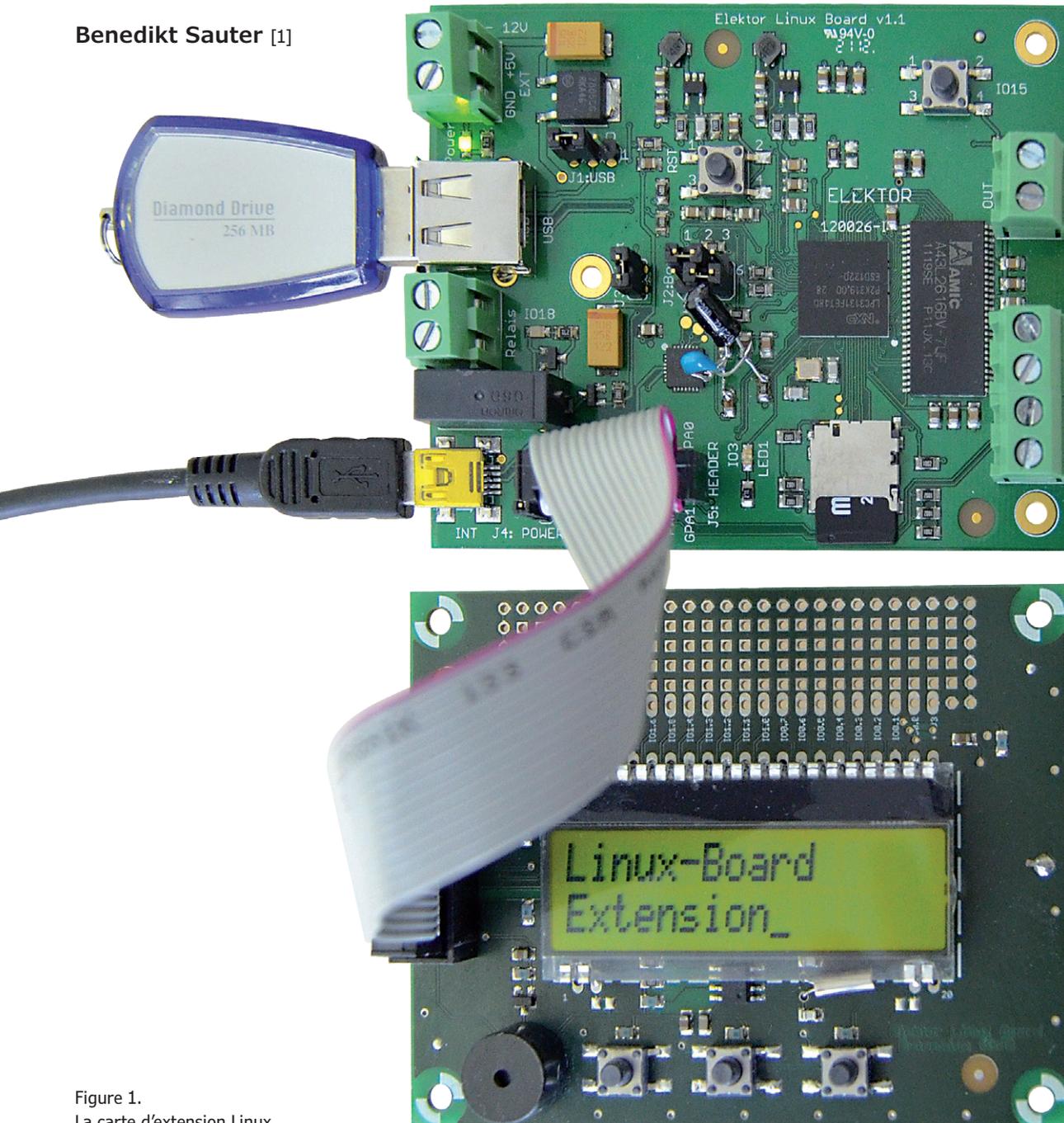


Figure 1.
La carte d'extension Linux.

Dans la série *embarquez Linux ! (5)*, nous avons vu comment dialoguer avec la carte Linux-Elektor à l'aide d'interfaces utilisateur HTML. En pratique, il n'est toutefois pas rare d'avoir à intervenir dans un processus de commande, sans pour autant avoir sous la main un PC et un navigateur. L'idéal dans ce cas est de disposer de quelques boutons et d'un afficheur pour saisir rapidement un paramètre ou connaître l'état du processus. L'idéal a une forme : celle de la carte d'extension présentée ici.

Tableau 1. Signaux sur le connecteur à 14 contacts						
broche	description	affectation		affectation	description	broche
1	GPA0	bouton gauche		bouton milieu	GPA1	2
3	GPA3	bouton droit		buzzer	PWM	4
5	SCL (I2C)	RTC et duplicateur		RTC et duplicateur	SDA (I2C)	6
7	MOSI (SPI)	afficheur		afficheur	MISO (SPI)	8
9	horloge système	-		afficheur	SCK (SPI)	10
11	GPIO14	afficheur		afficheur	GPIO11	12
13	3,3V	alimentation		alimentation	GND	14

Cette carte d'extension Linux offre tout ce qu'il faut pour mettre en œuvre une grande variété de projets. Trois boutons-poussoirs pour la saisie (**fig. 1**), quant aux sorties elles peuvent être envoyées sur un afficheur à 2 lignes et 16 caractères. Judicieusement placés sous l'écran, les boutons pourront commander un petit menu. La carte comprend en outre un buzzer pour les sorties acoustiques, une horloge temps réel (RTC) alimentée par pile, ainsi qu'un duplicateur de ports offrant 16 entrées/sorties numériques supplémentaires. Un espace pastillé de 1,5 x 6 cm a également été prévu pour les extensions maison.

Carte Linux-Elektor et carte d'extension sont toutes deux pourvues d'un connecteur à 2x7 contacts [2] (**fig. 2**). Il suffira donc d'un câble en nappe à 14 fils pour les relier entre elles. Ce câble ne sert pas qu'aux transports des signaux, il permet aussi à la carte mère d'alimenter en 3,3 V la carte d'extension.

Le **tableau 1** montre l'affectation des broches. La carte d'extension pourra bien sûr également être utilisée pour d'autres projets (personnels) ; nous vous renvoyons d'ailleurs à l'article du prochain numéro consacré à *l'Embedded Firmware Library*. La carte d'extension est disponible auprès

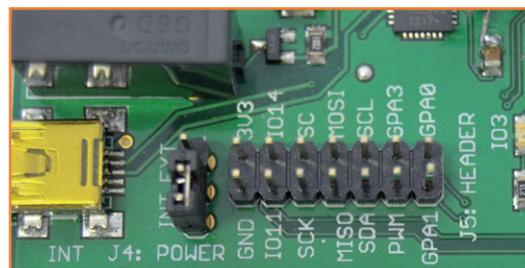


Figure 2. Le connecteur à 14 contacts sert à relier la carte d'extension Linux et la carte Linux-Elektor.

d'Elektor. Elle est proposée équipée en CMS et testée, vous n'aurez à souder que le LCD et quelques composants [3].

Schéma

Le schéma de la carte d'extension est reproduit sur la **figure 3**. La liaison à la carte Linux-Elektor se fait via K1. Les composants ont bien sûr été choisis compatibles avec du 3,3 V puisque l'alimentation l'imposait. Le LCD ne pouvait être piloté que par I²C ou SPI (l'adressage de la plupart des afficheurs alphanumériques s'effectue en effet en mode 4 bits ou 8 bits). Nous avons opté pour le DOGM162L-A [4] (*Electronic Assembly*), un afficheur équipé d'une interface SPI.

Nous avons connecté les boutons-poussoirs aux trois canaux A/N (GPA0, GPA1 et GPA3)

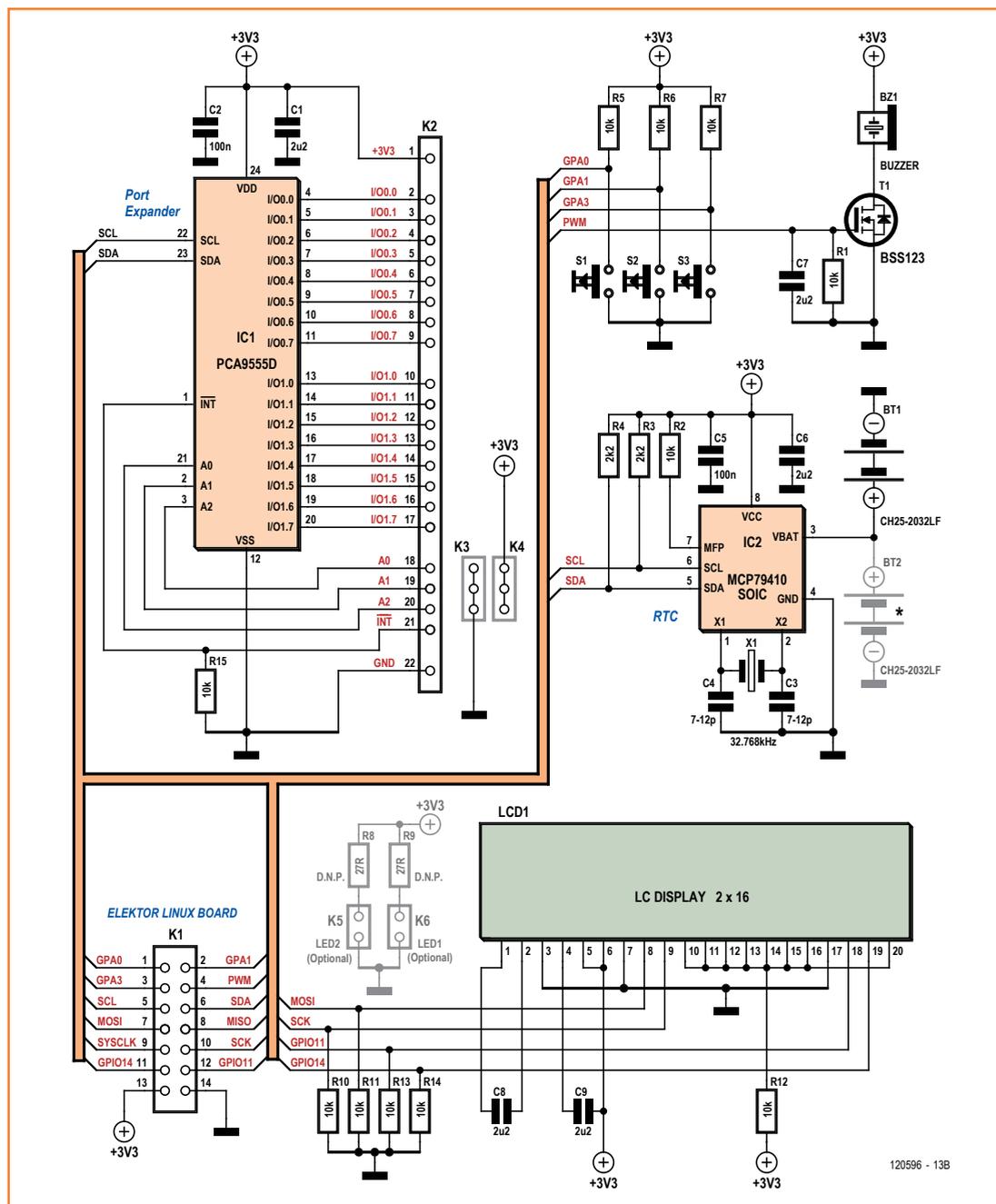


Figure 3. Schéma de la carte d'extension Linux.

du connecteur d'extension puisque ces canaux ne sont pas utilisés par la nouvelle carte. Vous pouvez aussi raccorder un commutateur à plusieurs positions et vous servir de diviseurs de tension pour connaître une position. Ici ne sont présents que des états « haut » (le convertisseur A/N mesure la tension maximale) et « bas » (le convertisseur A/N mesure 0). L'état « bas » indique par ailleurs que l'utilisateur appuie sur le bouton.

Le buzzer est relié au FET T1, lui-même relié à la sortie MLI (PWM) du LPC3131, ce qui permet de produire un son de fréquence constante. D'après sa fiche technique [5], le buzzer est prévu pour fonctionner sous 8 à 15 V ; comme il n'est ici alimenté qu'en 3,3 V, le volume sonore ne sera pas maximal. La carte comprend en outre une horloge temps réel, le circuit MCP9401 [6]. Cette horloge est tamponnée par une pile bouton afin que la carte

conserve l'heure exacte même après avoir été mise hors tension.

Comme duplicateur de ports d'E/S (IC1), nous avons choisi le PCA9595 de chez NXP [7]. Ce composant est accessible par I²C et fournit 16 E/S numériques supplémentaires. Les broches GPIO recevront par exemple des LED d'état, et/ou d'autres poussoirs (ou même un clavier matriciel).

Afficheur

Pour faciliter le pilotage de l'afficheur depuis vos applications, nous avons écrit en C un petit programme qui se lance depuis la ligne de commande. Vous pouvez télécharger et copier sur la carte SD le code source ou déjà compilé. Le plus simple est d'utiliser une clé USB. Téléchargez d'abord sur votre ordinateur le fichier [8] (pas d'espace dans l'adresse), copiez-le sur une clé USB, puis insérez la clé dans la carte Linux-Elektor (**fig. 4**). La console affiche normalement un message après insertion de la clé, qui doit ensuite être montée sur le système de fichiers :

```
mount /dev/sda1 /mnt/
```

Pour que le programme puisse être lancé par simple invocation de son nom (que la clé soit présente ou non), copiez-le dans le répertoire `/usr/local/bin` de la carte SD :

```
cp /mnt/dog-app /usr/local/bin
```

Vous pouvez vous assurer qu'il fonctionne en lançant depuis la console :

```
dog-app -h
```

L'argument `-h` demande l'affichage de l'aide.

Pour envoyer un texte sur l'afficheur, les deux cartes doivent bien entendu être reliées (veillez à ce que la marque rouge sur le câble en nappe soit située du côté de GPA1/GPA0).

L'envoi d'une chaîne de caractères sur l'écran se fait avec la syntaxe suivante :

```
dog-app -n -w 'Salut Elektor'
```

Tout ce qui est placé entre apostrophes (droites) n'est pas interprété par le shell.

Et pour écrire sur la seconde ligne de l'écran ? Il faut d'abord comprendre que la ligne d'un afficheur est constituée de plusieurs segments. Écrire à un certain endroit revient donc à positionner le curseur sur le segment correspondant. Au premier segment de la

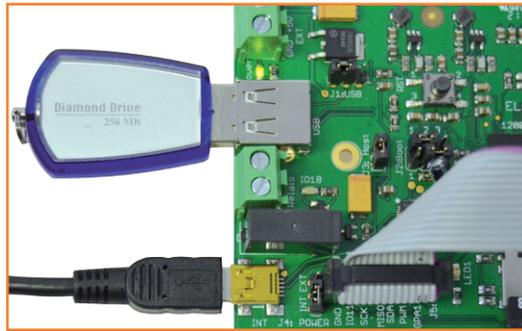


Figure 4.
Une clé USB insérée dans la carte Linux-Elektor.

première ligne correspond ici un décalage (*offset*) de 128, et on accède au premier segment de la seconde ligne avec un décalage de 192. La commande suivante écrit sur la seconde ligne :

```
dog-app -o "192" -w 'Salut c\'est encore moi'
```

Boutons-poussoirs

Les boutons sont reliés aux entrées du convertisseur A/N du LPC3131. L'accès aux canaux du convertisseur A/N à l'aide de son pilote a déjà été expliqué dans les précédents articles de la série *embarquez Linux !*

Pour connaître l'état du poussoir de gauche, il faut d'abord rediriger sa sortie dans le fichier `/dev/lpc313x_adc` :

```
echo 0 > /dev/lpc313x_adc
```

On affiche ensuite ce fichier avec la commande `cat` :

```
root@gnublin:~# cat /dev/lpc313x_adc
```

Si le poussoir est ouvert, la valeur lue sera très grande, proche de 1023 (0x3ff en hexadécimal) :

```
0x3ff
```

Si le poussoir est fermé, la valeur renvoyée par le CAN est 0 :

```
root@gnublin:~# cat /dev/lpc313x_adc
```

```
0x000
```

La commande `echo` pour connaître l'état du poussoir du milieu est :

```
echo 1 > /dev/lpc313x_adc
```

Figure 5.
Script pour le réglage de l'heure et de la date au démarrage.

```
case "$1" in
start)
# start here
if [ ! -d /sys/bus/i2c/devices/1-006f/ ]; then
echo mcp7940 0x6f > /sys/bus/i2c/devices/i2c-1/new_device
fi
if [ ! -c /dev/rtc0 ]; then
mknod /dev/rtc0 c 251 0 2>/dev/null &&
fi
# end
if [ "$VERBOSE" != no ]
then
echo "System time was `date`."
echo "Setting the System Clock using the Hardware Clock as reference..."
fi
```

Figure 6.
Chargement du module pca953x pour accéder aux ports GPIO du duplicateur.

```
Elektor GNUBLIN ELDK (Built by Poky 5.0) 5.0 gnuBLIN ttyS0
http://www.gnuBLIN.org

gnuBLIN login: root
root@gnuBLIN:~# modprobe pca953x
root@gnuBLIN:~# echo 98 > /sys/class/gpio/export
root@gnuBLIN:~# echo out > /sys/class/gpio/gpio98/direction
root@gnuBLIN:~# echo 1 > /sys/class/gpio/gpio98/value
root@gnuBLIN:~#
```

Tableau 2. Accès aux broches du PCA9555 via le fichier export

descripteur de broche	valeur pour /sys/class/gpio/export
IO0_0	98
IO0_1	99
IO0_2	100
IO0_3	101
IO0_4	102
IO0_5	103
IO0_6	104
IO0_7	105
IO1_0	106
IO1_1	107
IO1_2	108
IO1_3	109
IO1_4	110
IO1_5	111
IO1_6	112
IO1_7	113

Et pour le poussoir de gauche :

```
echo 3 > /dev/lpc313x_adc
```

Buzzer

Produire un son via le système de fichiers est très simple :

```
echo ff > /dev/lpc313x_pwm
```

Pour l'interrompre :

```
echo 00 > /dev/lpc313x_pwm
```

Vous pouvez produire une suite de bips en passant à la commande usleep la durée de la pause souhaitée entre deux sons, c'est-à-dire la durée de la suspension du programme.

Horloge temps réel

L'horloge a besoin d'un module particulier, déjà présent dans le système de fichiers. Pour le charger dans le noyau, entrez :

```
modprobe rtc-mcp7940
```

Pour que ce pilote soit chargé automatiquement au démarrage du système, ajoutez la ligne suivante au fichier /etc/modules :

```
rtc-mcp7940
```

Nous devons maintenant fournir au noyau

l'adresse I2C de l'horloge. La commande suivante s'en charge :

```
echo mcp7940 0x6f > /sys/bus/i2c/devices/i2c-1/new_device
```

Avant de pouvoir créer le fichier de périphérique de l'horloge, nous devons connaître le numéro majeur que lui a attribué le noyau :

```
cat /proc/devices
```

Le numéro majeur 251 est l'affectation standard. Cette information permet de créer le fichier de périphérique :

```
mknod /dev/rtc0 c 251 0
```

Réglons maintenant la date et l'heure avec la commande date. Le format est [MMDDhhmm[[CC]YY]]. Par exemple :

```
date 122014342012
```

renvoie sur la console :

```
Thu Dec 20 14:34:00 UTC 2012
```

Pour transférer date et heure dans le composant RTC, lancez :

```
hwclock -w
```

Date et heure de l'horloge temps réel se récupèrent avec :

```
hwclock -r
```

Avec les données précédentes, la sortie est :

Thu Dec 20 14:34:17 2012 0.000000 seconds

Avec une pile, il sera inutile de régler l'heure à chaque reconnexion de la carte d'extension. Le porte-pile peut du reste être monté sur l'une ou l'autre des faces de la carte (voir schéma).

Pour que l'heure du système puisse être réglée automatiquement au démarrage, le fichier `/etc/init.d/hwclock.sh` doit contenir les lignes reproduites sur la **figure 5**, et il est impératif que le module `mrtc-mcp7940` ait été enregistré dans le fichier `/etc/modules`.

Duplicateur de ports

Dans l'article *embarquez Linux ! (7)* [9], nous avons utilisé les outils `i2c-tools` et un programme C pour dialoguer avec le duplicateur PCA9555. Le module de noyau `pca953x` fournit quant à lui un accès très simple aux ports GPIO du duplicateur et du LPC3131. Des exemples de commandes sont reproduits sur la **figure 6**. Le **tableau 2** indique la correspondance entre ports GPIO et numéros de broches pour leur accès via le fichier `export`.

Nous présenterons une petite application de notre carte d'extension dans le prochain numéro. Nous verrons en particulier comment créer un menu pour le LCD et les boutons, menu que vous pourrez bien sûr adapter à vos besoins.

(120518 – version française : Hervé Moreau)

Liens

- [1] sauter@embedded-projects.net
- [2] www.elektor-magazine.fr/120181
- [3] www.elektor-magazine.fr/120596
- [4] www.lcd-module.de/pdf/doma/dog-m.pdf [en allemand]
- [5] <http://goo.gl/ToKZZ>
- [6] <http://ww1.microchip.com/downloads/en/DeviceDoc/22266D.pdf>
- [7] www.ti.com/lit/ds/symlink/pca9555.pdf
- [8] http://gublin.googlecode.com/git/lpc3131/gublin_package/src/gublin-dogm/usr/bin/gublin-dogm
- [9] www.elektor-magazine.fr/120518
- [10] www.elektor-magazine.fr/120146

Liste des composants

Résistances :

R1, R2, R5, R6, R7, R10, R11, R12, R13, R14, R15
= 10 kΩ
R3, R4 = 2,2 kΩ
R8, R9 = 27 Ω (optionnelles)

Condensateurs :

C1, C6 à C9 = 2,2 μF
C2, C5 = 100 nF
C3, C4 = 7 à 12 pF

Semi-conducteurs :

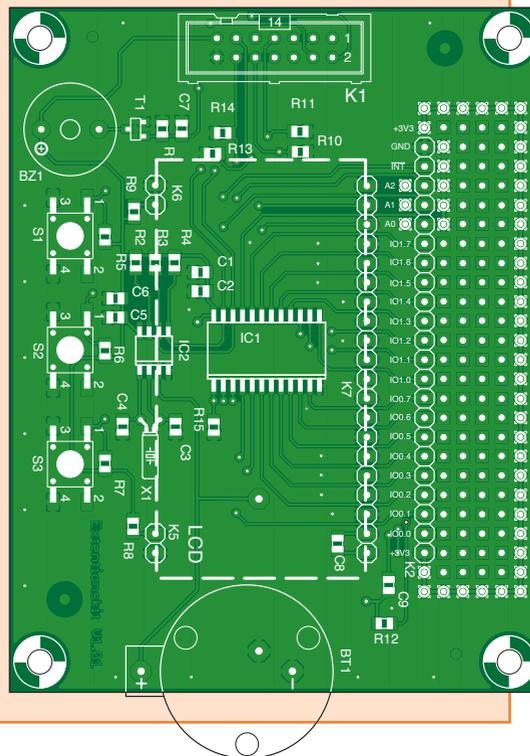
T1 = BSS123
IC1 = PCA9555D, SO-24
IC2 = MCP7940, SOIC-8

Divers :

X1 = quartz horloger 32768 kHz
K1 = connecteur femelle à 14 contacts
K2 = embase SIL à 3 broches ou à 22 broches (option.)
K3, K4 = embase SIL à 3 broches
K5, K6 = embase SIL à 2 broches (optionnelle)
BT1 ou BT2 = porte-pile CH25-2032LF
BZ1 = buzzer
S1, S2, S3 = boutons-poussoirs CMS 9314
LCD1 = LCD 2x16, DOGM162L-A de chez Electronic Assembly
circuit imprimé

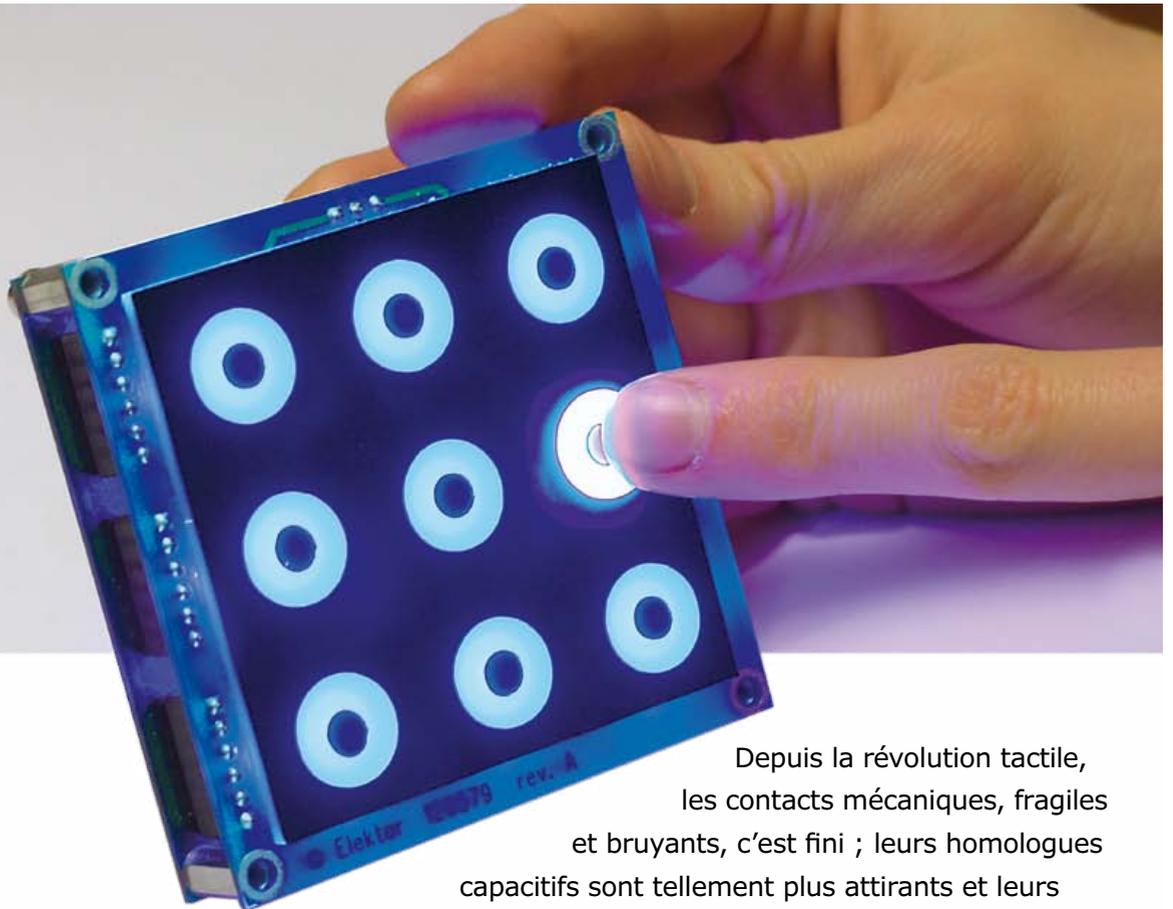
Module/kit 120596-91:

carte montée avec CMS et testée + LCD1,
X1, K1 à K4, BZ1, BT1 à monter soi-même



Pattern Lock

clavier tactile à reconnaissance de motif



David Ardouin [5]
Nantes

Depuis la révolution tactile, les contacts mécaniques, fragiles et bruyants, c'est fini ; leurs homologues capacitifs sont tellement plus attirants et leurs avantages si nombreux. Pourtant, à en juger par la modeste présence de cette technique dans les montages que publie Elektor, il reste beaucoup de monde à convaincre de son intérêt. C'est pourquoi je vous en propose non seulement une approche ludique, afin de démystifier le sujet et bien le comprendre, mais aussi, pour vous encourager à utiliser à votre tour des interfaces tactiles, une application sérieuse et bien étudiée, à partir de la méthode de détection *QTouch* d'Atmel.

C'est après avoir développé une interface tactile en milieu professionnel que j'ai eu envie de partager mon expérience sous la forme d'une carte de démonstration de la technologie capacitive. Avec la mode des téléphones intelligents sous le système d'exploitation *Android* et leur fameux *motif géométrique de déverrouillage (pattern lock)*, il m'a paru intéressant de moderniser le système bien connu du *digicode* : au lieu de taper un code

sur un clavier de 9 touches, on dessine un motif géométrique du bout du doigt. Mon *Pattern Lock* est une élégante serrure tactile rétro-éclairée, capable de mémoriser séparément deux motifs composés chacun de 4 à 9 touches. La reconnaissance de l'un de ces codes permet d'activer le circuit de votre choix au moyen d'un relais. Avant d'entrer dans le vif du sujet, attardons-nous sur le principe des interfaces tactiles. Nos

lecteurs vétérans n'ont pas oublié qu'Elektor a publié une foule de touches capacitives en tout genre, à une époque où il n'était question que de techniques analogiques. L'aventure était passionnante, mais le résultat souvent décevant. Grâce à l'utilisation de microcontrôleurs, l'efficacité des techniques d'aujourd'hui, comme celle qui est décrite ici, est sans commune mesure avec le bricolage d'antan.

Limites et avantages

Question directe : « à quoi bon une touche capacitive ? » Selon leur fonction, les interfaces tactiles se distinguent notamment selon leur *degré de liberté*, terme cher aux mécaniciens. Les capteurs les plus simples sont les poussoirs et autres interrupteurs pour lesquels votre liberté est réduite à presque rien ; ces commandes, dites de *dimension 0*, ne connaissent que deux états : actif ou inactif, fermé ou ouvert, allumé ou éteint, 0 ou 1. Ce sont des *touches capacitives* (**fig. 1a**), elles opèrent une commande simple en tout ou rien. L'interface tactile offre la possibilité d'une commande capacitive plus fine, p. ex. une commande linéaire, proportionnelle au mouvement du doigt qui glisse sur une surface, de haut en bas. On parle de *glissière*, mais cette course unidimensionnelle, qui ne permet donc de faire varier qu'une variable, n'est pas nécessairement rectiligne ; elle pourra être circulaire, comme sur un très célèbre lecteur de mp3, et l'on parle alors de *rotor* (**fig. 1b**).

Si la zone tactile est une surface, où le doigt est libre de se déplacer dans toutes les directions, comme sur un écran de téléphone, il devient possible de moduler en même temps *deux* variables de sortie (**fig. 1c**). Cette technique, bien qu'assez proche des deux autres, sort du cadre de cet article.

Le grand avantage de touches ou de glissières capacitives réside dans la liberté totale de disposition des organes de commande, des inscrip-

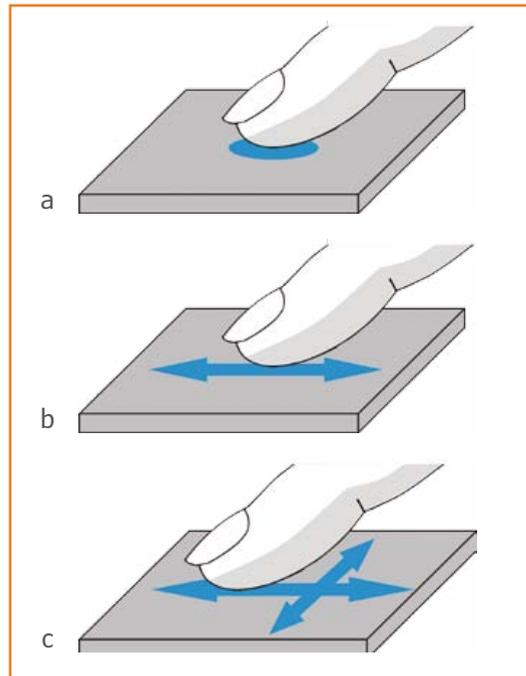


Figure 1.
Touche simple, de dimension 0 (a) ou surface tactile à une dimension ou glissière (b) ou surface tactile à deux dimensions (c).

tions et des pictogrammes utilisés. Sans oublier l'aspect *high-tech*, souvent considéré comme plus attractif que les vieux boutons ronds et carrés. Une touche tactile présente aussi l'avantage d'une parfaite étanchéité ; elle est inusable et facile à éclairer de l'intérieur. Si vous envisagez d'équiper de touches tactiles un montage déjà commandé par un μC Atmel, et qu'il reste quelques broches et un peu de ressources disponibles, l'incorporation de touches capacitives ne coûtera que quelques composants passifs et un peu de place sur le circuit imprimé.

Rappels de physique

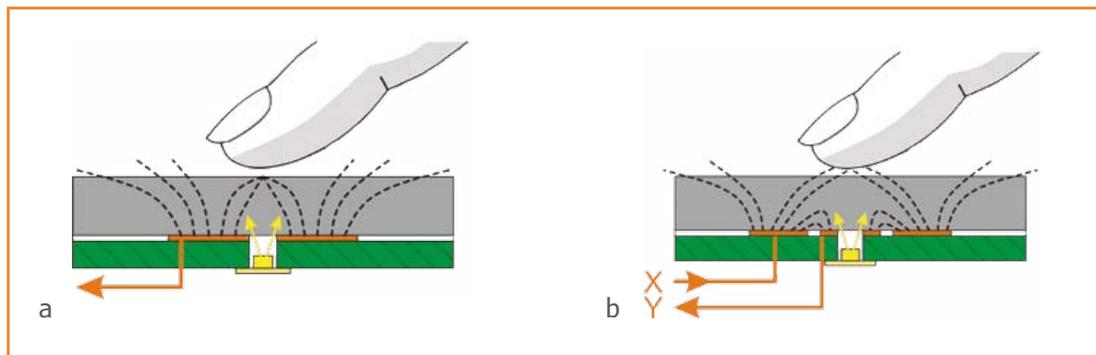
Un condensateur est constitué d'une paire d'électrodes conductrices séparées par une couche d'isolant dite diélectrique. Selon la grandeur de la capacité à obtenir, on joue sur la dimension des électrodes, leur écartement et le matériau

L'auteur

Quand j'étais petit, je rêvais d'être « bricoleur comme papa ». Je n'ai pas abandonné ce rêve d'enfance, grâce aux cours de technologie au collège et aux nombreux montages réalisés à partir de magazines. Puis je me suis orienté vers l'électronique : un Bac S puis un cursus de 5 ans en école d'ingénieurs à l'ESEO d'Angers m'ont permis d'en faire mon métier sans entamer ma passion. Entre mes rôles de mari et de papa, je trouve du temps libre pour mettre au point quelques-unes des idées qui fourmillent dans ma tête et pour pratiquer un peu d'aéromodélisme.



Figure 2.
Électrode simple (a) et électrodes couplées (b).



du diélectrique. Pour assurer une bonne sensibilité, on cherchera la plus grande différence de capacité possible entre l'absence et la présence du doigt. Cette variation se joue sur quelques dixièmes de picofarads seulement. Pour la fiabilité de vos capteurs, il faudra donc respecter quelques règles.

Les jumelles

Pour la détection capacitive, deux types d'électrodes sont en usage, avec chacune des avantages et des inconvénients. La première est la plus simple (**fig. 2a**) ; la touche est constituée d'une pastille conductrice de l'ordre de 10 mm de diamètre, d'une résistance et d'un condensateur, sans oublier, pour former la surface sensible, le matériau isolant. Une telle brique fonctionnelle peut être dupliquée pour former autant de boutons que vous le souhaitez, tant qu'il y a, sur le μC qui en assure la commande, le compte de broches requis, soit $2n + 1$ broches, où n est le nombre de touches. Cette électrode forme l'une des armatures d'un condensateur : elle est à la fois émettrice et réceptrice de charges électriques. Le doigt forme la seconde électrode : quand il s'approche, la capacité du condensateur augmente considérablement, ce qui permet la détection. Contrairement aux idées reçues, nul besoin d'être à la terre ou à la masse du système pour que la détection ait lieu. Atmel appelle cette première méthode de détection *QTouch* ou *Self-Capacitance Keys*.

Quand le nombre de touches à réaliser dépasse la dizaine, on préférera une seconde technique, dite à *capacité mutuelle*, ou *QMatrix*, avec des électrodes couplées (**fig. 2b**). Chaque touche comporte une électrode pilote (X) et une électrode réceptrice (Y). En l'absence de détection, une partie des charges électriques émises par l'électrode X est reçue par l'électrode Y, mais dès qu'un doigt s'approche, il perturbe cet équilibre en capturant une partie de la charge, diminuant ainsi la capacité mesurée. Une disposition en matrice de ces électrodes permet de réaliser jusqu'à 64 touches avec seulement 8 X et 8 Y. En associant trois touches convenablement disposées, l'une et l'autre de ces méthodes permettent de réaliser aisément des glissières ou

Figure 3.
Organisation des électrodes couplées du circuit tactile.

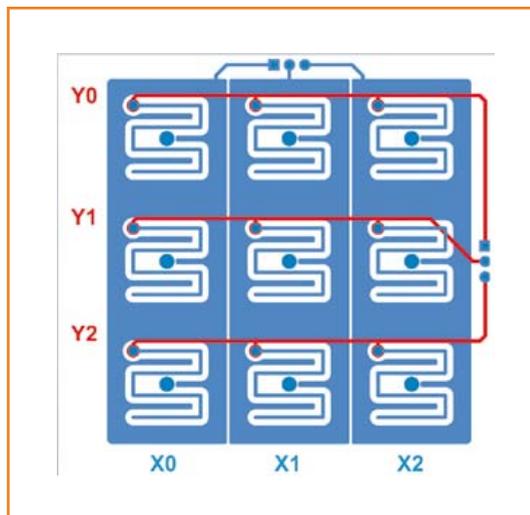
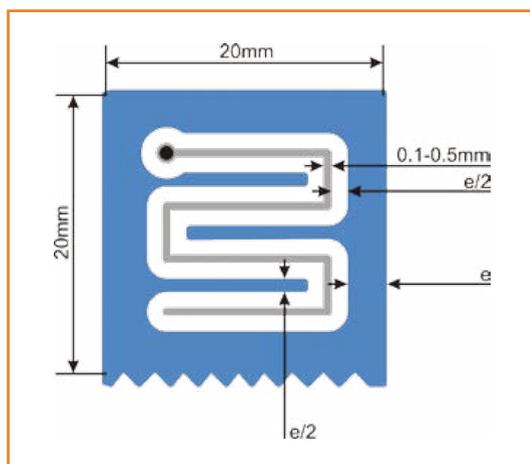


Figure 4.
Détails d'une électrode.



des rotors. Cependant, le dessin de telles électrodes entrelacées demande une attention particulière. Ces particularités sont très bien détaillées dans le document [1] où vous trouverez toutes les règles à respecter. Dès la conception, il est crucial d'obtenir la capacité la plus faible possible en l'absence de détection ; pour cela, on veillera à éloigner des touches tout ce qui peut l'être, particulièrement le plan de masse. Pour les électrodes en matrice, un soin particulier sera apporté au routage des lignes Y. Les lignes X sont moins sensibles aux perturbations.

Bases du fonctionnement

Ma surface tactile comporte neuf touches capacitatives, disposées en matrice XY. Le carré de 3x3 touches est idéal (**fig. 3**). La scission en deux circuits imprimés a une double fonction : réduire l'encombrement tout en isolant le plus possible les pistes sensibles des couplages parasites avec les autres composants. Le dessin des électrodes a été optimisé pour une épaisseur de matériau isolant de 3 mm ; la forme en lacets apporte un couplage maximal entre électrodes au niveau des touches mais rien ne vous empêche de les adapter à vos besoins (**fig. 4**).

La cote 'e' représente l'épaisseur du matériau isolant placé sur les touches. Les LED D1 à D9, au format 1206, sont montées à l'envers, au dos de ce premier circuit imprimé, pour permettre le rétro éclairage de chaque zone à travers un trou. Ceci permet d'illuminer chaque zone individuellement. Les condensateurs C1 à C9, montés en parallèle, sont indispensables. Sans eux, la sensibilité serait dégradée. Cinq connecteurs J1 à J5 permettent enfin la connexion entre ces deux circuits en transportant les signaux X, les Y ainsi que les deux bornes de chaque LED.

La simplicité de l'électronique (**fig. 5a**) nécessaire au fonctionnement des touches est rendue possible grâce à l'intégration de la bibliothèque *QTouch* dans le microcontrôleur ATtiny88 de la série AVR d'Atmel. Durant la phase d'acquisition d'une touche, l'électrode pilote (X) est stimulée par un train d'impulsions carrées à haute fréquence. Les charges émises par ce plan de cuivre sont captées en partie par l'électrode Y proche et sont accumulées dans C19 à C21 à travers R14 à R16. À la fin du train d'impulsions, un chronomètre est déclenché dans le microcontrôleur pour mesurer le temps nécessaire à la décharge des condensateurs à travers R11 à R13. Lorsque, pendant un cycle d'acquisition, vous approchez votre doigt, celui-ci absorbe une partie des charges normalement stockées dans le condensateur. À la fin du train d'impulsions, la tension aux bornes de ce condensateur est donc plus faible et la durée de sa décharge également réduite, ce qui ne manque pas d'être mesuré par le chronomètre.

Ce cycle de mesure, classique et éprouvé, est ensuite répété par le μC pour chaque combinaison X-Y.

En l'absence de contrainte de précision, ce composant est cadencé par son horloge interne à 8 MHz.

L'affectation des broches du μC aux électrodes de la matrice est régie par plusieurs contraintes que nous ne détaillerons pas ici. Pour vous guider pour cette étape, *Atmel* fournit heureusement le logiciel *AVR QTouch Studio*. Les électrodes X sont pilotées à travers R8 à R10 qui réduisent les émissions électromagnétiques. Leur valeur n'est pas critique.

Les sorties PD5 et PB0 du μC pilotent les relais reed RL1 et RL2 à travers deux MOSFET écono-

En bonus : Pattern Piano...

Mes filles de dix-huit mois et trois ans adorent jouer avec le prototype du *Pattern Lock*, mais celui-ci ne se montre pas toujours très conciliant et manifeste son mécontentement en allumant sa LED centrale. Devant la frustration de mes apprenties électroniciennes, j'ai transformé le verrou électronique récalcitrant en remplaçant simplement le relais RL1 par un petit haut-parleur 8 Ω en série avec une résistance de 22 Ω (**fig. 9**). Quelques lignes de code plus loin, le *Pattern Piano* était né, pour leur plus grand bonheur et celui de vos oreilles ! Les fréquences fondamentales produites correspondent aux notes comprises entre Do (525 Hz) et Ré de l'octave supérieure (1180 Hz) et permettent même de jouer de petites mélodies. Essayez, c'est amusant !



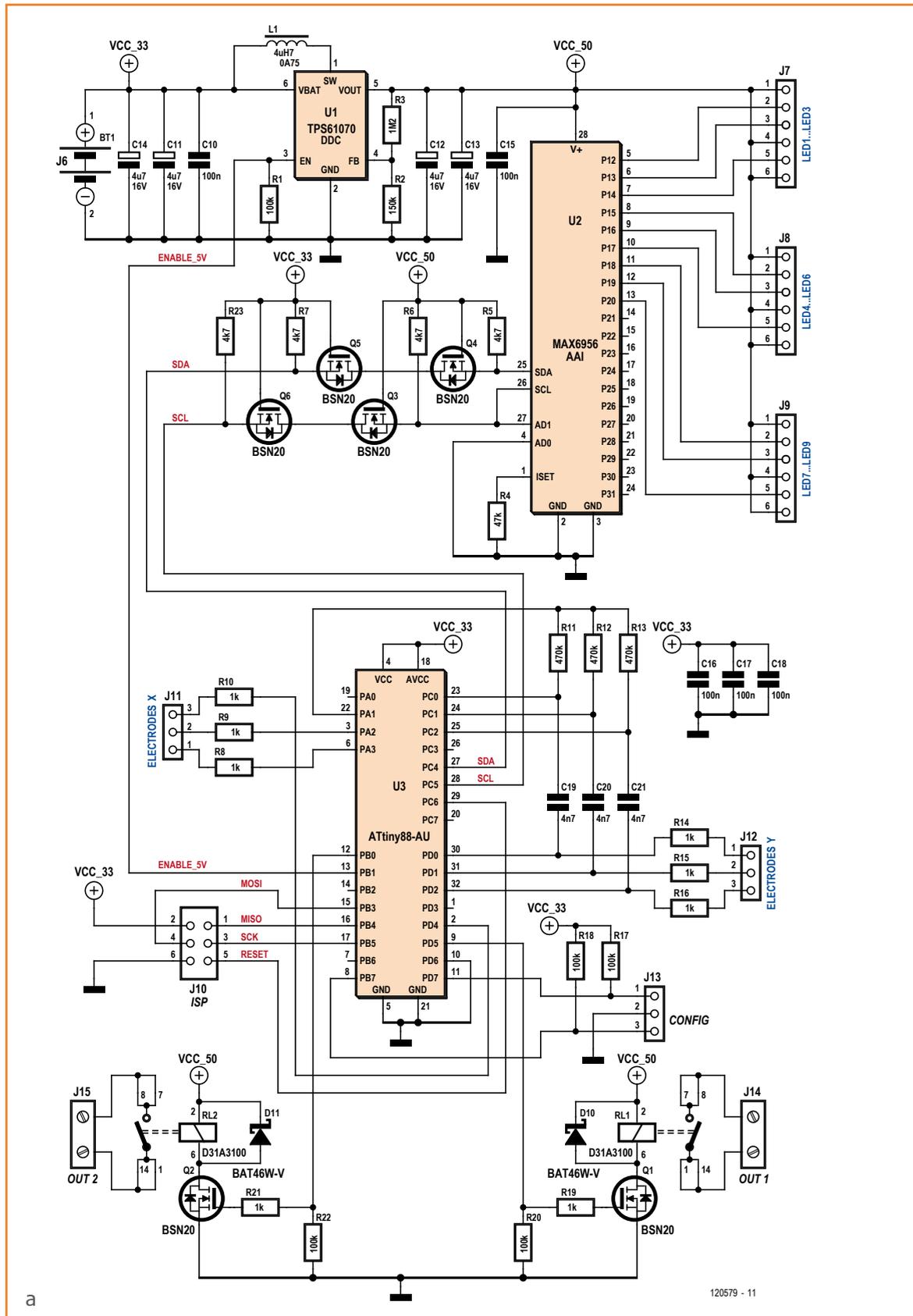
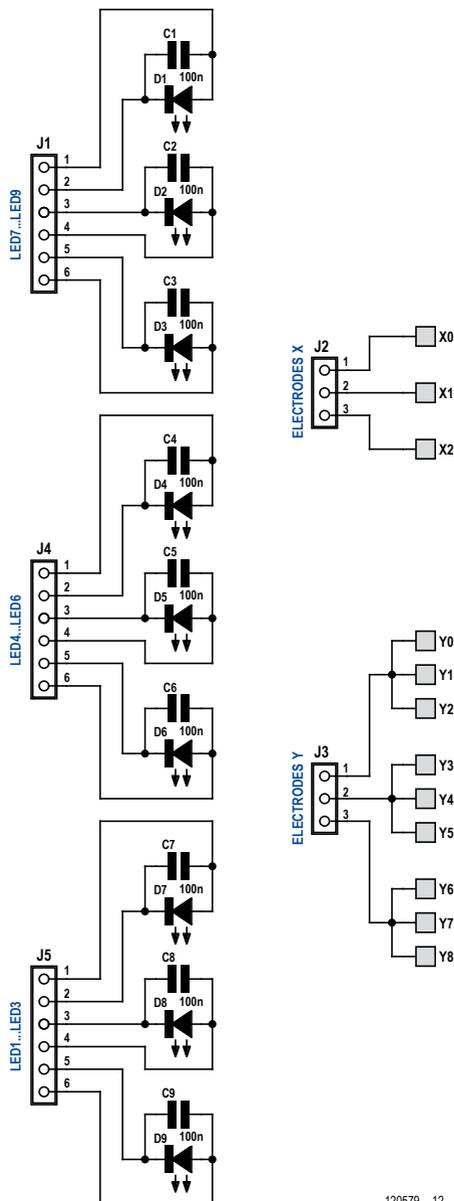


Figure 5.
Schéma du clavier tactile à reconnaissance de motifs.



120579 - 12

b

miques, sans oublier les indispensables diodes de roue libre D10 et D11. On les trouve parfois intégrées au relais, mais j'ai préféré les garder séparées pour pouvoir les adapter à d'autres références.

Deux entrées PB7 et PD7 du μC permettront au système de faire l'apprentissage des motifs à reconnaître. Elles devront être cachées pour des raisons de sécurité évidentes. La programmation d'un code se fait en plaçant un cavalier entre deux broches de J13 avant la mise sous tension du système.

Le soin porté à l'optimisation de la consommation électrique permet de se contenter de deux piles AAA. Le μC ATtiny88 qui passe la plupart de son temps à dormir, est alimenté directement par les piles. Son chien de garde intégré le réveille toutes les 1200 ms pour qu'il effectue une acquisition des touches. Si une présence est détectée, il passe en mode actif et met sous tension l'alimentation 5 V, assurée par U1, un régulateur élévateur, et ses composants annexes. Cette tension alimente les relais et les LED bleues dont la tension directe de 3,3 V environ est trop élevée pour les alimenter directement à partir des piles. Contrairement à d'autres régulateurs élévateurs, le TPS61070 présente en mode veille l'avantage de déconnecter complètement sa sortie et de ne plus consommer que 0,5 μA !

Pour qu'un pavé tactile comme celui-ci soit attractif, il faut non seulement qu'il soit illuminé, mais il faut encore que cet éclairage soit dynamique et suive les mouvements du doigt. La commande des LED (**fig. 5b**) est assurée par le μC , mais celui-ci ne pourrait pas fournir directement le courant aux LED. Cette tâche est confiée à un MAX6956, qui permet de piloter jusqu'à 20 sorties en leur fournissant un courant constant, configurable individuellement par le bus I²C. Le courant maximal d'U2 est limité par R4 à environ 20 mA par sortie. Comme le MAX6956 ne dispose pas de mode de veille à faible consommation, son alimentation est coupée quand le circuit est au repos.

Q3 à Q6 forment deux translateurs de niveaux bidirectionnels qui permettent la communication I²C entre les deux circuits intégrés U2 et U3 alimentés sous des tensions différentes. D'autre part, ils évitent que lorsque son alimentation est coupée, U2 soit alimenté par les broches SDA et SCL. Les MOSFET BSN20 sont particulièrement adaptés à cet environnement basse tension grâce à leur faible tension de commutation grille-source (V_{gs}).

Liste des composants

Circuit principal

Résistances (CMS1206) :

R1, R17, R18, R20, R22 = 100 k Ω

R2 = 150 k Ω

R3 = 1,2 M Ω

R4 = 47 k Ω

R5 à R7, R23 = 4,7 k Ω

R8 à R10, R14 à R16, R19, R21 = 1 k Ω

R11 à R13 = 470 k Ω

Condensateurs (CMS1206) :

C10, C15 à C18 = 100 nF 50 V

C11 à 14 = 4,7 μ F 16 V

C19 à C21 = 4,7 nF 50 V

Inductance :

L1 = 4,7 μ H 0,75 A, p. ex. Bourns - SRR3011-4R7YL, Farnell 1828144

Semi-conducteurs :

U1 = Texas Instruments - TPS61070DDC, Farnell 1461062

U2 = Maxim IC - MAX6956AAI, DigiKey MAX6956AAI+T-ND

U3 = Atmel - ATtiny88-AU, Farnell 1704570

D10 à D11 = BAT46W-V

Q1 à Q6 = BSN20, Farnell 1081309

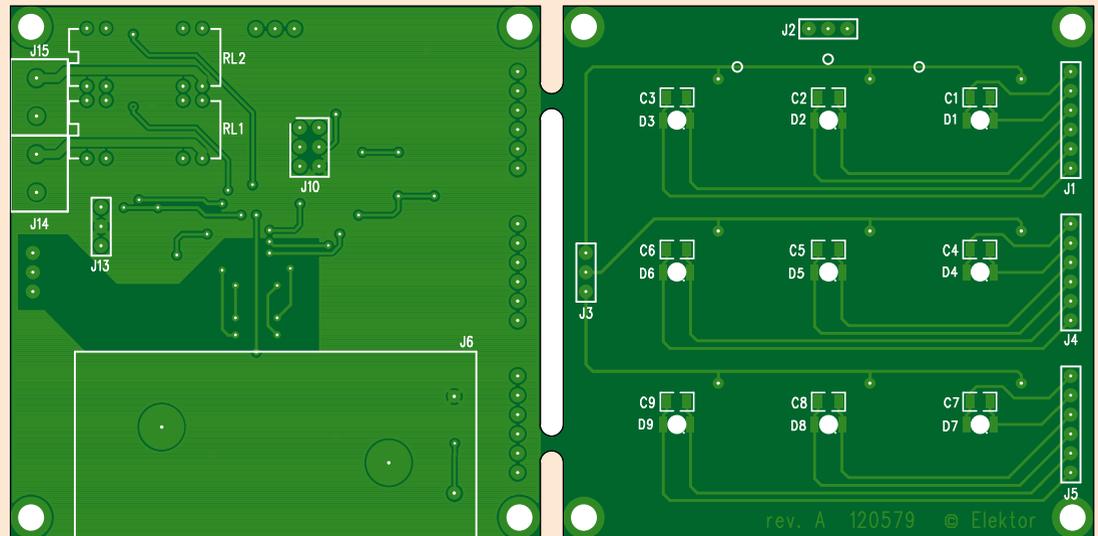


Figure 6.
Dessins des circuits imprimés. La cathode des LED est marquée par un minuscule point en bas à droite du symbole.

J10 est le connecteur compatible ISP qui permet la programmation et le diagnostic *in situ* du microcontrôleur.

Soudure et assemblage

Tous les composants sont disponibles dans les bonnes crémeries, ce sont presque tous des CMS, mais j'ai opté pour de gros boîtiers 1206 afin de faciliter le soudage aux moins expérimentés. Séparez d'abord les deux circuits imprimés (fig. 6) à l'aide d'un disque à tronçonner, d'une petite scie ou d'une pince coupante. Commencez par implanter les composants de la platine supérieure comportant les touches, les neuf LED et neuf condensateurs. Afin de garantir leur alignement, enfichez avant de les souder les parties mâles et femelles des connecteurs J1 à J5, J7 à J9 et J10 à J12 dans les platines superposées. La soudure d'U2 et U3 est délicate ; j'utilise la méthode forte qui consiste à souder d'abord toutes les broches entre elles sur le circuit à l'aide

d'une grosse panne, sans craindre les courts-circuits, puis j'applique de la tresse à dessouder pour absorber l'excès de soudure et ne laisser que le strict nécessaire au maintien des broches sur le cuivre.

L'assemblage mécanique de touches capacitives demande une attention particulière, d'autant plus que celles-ci sont rétro éclairées. Après de patientes recherches, j'ai fini par mettre la main chez www.plexiglas-shop.com sur du plexiglas de 3 mm, capable de diffuser en auréoles de l'ordre de 15 mm de diamètre la lumière émise par les LED (voir liste des composants).

Pour le bon fonctionnement des touches, il ne doit subsister **aucun jeu mécanique** entre le circuit imprimé et le diélectrique (c'est-à-dire l'indispensable plaque de plexi - inutile d'essayer sans, ça ne marche pas !). J'ai collé ces deux couches à l'adhésif double face transparent. Une fine couche de colle époxy sur toute la surface conviendrait aussi. Vous pourrez expérimenter avec d'autres

Divers :

J6 = support pour 2 piles AAA, p. ex. Keystone – 2468, Farnell 1650679
 J7 à J9 = barrette à 6 contacts, П 2,54 mm
 J10 = barrette à 2x3 contacts, П 2,54 mm
 J11 à J13 = barrette à 3 contacts, П 2,54 mm
 J14, J15 = bornier à vis, П 5,0 mm, p. ex. Camden-Boss - CTB5202/2, Farnell 1717001
 RL1, RL2 = relais Reed, p. ex. Celduc - D31A3100, Farnell 1214549
 circuit imprimé 120579-1
 plaque de plexiglas 60x60x3 mm truLED® Blanc WH72 GT, www.plexiglas-shop.com

Circuit tactile

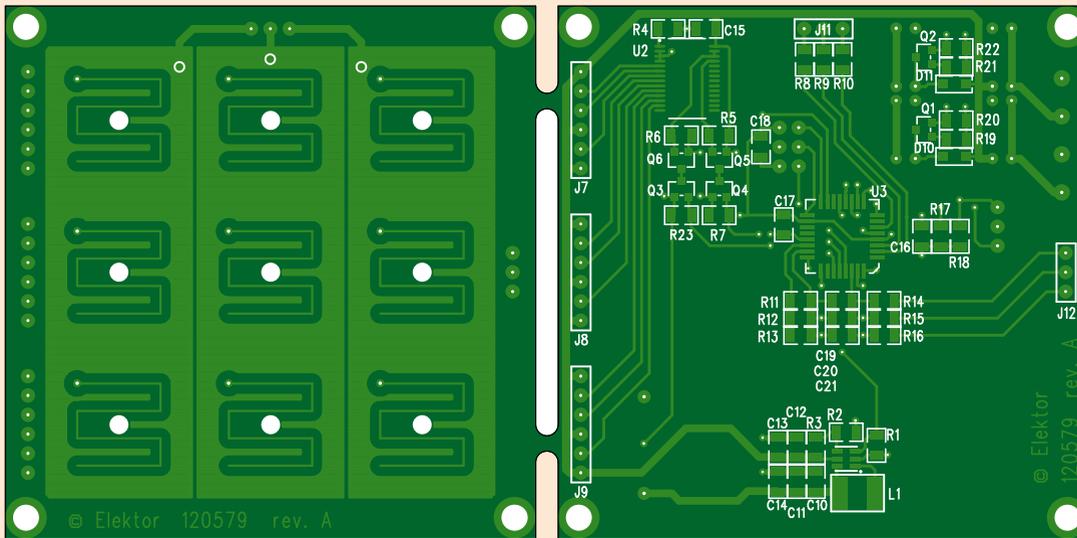
Condensateurs (CMS1206) :
 C1 à C9 = 100 nF 50V

Semi-conducteurs :

D1 à D9 = LED, bleu, CMS1206, p. ex. Kingbright - KPTD3216QBC-D, Farnell 8530050

Divers :

J1, J4, J5 = embase à 6 contacts, П 2,54 mm
 J2, J3 = embase à 3 contacts, П 2,54 mm



matériaux, transparents ou non, à condition qu'ils soient isolants, comme par exemple le verre qui est le diélectrique idéal. Le dessin de mes touches est optimisé pour une épaisseur de 3 mm. Si votre plaque diélectrique est plus ou moins épaisse, la détection sera moins bonne.

Les connecteurs J1 à J5 doivent être montés de telle façon que leurs broches n'affleurent pas à la surface de la platine capacitive. Il faut les souder du côté où ils sont montés. À défaut de quoi la plaque de plexi, qui doit être solidaire de la platine capacitive, devrait être découpée autour des contacts car ceux-ci dépasseraient. Ce serait mal pratique et inélégant.

Pour indiquer la position (et la forme) des touches vous utiliserez, à votre convenance, un vinyle adhésif découpé ou une peinture aérosol et des masques. Les cotes de la face avant que j'ai réalisée (**fig. 7**) vous serviront de base avant de laisser libre cours à votre créativité.

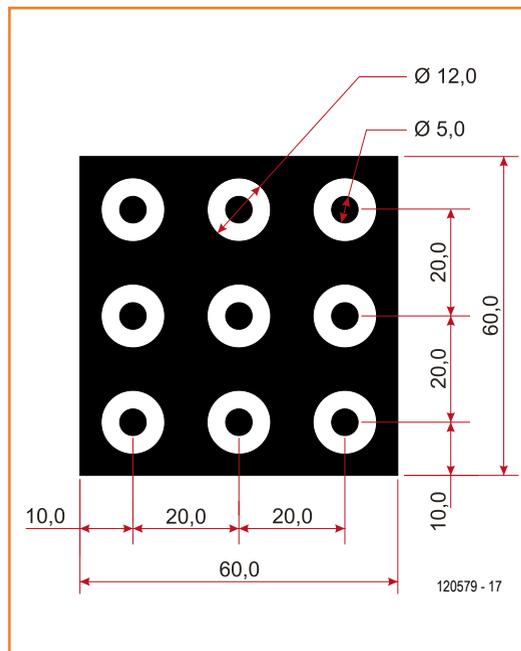


Figure 7. Proposition de décor pour la face avant.

Logiciel embarqué et bibliothèque *QTouch*

La simplicité de la conception matérielle des capteurs capacitifs est rendue possible principalement par l'implémentation dans la bibliothèque *QTouch* de plusieurs algorithmes assez élaborés. Elle gère p. ex. la calibration périodique automatique des mesures et suit les variations lentes de capacité dues aux variations des conditions environnementales (humidité). Elle permet également de comparer les valeurs des touches adjacentes pour ne garder que la plus forte d'entre elles, très pratique quand les surfaces sensibles sont petites et que nos gros doigts débordent (algorithme AKS, pour *Adjacent Key Suppression*). Enfin, pour éviter le phénomène de touche *collée* (c'est-à-dire activée malgré l'absence de contact), une durée maximale est paramétrable au-delà de laquelle une nouvelle calibration est effectuée. Ces différents paramètres sont ajustables dans le fichier *touch_config.h* ; j'ai gardé les valeurs par défaut. L'affectation des numéros de touches, leur sensibilité et l'appartenance à un groupe AKS sont eux définis dans la fonction *QT_config_keys* du fichier *QTouch.c*.

Le code source du logiciel est disponible et commenté en détails [3]. L'intégration de la bibliothèque *QTouch* suit les règles dictées par le document [2] et est illustrée par le pseudo-code disponible en **fig. 8**. À la mise sous tension, le système est initialisé par l'appel des fonctions *QT_config_sensors*, *qt_init_sensing* et *QT_set_parameters*. Par la suite, chaque appel de *qt_measure_sensors* lance une mesure. Si une variation de capacité est détectée, la bibliothèque effectue plusieurs acquisitions rapprochées afin d'infirmier ou de confirmer l'activation de l'une des touches. Cette répétition des acquisitions est essentielle pour la fiabilité des mesures tout en maintenant un temps de réponse rapide. En cas de changement dans l'état des boutons, la variable *u16_flagStatus* est mise à jour et c'est l'appel de la macro *GET_SENSOR_STATE()* qui vous indiquera l'état de chacun. La fréquence de mesure dépendra de votre logiciel et constitue un compromis entre réactivité, temps d'occupation du processeur et consommation électrique. Pour obtenir une réponse qui suive agréablement le glissement du doigt sur la surface tactile, j'ai choisi une période d'acquisition de 50 ms qui rend la détection très réactive. Pour une application plus classique où les touches sont activées une par une, on peut augmenter ce délai à 100 voire 200 ms. Au-delà, l'utilisa-

teur trouvera la détection lente, voire erratique. Autour de la détection tactile, le reste du logiciel est implémenté dans une machine à états impossible à reproduire ici en raison de son encombrement mais que je vous recommande de télécharger sur [4] pour l'étudier.

L'implémentation de la bibliothèque *QTouch* occupe entre 2 et 4 Ko de mémoire de programme selon les configurations et demande environ 250 octets de RAM. Elle utilise également ponctuellement *Timer 1* qu'il vaut donc mieux éviter d'utiliser durant les phases de mesure. Le code complet du *Pattern Lock*, compilé avec *IAR Embedded Workbench*, occupe un peu moins de 6 Ko de mémoire de programme et 500 octets de RAM et tient donc parfaitement dans l'ATtiny88.

Utilisation et mise au point

La sécurité offerte par ce système est un compromis entre inviolabilité et rapidité de déverrouillage. Il serait téméraire de lui confier l'ouverture de votre porte d'entrée. En revanche, l'allumage sécurisé de l'ordinateur du salon serait un rôle adapté. C'est dans cette optique que j'ai choisi une alimentation 100% autonome afin d'ouvrir au mieux l'éventail de ses possibilités. Cette alimentation par piles impose quelques compromis pour maintenir une autonomie suffisante. Les relais, gros consommateurs d'énergie, ne sont activés que pendant une demi-seconde à la reconnaissance d'un motif. Ce mode de fonctionnement est adapté au remplacement du bouton d'allumage d'un ordinateur ou au pilotage d'une gâche électrique de porte. Si vous souhaitez alimenter le montage en permanence par une alimentation stabilisée, il suffira de modifier quelques lignes de code pour rendre permanents l'activation des relais et l'allumage des LED.

Dès la mise sous tension, le système se met en repos tous feux éteints, mais effectue périodiquement des mesures. S'il détecte la présence d'un doigt sur n'importe quelle zone pendant plus de 1,2 s, il se réveille et allume les LED à faible puissance. L'utilisateur peut saisir son code de déverrouillage. Si le motif géométrique est correct, un **O** s'affiche brièvement et le relais correspondant est activé. Si le motif est erroné, c'est un **X** qui s'affiche. L'utilisateur dispose de trois tentatives pour saisir correctement un code, après quoi la serrure se bloque en allumant sa LED centrale pendant 10 s puis se rendort.

Pour programmer un nouveau motif, à la pre-

```

#include "QTouch.h"
#include "touch_api.h"

void main(void){
  // Configure the Sensors as keys or Keys With Rotor/Sliders in this function
  QT_config_sensors();
  // initialise touch sensing
  qt_init_sensing();
  // Set the parameters like recalibration threshold, Max_On_Duration etc
  QT_set_parameters();

  do{
    // Autonomous loop which performs capacitive sensing every 50ms
    if (g_u8_flagMeasure){
      g_u8_flagMeasure = 0;
      do{
        // one time measure touch sensors
        u16_flagStatus = qt_measure_sensors(g_u16_current_time_ms_touch);
        //Set flag if multiple measurements are required by library
        u16_flagBurst = u16_flagStatus & QTLIB_BURST_AGAIN;
      }
      while(u16_flagBurst);

      // One touch is in detect
      if (u16_flagStatus & QTLIB_IN_DETECT){
        u16_touchState = 0;
        //Find the touched key number
        for (i=0; i<9; i++){
          //Check if this key is in touch
          if (GET_SENSOR_STATE(i))
            u16_touchState |= (1<<i);
        }
      }
      else
        u16_touchState = 0;
    }

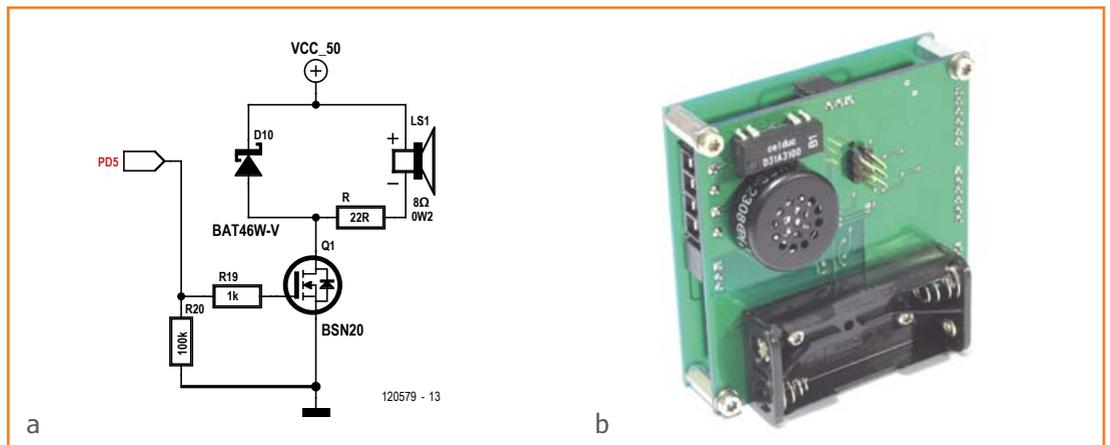
    //Process keys
    if (u16_touchState){
      //Look for which key is touched
      for (i=0; i<9; i++){
        if (u16_touchState & (1<<i)){
          u8_ledCurrent[i] = LED_CURRENT_HIGH; //Update corresponding led
          MI_PlayTone(i); //Play Tone
        }
        else
          u8_ledCurrent[i] = LED_CURRENT_LOW;
      }
    }
    else
    {
      for (i=0; i<9; i++){
        u8_ledCurrent[i] = LED_CURRENT_LOW;
      }
      MI_StopTone();
    }
    LED_Update(u16_ledValue, u8_ledCurrent); //Update led current
  }
  while (1);
}

//Timer 0 interrupt function, called every 50ms
__interrupt void MI_Timer0Interrupt(void){
  //Here every 50ms
  g_u16_current_time_ms_touch += 50;
  g_u8_flagMeasure = 1;
  return;
}

```

Figure 8.
Appel et utilisation de la bibliothèque *QTouch*. Ici l'extrait provient du logiciel du *PatternPiano* (fig. 9) une variante dont la structure est similaire mais plus épurée et donc plus simple à comprendre.

Figure 9.
Transformation du *Pattern Lock* en *Pattern Piano* (a). Un des relais est remplacé par un mini HP (b).



mière mise sous tension ou en cas d'oubli, coupez l'alimentation puis placez le cavalier sur J1 : entre les broches 1 et 2 pour l'activation de la première sortie, et entre les broches 2 et 3 pour la seconde sortie. Mettez le système sous tension (cavalier toujours en place), réveillez-le en mettant le doigt sur n'importe quel bouton puis saisissez votre nouveau motif, composé de quatre à neuf points. Une fois le code mémorisé, l'interface s'endort. Vous pouvez retirer le cavalier et utiliser la serrure.

Les électrodes Y passent par J3/J12. Evitez d'y poser vos doigts ou de placer une masse métallique à proximité pour ne pas perturber la détection.

En veille, la consommation du montage est de l'ordre de 5 μ A, ponctuée de périodes de réveil de 2,4 mA pendant 8 ms, toutes les 1200 ms. Le

courant moyen en veille s'établit ainsi autour de 21 μ A, l'autonomie avec deux piles AAA est de plusieurs années ! Les codes d'accès sont sauvegardés en EEPROM et donc maintenus même en l'absence d'alimentation.

(120579)

Liens & références

- [1] Atmel - QTAN0079 Sensor design guide www.atmel.com/Images/doc10752.pdf
- [2] Atmel - Atmel QTouch Library 5.0 www.atmel.com/Images/doc8207.pdf
- [3] Atmel - QtouchStudio www.atmel.com/tools/QTTOUCHSTUDIO4_3_1.aspx
- [4] www.elektor.fr/120579
- [5] david.ardouin.projects@gmail.com

Figure 10.
Pour faciliter le montage de la plaque sur le clavier, soudez les connecteurs par en-dessous pour qu'ils ne dépassent pas.



- ✓ Notre service après-vente – Votre atout
- ✓ Plus de 40 ans d'expérience
- ✓ Plus de 40.000 produits en stock
- ✓ Livraison rapide
- ✓ Pas de montant minimal de commande
- ✓ Grandes marques & véritable qualité

reichelt – Made in Germany

Nos clients profitent de la communication électronique rapide et simple de notre boutique en ligne. Nous pouvons répondre aux demandes de service après-vente en allemand et en anglais. L'équipe reichelt se fera un plaisir de vous répondre.

Notre niveau de qualité a été certifié conforme à la norme DIN EN ISO et garantit à nos clients un management global de la qualité et un niveau élevé et constant de la qualité s'agissant du traitement des commandes et de la livraison.

Eléments de construction

Platines d'essais

Platines d'essais

Pour la pose rapide de commandes d'essais électroniques souder.



9,95
STECKBOARD START

Le kit de débutant

Tout ce qu'il faut pour un démarrage rapide !

- 3x bornes de branchement, 4 mm
- 1x terminal de contact, 630 contacts
- 2x terminaux répartiteurs, 200 contacts
- avec kit de pontage de 70 pièces
- Dimensions : 183 x 95 mm



(STECKBOARD 4K7V)

- avec branchements et plaque de fond en aluminium

STECKBOARD 2K1V	Contacts 1280/100	7,90
STECKBOARD 2K4V	1280/400	10,90
STECKBOARD 3K5V	1920/500	13,90
STECKBOARD 4K7V	2560/700	16,90
kit de pontage de 140 pièces		
STECKBOARD DBS		3,95

Les experts techniques achètent ici

La qualité professionnelle à des prix discount !

AVR

Outil de développement pour contrôleur AVR 8 et 32 bits

- Possibilité de débogage On-Chip pour débogage Source-Level-Symbolic, NanoTrace et programmation Controller
- supporte la programmation via JTAG, SPI, PDI, aWire
- supporte le débogage via debugWire, JTAG, PDI, aWire

169,00
AT JTAG ICE3



Technique du boîtier

Boîtier du système

pro
Ma



Boîtier du système

3 Unités en hauteur, 10"/19"

- spécial pour les circuits imprimés en Euroformat
- vendu sous forme de kit à monter, montage simple
- avec poignées, creux des poignées et pieds en plastique
- aluminium anodisé, cache latéral bleu

	Ouverture	Profondeur	47,25
GEH SG 1-10	203,8 x 112,8 mm	236 mm	
GEH SG 1-19	427,8 x 112,8 mm	236 mm	67,30

Le boîtier professionnel pour le Raspberry Pi

- Snap-On-System (livré sans Raspberry Pi)

TEK-BERRY	blanc	4,95
TEK-BERRY SW	noir	4,95
TEK-BERRY TR	transparent	4,95



Opto-électronique

Ecrans



ELECTRONIC ASSEMBLY
making things easy

EA EDIP-TFT43ATP

196,00

IMMÉDIATEMENT PRÊT À FONCTIONNER.
EMBARQUÉ & INTELLIGENT

Tout est déjà à bord !

Ecran graphique TFT avec clavier tactile

- Ecran TFT : 4,3"/10,9 cm, 480 x 272 pixels, 16-bit color (65 536 couleurs)
- clavier tactile analogique
- RS-232, Bus I²C, Bus SPI
- 8 E/S numériques
- 2 entrées analogiques

Module LCD-DIP

Dotmatrix avec 4 x 20 caractères

- avec contrôleur standard compatible KS0073/HD44780
- Barrette à broches intégrée
- noir avec éclairage LED blanc
- Dimensions du module : 75 x 27 mm



LCD 204SW DIP **27,95**

Commander maintenant ! www.reichelt.fr

Hotline en langue anglaise : +49 (0)4422 955-360

Prix du jour !
Prix à la date du :
04.02.2013

reichelt elektronik
Votre partenaire compétent pour

Eléments de construction • Alimentation électrique • Technique de mesure • Outillages pour atelier
Domotique & technique de sécurité • Technique de réseau & informatique • Technique Sat/TV

mettez le cap sur Arduino (5)

David Cuartielles
(Espagne) et
Bobbie Cuartielles
(Suède)

Sus au voleur de gâteaux !



Un mystérieux voleur pique des gâteaux dans la boîte de la cuisine ! Après avoir réuni le conseil des sages, nous avons décidé de mettre en place un piège pour le confondre. Rassurez-vous, nous n'avons l'intention de blesser personne mais sommes fermement décidés à obtenir des preuves du délit.

La mission est claire et nous disposons de bon nombre d'outils pour arriver à nos fins. Passons nos costumes d'inventeurs et fabriquons une machine qui prend des photos du malotru qui ose s'introduire dans notre précieuse boîte. Sus au voleur de gâteaux !

Le matériel

Pour reproduire le projet de ce mois-ci il vous faudra :

- une carte Arduino Uno
- un ordinateur avec les *EDI Processing* et Arduino installés
- une webcam
- un câble USB
- une plaque à essais
- une résistance de 1 M Ω
- des fils
- un module *TinkerKit* : une LED blanche à haute luminosité + fil *TinkerKit*

L'idée

Nous allons créer un piège constitué d'un jouet fait maison (cf photo) qui surveillera en permanence la boîte à gâteaux. Lorsque quelqu'un touchera la boîte, une caméra cachée dans le jouet prendra une photo, et avec un peu de chance révélera le visage du voleur.

La capture de l'image sera réalisée par un programme écrit dans *Processing* qui se chargera de prendre des photos à l'aide de la caméra cachée dans le jouet. La caméra sera activée à l'aide d'une carte Arduino qui se chargera de détecter si quelqu'un touche notre précieuse boîte. La détection proprement dite sera faite en utilisant les périphériques des puces ATmega qui permettent de réaliser des capteurs capacitifs. Le concept est assez simple et l'intégration des composants le sera aussi. Nous allons commen-

cer par créer un *sketch Processing* capable de déclencher la prise d'une photo lorsque la carte Arduino, reliée à l'ordinateur par un port série, le lui demandera.

Installation de Processing

Téléchargez *Processing* depuis son site internet [1]. Si vous n'avez lu aucun des articles de la série *mettez le cap sur Arduino*, sachez que *Processing* est un environnement de développement multi-plateforme dédié à la création de programmes Java, Android et Javascript. Il vise les gens souhaitant apprendre à coder et faire des choses compliquées de manière simple. Nous allons utiliser une webcam ; assurez-vous donc que la vôtre fonctionne correctement avec *Processing*. Si vous utilisez Windows ou MacOS, allez faire un tour sur [2] si ça ne marche pas du premier coup ; si par hasard vous êtes un utilisateur de Linux (c'est mon cas), la page [3] vous indiquera les paquets à installer afin que *Processing* puisse commander votre webcam. Nous allons utiliser les bibliothèques *Video* et *Serial* de *Processing*, toutes deux installées par défaut avec l'EDI. Cette fois-ci il n'y aura donc aucune bibliothèque à installer.

Capturer une image (depuis la webcam)

Les fonctions vidéo de la bibliothèque de la dernière version de *Processing* s'appuient sur *GStreamer* [4], une suite logicielle capable d'accéder aux différents périphériques vidéo reliés à votre ordinateur, gérer les flux, décoder de la vidéo, etc. L'accès au flux vidéo en provenance d'une caméra reliée à votre ordinateur se fait à l'aide d'un descripteur de fichier. Par exemple, sur les machines dotées d'un système type Unix (comme Linux ou MacOSX), les périphériques vidéos se présentent comme des fichiers nommés `/dev/videoXX`, `XX` représentant un nombre identifiant chacune des caméras. Pour ce projet, j'utilise mon ordinateur portable, `/dev/video0` correspond à la caméra intégrée et `/dev/video1` à celle que j'ai placée dans le jouet.

La première chose à prendre en compte est que la résolution et la fréquence de capture de votre caméra peuvent être configurées de plusieurs façons. Je vous conseille d'exécuter le programme du **listing 1** qui affichera dans la console de l'EDI *Processing* les différentes configurations prises en charge par votre ordinateur.

Tableau 1.

index	nom	résolution	ips
[0]	/dev/video0	1280x720	10
[1]	/dev/video0	960x540	10
[2]	/dev/video0	800x448	15
[3]	/dev/video0	640x480	30
...			
[86]	/dev/video0	320x240	15

Listing 1

```
/**
 * Vérification des caméras dans Processing
 */

import processing.video.*;

Capture cam;

void setup() {

    String[] cameras = Capture.list();

    if (cameras.length == 0) {
        println("Aucune caméra n'est disponible.");
        exit();
    } else {
        println("Caméras disponibles :");
        for (int i = 0; i < cameras.length; i++) {
            println("[+" + i + "] " + cameras[i]);
        }
    }
}

void draw() {
}
```

Listing 2

```
if (cameras.length == 0) {
    println("Aucune caméra n'est disponible .");
    exit();
} else {
    println("Caméras disponibles :");
    for (int i = 0; i < cameras.length; i++) {
        println("[+" + i + "] " + cameras[i]);
    }

    // La caméra peut être initialisée à l'aide d'un
    // des éléments du tableau renvoyé par list():
    cam = new Capture(this, cameras[3]);
    cam.start();
}
```

Vous serez sans doute surpris par la longueur de la réponse ; dans mon cas, et rien que pour la caméra intégrée à mon ordinateur, j'ai obtenu une liste de pas moins de 86 configurations possibles, dont un extrait est disponible dans le **tableau 1**. Il vous faudra ensuite choisir la configuration souhaitée en fonction de la taille des images que vous désirez. J'ai choisi une résolution de

640 x 480 pixels, soit la configuration n° 3 du tableau.

Vous remarquerez qu'il peut y avoir plusieurs configurations possédant en apparence les mêmes paramètres, il vous faudra alors faire des essais pour déterminer celle qui donne les meilleurs résultats.

La sélection de la caméra devra se faire dans la fonction *setup* du programme. Il vous suffira de la modifier pour y inclure le code du **listing 2** qui permet d'activer la caméra sélectionnée.

Listing 3

```
void draw() {
  if (cam.available() == true) {
    cam.read();
  }
  image(cam, 0, 0);

  // pour que la détection du clavier fonctionne, vous
  // devez
  // au préalable avoir cliqué sur la fenêtre de
  // l'application (donné le focus)
  if(keyPressed) {
    if (key == 'p' || key == 'P') {
      saveFrame("pic-#####.png");
    }
  }
}
```

Listing 4

```
void draw() {
  if (cam.available() == true) {
    cam.read();
  }
  image(cam, 0, 0);
  String timeStamp = String.format("%02d", hour());
  timeStamp += ":" + String.format("%02d", minute());
  timeStamp += ":" + String.format("%02d", second());
  timeStamp += " " + year();
  timeStamp += "/" + String.format("%02d", month());
  timeStamp += "/" + String.format("%02d", day());

  text(timeStamp, 10, height-10); // incrustation sur l'image

  // pour que la détection du clavier fonctionne, vous devez
  // au préalable avoir cliqué sur la fenêtre de
  // l'application (donné le focus)
  if(keyPressed) {
    if (key == 'p' || key == 'P') {
      saveFrame("pic-#####.png");
      println("capture de l'image à : « + timeStamp); //
      // affichage dans la console
    }
  }
}
```

Utilisation du clavier dans Processing

Nous voulons capturer une image lorsque quelqu'un touche la boîte, mieux vaut donc commencer par vérifier que la capture fonctionne. Je vous propose de réaliser ce test en utilisant le clavier pour déclencher la capture d'une image. Modifiez le code du listing 2 pour y inclure celui du **listing 3**. Une image sera capturée à chaque fois que vous appuierez sur la touche "p" ou "P". La ligne en gras est un appel à la méthode qui permet d'enregistrer une image au format PNG dans le dossier du *sketch* sous le nom *pic-#####*. Les dièses représentent le numéro de l'image. Chaque image sera enregistrée avec un numéro différent. Vous remarquerez que **saveFrame()** n'enregistre que le cadre de l'application, pas le reste du bureau.

Datons l'image !

Étant donné que **saveFrame()** écrit une copie fidèle du cadre de votre application dans un fichier, il est possible d'ajouter des informations par dessus l'image. Dans notre cas, nous voulons savoir *quand* la boîte a été ouverte et les précieux gâteaux dérobés. En jargon policier : il nous faut une preuve de l'heure du délit. Le code du **listing 4** nous permettra d'incruster la date et l'heure sur l'image de la webcam. Un message sera également affiché sur la sortie standard.

Un exemple de capture est reproduit en **figure 1**. La résolution, à cause de la webcam, est faible mais cela satisfera les détectives que nous sommes.

La prochaine étape consiste à écrire le code Arduino permettant de détecter lorsque quelqu'un touche la boîte et en notifier l'ordinateur.

Détection tactile avec Arduino : la bibliothèque CapacitiveSensor

Dans l'archive 120745-11.zip correspondant à cet article vous trouverez un dossier *Arduino*

contenant un sous-dossier *libraries*. L'archive peut être téléchargée gratuitement sur [6]. Copiez le contenu de ce dossier dans le dossier *libraries* de votre installation Arduino. Ce dernier se trouve dans votre dossier *sketchbook*, habituellement *My Documents/Arduino* si vous êtes sous Windows ou *Documents/Arduino* pour MacOS et Linux.

La bibliothèque *CapacitiveSensor* a été créée en 2008 par Paul Badger [5] et n'est livrée qu'avec un seul, mais très utile, exemple. Je vous recommande de jeter un coup d'œil à la page de l'Arduino *playground* référencée en [5]; vous y trouverez tout ce que vous avez besoin de savoir sur le fonctionnement des capteurs capacitifs. Pour faire court, la bibliothèque utilise deux entrée/sortie numériques : une pour envoyer une impulsion et l'autre pour la mesurer. Les deux pattes sont reliées entre elles par une résistance (1 M Ω ici) et la patte servant à la mesure est reliée à un fil ou une pièce métallique qui se comportera comme un condensateur. La simple présence du corps humain modifiera la capacité de la charge sur la patte de mesure et donc le temps nécessaire pour que l'impulsion envoyée par l'autre patte n'atteigne un niveau qui soit interprété comme haut. La bibliothèque *CapacitiveSensor* mesure ce temps — s'il dépasse une certaine valeur, il est très probable qu'un humain soit en contact avec la boîte et nous pouvons photographier le coupable la main dans le sac, ou plutôt dans la boîte. Les capteurs capacitifs sont omniprésents dans nos vies numériques. Ils sont tellement sensibles que l'on peut les cacher derrière du plastique, du bois, ou n'importe quel autre matériau isolant. Pour les étendre, il suffit d'ajouter un peu de métal et d'ajuster quelques composants passifs (résistances et condensateurs).

Le circuit que nous allons ajouter à notre Arduino se résume à une résistance de 1 M Ω et quelques fils. Pour compléter le tout, et comme notre vigile numérique devra être capable de prendre des photos la nuit, nous avons ajouté une carte *TinkerKit* dotée d'une LED à haute luminosité afin d'éclairer la scène. La bouche de notre vigile sera faite d'un morceau d'une feuille de plastique acrylique (Plexiglass) afin que la lumière puisse passer au travers. Référez-vous au schéma de la **figure 2** pour plus de détails sur l'assemblage. Côté logiciel, le code Arduino nécessaire pour réaliser ces opérations est plutôt simple (**Listing 5**). Ceux d'entre vous qui ont l'expérience des Arduino remarqueront que j'ai mis en gras les lignes appelant la nouvelle bibliothèque. Vous

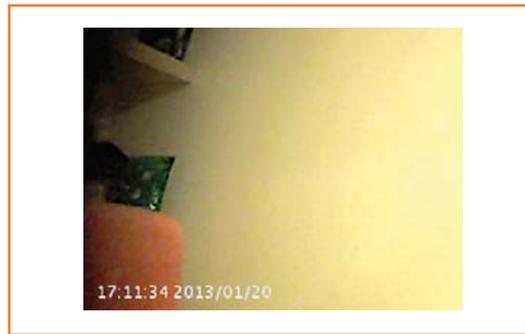


Figure 1.
Une image horodatée en provenance de la webcam.

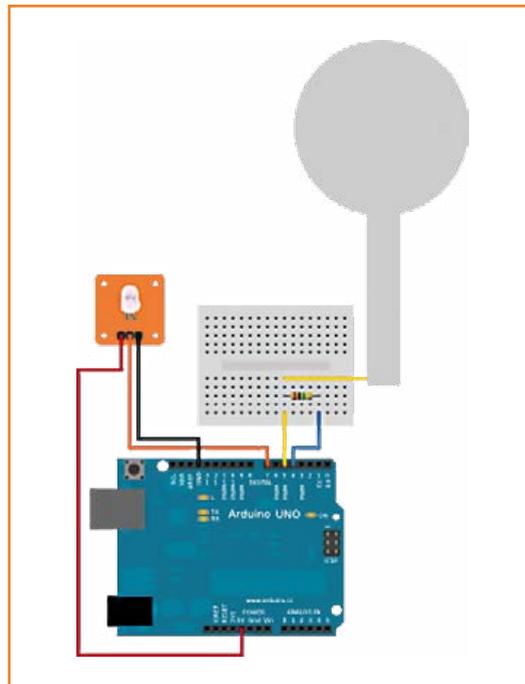


Figure 2.
Schéma de notre projet (fait avec Fritzing).

devez configurer **THRESHOLD** pour que cela fonctionne chez vous. Vous verrez en **figure 3** l'antenne que nous avons découpée dans du papier d'aluminium. Il faudra la placer sous la boîte à gâteaux. La valeur mesurée par la bibliothèque variera en fonction de la taille et de la forme de cette antenne. Voici les étapes nécessaires pour configurer le seuil de détection **THRESHOLD** :

- reliez votre circuit au papier d'alu
- mesurez la valeur à vide **DEFAULT_VALUE**. Cette valeur dépend de la taille de votre antenne et devrait se trouver aux alentours de 200
- touchez l'antenne et notez la valeur **MAX_VALUE** affichée dans la console série
- calculez **THRESHOLD = (MAX_VALUE + DEFAULT_VALUE) / 2** et changez la valeur dans votre code

Lorsque la valeur mesurée sur notre capteur fait maison sera supérieure à **THRESHOLD**, le pro-

gramme enverra sur le port série le caractère 'p' à l'application *Processing*. Il ne nous reste qu'à modifier le programme *Processing* pour que la prise de photo soit déclenchée depuis le port série et mettre le piège en place.

Déclencher la caméra à partir de l'Arduino

Il vous suffira d'inclure la bibliothèque *Serial* dans votre programme *Processing* et lire les données du port série au fur et à mesure de leur arrivée (**listing 6**).

Listing 5

```
#include <CapacitiveSensor.h>

#define THRESHOLD 500 // Cette valeur devra être réglée

// déclarez les pattes à utiliser
// résistance de 1M entre les pattes 4 & 5
// LED haute luminosité sur la patte 7
CapacitiveSensor cs_4_5 = CapacitiveSensor(4,5);
int ledPin = 7;

void setup() {
  // décommentez pour l'autocalibration de la voie 1
  //cs_4_5.set_CS_Autocal_Millis(0xFFFFFFFF);

  // configuration du port série
  Serial.begin(9600);

  // configuration de la patte pour la LED
  pinMode(ledPin, OUTPUT);
}

void loop() {
  long sensorReading = cs_4_5.capacitiveSensor(30);

  if (sensorReading > THRESHOLD) {
    // illumination de la LED
    digitalWrite(ledPin, HIGH);

    // ordonner à l'ordinateur de prendre une photo
    Serial.write('p');

    // attente et extinction de la lumière
    delay(2000);
    digitalWrite(ledPin, LOW);
  }
  else {
    // décommentez pour la configuration du système
    // Serial.println(sensorReading);
    // delay(200);
  }
}
```

Mise en place du piège

Notre ami Tien Pham a dessiné un petit monstre à la silhouette de cyclope et l'a fait découper au laser dans le *FabLab* du coin. Tien a habilement mit à profit la webcam pour constituer le cou de notre vigile. Le capteur de la caméra est caché derrière les yeux et la LED à haute luminosité derrière la bouche. Il reste assez de place derrière notre petit monstre pour y cacher la carte Arduino et la plaque à essais que nous avons utilisés. Le chef-d'œuvre de Tien est visible en **figure 4**.

Comme expliqué plus haut, le capteur est fait de papier d'aluminium. Il a la forme d'un 'i' coiffé d'un très gros point. Le point est l'antenne sur laquelle nous allons disposer la boîte. Le fil relié à la plaque à essais et la patte 5 est collé à l'antenne avec du ruban adhésif.

Vous remarquerez que nous avons utilisé une boîte métallique ; cela devrait également fonctionner avec d'autres matériaux mais il faudra sans doute rendre le capteur plus sensible en augmentant la valeur de la résistance, 10 MΩ par exemple.

Le mot de la fin

Voici un projet idéal pour s'amuser un week-end de pluie ou quand un voleur rôde dans les parages. Tout a été assemblé à partir de reliquats d'autres projets. Il reste encore pas mal d'améliorations possibles : vous pourriez par exemple souder tous les composants sur un *shield* de prototypage, chercher un meilleur matériau pour la construction du capteur, etc. Le résultat actuel est visible en **figure 5**.

Avec ce projet, je voulais montrer qu'il est possible d'assembler un prototype crédible avec très peu de choses, même lorsqu'il s'agit d'électronique numérique. Si je trouve le temps, je modifierai le code *Processing* pour envoyer les images directement sur un serveur web, rendant l'information accessible à distance en un instant. Mais ! Voilà un excellent exercice pour vous, cher lecteur !

Je me réjouis de la publication récente par Elektor de l'excellent **livre de Clemens Valens [7]**, qui met en œuvre la plateforme Arduino comme outil de maîtrise des microcontrôleurs en général. Il le fait de façon vivante, plaisante même, et en même temps très sérieuse. Je suis sûr que ce livre est promis à un beau succès.

(120745 – version française : Kévin PETIT)

Références

- [1] Le projet Processing : <http://processing.org>
- [2] Ça coince entre webcams & Processing : http://wiki.processing.org/w/Video_Issues
- [3] GStreamer Linux & webcam dans Processing : <http://goo.gl/ZTCdZ> ou <http://forum.processing.org/topic/how-is-video-on-linux-handled#2508000001764427>
- [4] Le site officiel de Gstreamer : <http://gstreamer.freedesktop.org/>
- [5] Bibliothèque CapacitiveSensor de Paul Badger : <http://playground.arduino.cc/Main/CapacitiveSensor>
- [6] www.elektor-magazine.fr/120745
- [7] Maîtrisez les microcontrôleurs à l'aide d'Arduino www.elektor.fr/arduino

Remerciements

Merci à Paul Badger pour la bibliothèque Arduino *Capacitive Sensor* qu'il a créée en 2008, à Paul Stoffregen qui a adapté le code pour l'EDI Arduino version 1 et ultérieures et à Tien Pham du FabLab de Malmö pour la conception de notre monstrueux vigile.



Figure 3. Notre capteur tactile fait maison.



Figure 4. Le cou de notre gardien est constitué de la webcam.

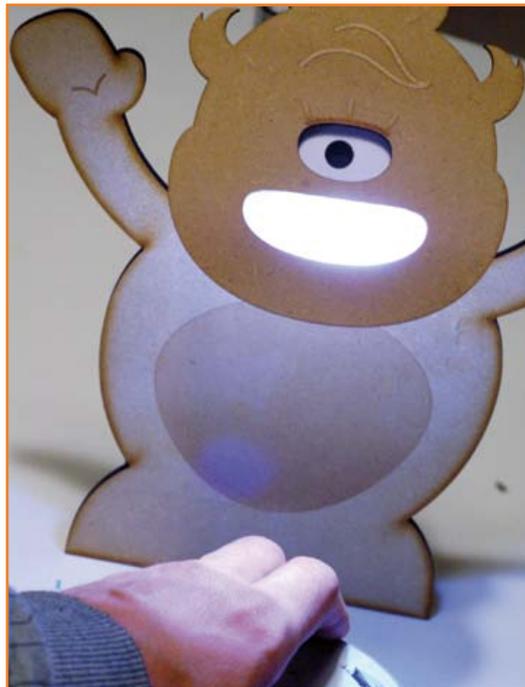


Figure 5.
La main dans le sac !
Qui est notre voleur ?

Listing 6

```
import processing.video.*;
import processing.serial.*;

Capture cam;
Serial myPort; // le port série

void setup() {
  [...]

  // nous n'avons pas besoin d'une grande vitesse de capture
  frameRate(1);

  // affichage des ports série disponibles
  println(Serial.list());
  // ouverture du port utilisé à la vitesse choisie
  myPort = new Serial(this, Serial.list()[0], 9600);
}

void draw() {
  [...]

  if (myPort.available() > 0) {
    int inByte = myPort.read();
    if(inByte == 'p') {
      saveFrame("pic-#####.png");
      println("capture de l'image à : " + timeStamp);
    }
  }
}
```

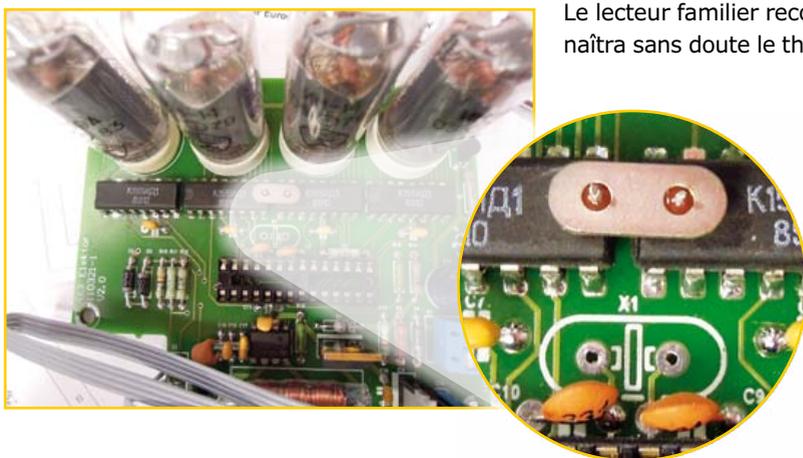
quartz muet

Thijs Beckers
(Elektor)

Il n'y a pas que les CMS qui peuvent poser des problèmes d'implantation ou d'encombrement. C'est ce que j'ai constaté avec un quartz tout ce qu'il y a d'ordinaire, à monter sur un circuit imprimé normal et de conception pourtant réputée professionnelle.

Vous pensez peut-être que je vais chercher la petite bête pour alimenter ma rubrique mensuelle sur cette page ? Pas du tout, je n'ai pas eu à la chercher, elle était bien là, comme le montrent la photo et le détail agrandi.

Le lecteur familier reconnaîtra sans doute le ther-



momètre hygromètre à tubes Nixie publié par Elektor en juin 2012 [1]). Pour la photo, le quartz a été dessoudé et posé à l'envers pour qu'on voie mieux : la taille des deux îlots de soudure est plus grande que la zone isolée autour des broches du quartz. Si ce celui-ci est implanté à ras du circuit imprimé, son boîtier métallique entrera en contact avec les pastilles et les court-circuitera. Le quartz ne risque pas d'osciller !

Une fois décelé, le problème est facile à résoudre. Il suffit d'un intercalaire isolant ou de veiller à maintenir le quartz à un millimètre ou deux de la carte au moment de souder.

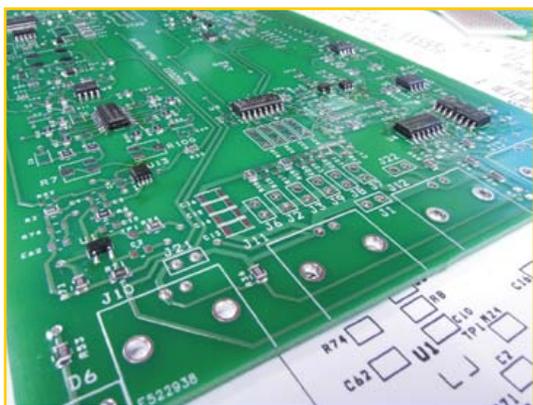
C'est un détail anodin, mais la différence entre celui qui le connaît et celui qui l'ignore se chiffrera, en cas de panne, en heures de recherche perdues. La prochaine fois que vous aurez un problème de signal d'horloge sur un circuit à quartz, vous saurez par où commencer vos investigations.

(120750)

Lien :

[1] www.elektor.fr/110321

qu'est-ce qui mijote ?



Vous avez sans doute déjà remarqué l'un des poids lourds de ce numéro, le deuxième article consacré à notre LCR-mètre de précision. Cet appareil est au point, trois prototypes du circuit imprimé que nous présenterons le mois prochain fonctionnent parfaitement. Pour l'heure nous en rédigeons la notice, assez volumi-

neuse parce que détaillée, notamment pour ce qui concerne la procédure de réglage, en grande partie automatique, au terme de laquelle l'appareil s'en tient aux 0,05% que l'on attend de lui. Il n'est pas étonnant que tout ce qu'il y a à

dire sur un appareil aussi élaboré ne tienne pas dans la revue. En plus de la vingtaine de pages que nous lui avons déjà consacrées, en plus du troisième article à publier le mois prochain, il y aura donc un *Mode d'emploi* et une *Notice de mise en route* téléchargeables !

Jean-Jacques Aubry, son concepteur, et le labo d'Elektor n'ont pas ménagé leurs efforts pour ce morceau de bravoure (qui ne pâlit pas devant un *Hameg HM8118 Programmable LCR Bridge*). La photo donne un avant-goût du montage : des CMS presque partout. Nos prototypes sont assemblés et soudés à la main, mais tout le monde n'aura pas la patience, l'outillage, le temps et le savoir-faire requis. C'est pourquoi **elektorPCB-service** proposera le module prêt à l'emploi, avec ou sans la carte d'extension.

(130092)

régulateurs de tension à découpage

avec composants traversants

Philip J., étudiant stagiaire il y a quelques mois au labo d'Elektor, inspiré par l'article sur les **régulateurs de tension à découpage** publié dans Elektor n° 413 en nov. 2012 [1]), s'est mis en tête de concevoir à son tour un tel substitut aux célèbres régulateurs tripodes. Limité qu'il était par son budget d'étudiant, une de ses préoccupations prioritaires fut la réduction des coûts. Philip est donc parti sur une carte simple face qu'il pourrait graver chez lui, de préférence sans aucun CMS, mais avec d'ordinaires composants traversants (*through-hole technology*). L'encombrement minimum et la compatibilité broche à broche avec les classiques boîtiers TO-220 étaient deux autres de ses critères importants.

Après quelques recherches, Philip a dérogé pour l'un des composants à la règle qui lui imposait la THT, ce qui du coup l'a obligé à prévoir un mini circuit imprimé adaptateur. Cela concernait le régulateur abaisseur de type TPS62150 3-17 V 1 A 3 MHz de *Texas Instruments* — identique à celui du circuit original, mais en boîtier 16QFN au pas de 0,5 mm.

Voilà le résultat de son étude. Outre l'attention accordée à la conformation de l'adaptateur, il a également fallu raccourcir au maximum les boucles de réaction et de C_{out} . À part le format du condensateur du filtre de sortie, obligatoirement du type à faible résistance série équivalente (ESR) et le régulateur CMS déjà mentionné, rien que des composants ordinaires.

Résultat : à peine 5 € dépensés pour le tout. C'était donc bien parti. Vinrent les essais.

Et la déception. En charge, le circuit était loin de fournir les 5 V attendus. Même très loin. Fort des

conseils de Raymond, le concepteur des régulateurs de tension à découpage que Philipp pensait égaler (et rêvait peut-être de dépasser), ce dernier a fini par arriver à obtenir 5 V en sortie de son circuit, mais sans pouvoir en tirer plus de 80 mA.

La régulation et la tension de sortie résiduelle ne méritaient pas qu'on en parle ici autrement qu'avec compassion. Dans le dossier laissé par son géniteur à la fin de son stage, nous n'avons pas retrouvé de copies d'écran d'oscilloscope de la tension de sortie. Les aurait-il effacées pour se faire oublier ?

Un stage est fait pour apprendre et on n'apprend jamais mieux qu'en faisant des erreurs. Le tout, pour apprendre beaucoup, est d'en faire chaque fois de nouvelles.

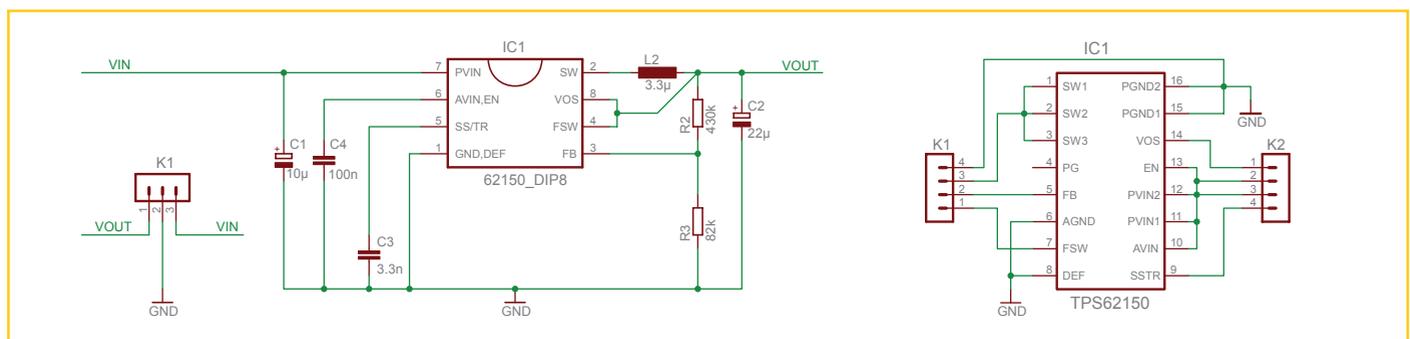
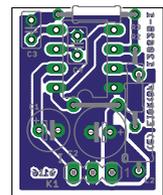
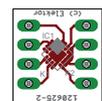
Si **vous** voyez comment améliorer le circuit laissé en l'état par Philip à la fin de son stage, n'hésitez pas à vous en inspirer. Toute sa documentation (schéma sous *Eagle v6*) est téléchargeable librement [2]. Et pour discuter du projet, il suffit de se brancher sur .Labs [3].

Thijs Beckers (Elektor)

Liens

- [1] www.elektor.fr/120212
- [2] www.elektor.fr/120625
- [3] www.elektor-labs.com/120625

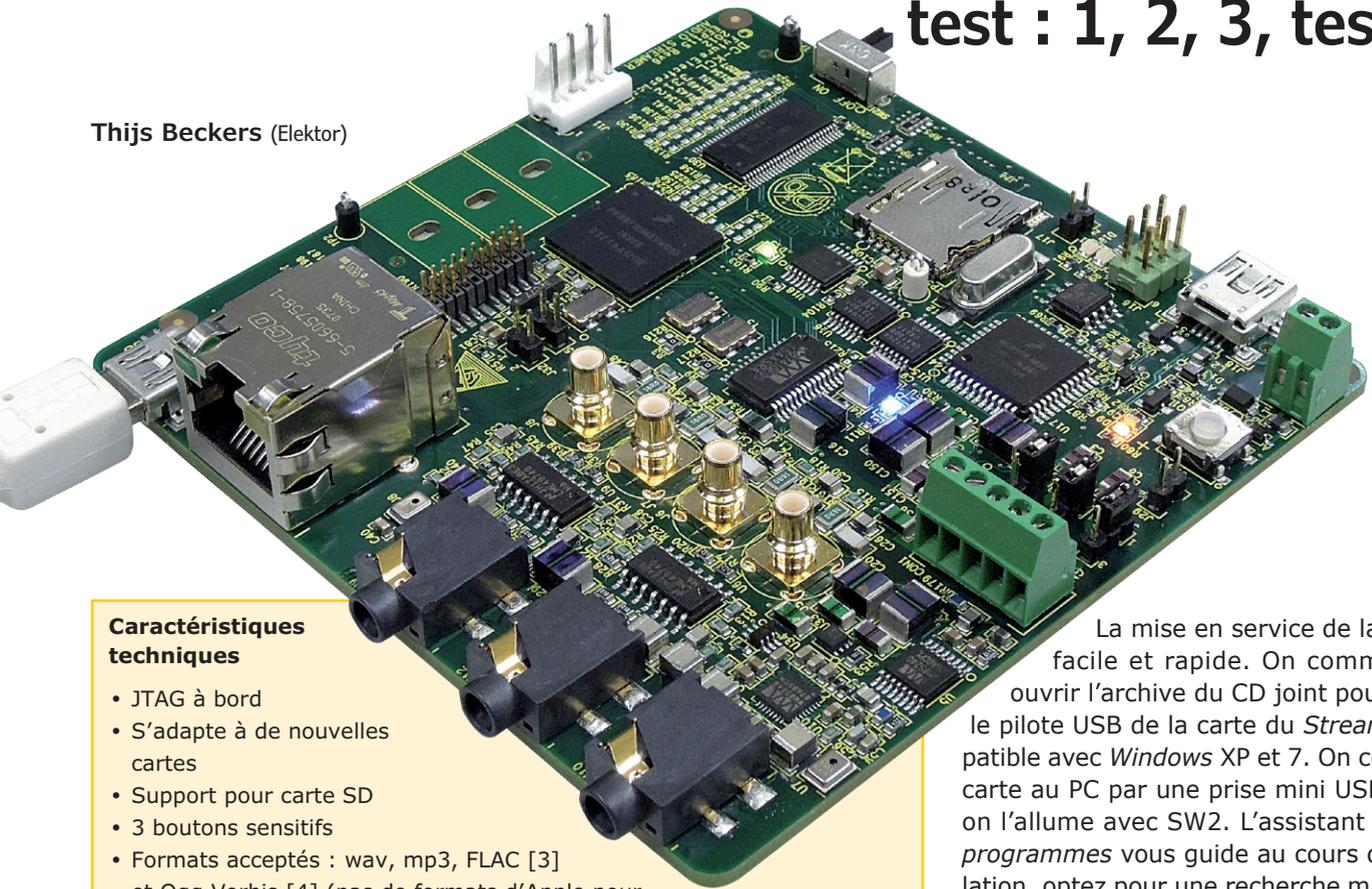
(120625)



flux continu audio

test : 1, 2, 3, test...

Thijs Beckers (Elektor)



Caractéristiques techniques

- JTAG à bord
- S'adapte à de nouvelles cartes
- Support pour carte SD
- 3 boutons sensitifs
- Formats acceptés : wav, mp3, FLAC [3] et Ogg Vorbis [4] (pas de formats d'Apple pour cause de licences)
- Codec WM8904 [2] à très basse puissance
- CNA : WM8524 (bas coût) & WM8741 (haut de gamme)
- Microcontrôleur Freescale Kinetis K60 Cortex-M4 ARM à 150 MHz
- SRAM de 8 Mo (256 Ko x 36)
- Condensateurs multicouches à film de polymères haut de gamme
- Interface Micro-Blox à 160 voies

Nous avons reçu l'un des plus récents kits de développement de *Future Electronics*, le *Streamer Audio* [1] pour l'essayer. L'intention de la firme est de diffuser ce membre de la famille *Micro-Blox* comme « système de développement rapide à des fins de validation de concept pour OEM », destiné à fournir aux concepteurs un moyen rapide et confortable de mise en œuvre de nouvelles applications avec la dernière génération de convertisseurs N/A audio et les codecs de *Wolfson Microelectronics* [2]. En encadré, un résumé des caractéristiques.

La mise en service de la carte est facile et rapide. On commence par ouvrir l'archive du CD joint pour installer le pilote USB de la carte du *Streamer*, compatible avec *Windows XP* et *7*. On connecte la carte au PC par une prise mini USB J20 puis on l'allume avec SW2. L'assistant *nouveaux programmes* vous guide au cours de l'installation, optez pour une recherche manuelle du pilote et sélectionnez celui du fichier zip. L'installation terminée, vous pouvez écouter en continu de l'audio en provenance de la connexion USB ou du réseau. Le guide de démarrage rapide fournit une bonne description des réglages de la connexion réseau. Une interface graphique simple, *Audio Demo*, facilite la configuration et l'utilisation de certaines options de la carte.

Après le choix de la connexion, un menu déroulant accessible par l'onglet *Configure* permet le choix du CNA et du codec, la sélection de la sortie (ligne ou casque), d'un égaliseur paramétrique, de différentes courbes de réponse des filtres, du mode antirepliement, du silencieux doux et de la source pour l'horloge maître : *Kinetis K60* ou oscillateur externe.

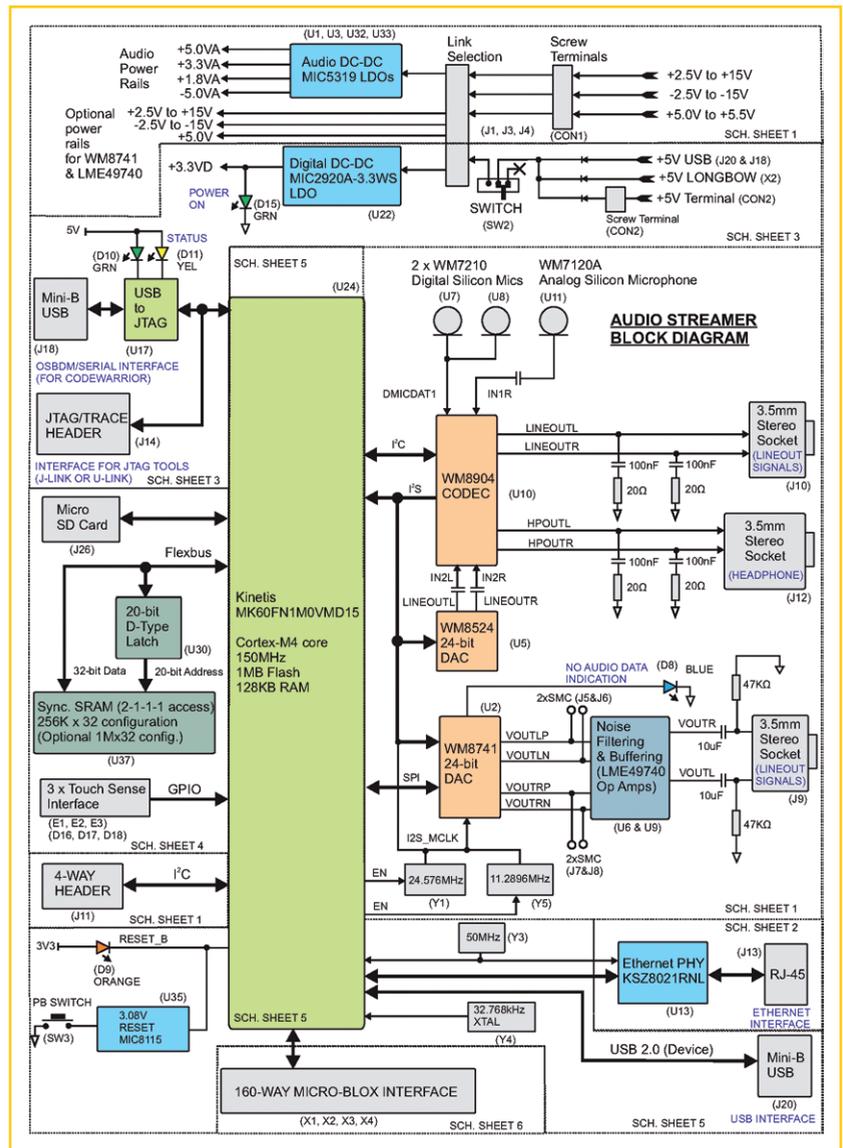
La collection des fichiers chargés d'origine est visible sous l'onglet *Play*. Vous pouvez y ajouter vos propres fichiers de test, sur votre PC ou sur la carte SD dans le support. Deux glissières règlent volume et balance du signal de sortie. La carte peut aussi enregistrer au format *wav* sur

le PC connecté. Il y a pour cela deux sources de sons sur la carte : un microphone stéréo numérique sur puces, deux WM7210 en couple stéréo, et un microphone analogique sur puce aussi, un WM7110. Les deux alimentent un codec à bas coût, le WM8904. Voilà donc une belle plateforme de test de base pour un tas d'applications, telles que la sécurité ou les communications par interphone. Orienté vers les applications hi-fi, le CNA haut de gamme WM8742 de *Wolfson Microelectronics* constitue une excellente base pour une utilisation de lecture audio en flux continu sur SD, USB et Ethernet. Les connecteurs SMC J5-8 délivrent des signaux de sortie de ligne symétriques issus directement du CNA et des amplificateurs opérationnels et le rapport S/B peut être amélioré par un filtrage à leur niveau. Les circuits audio sont dotés de condensateurs multicouches à film de polymères. Pour les applications qui ne requièrent pas la qualité hi-fi, une variante existe avec le CNA WM8524. Le décodage complexe des flux en FLAC et Ogg Vorbis ne demande que 30 à 40 % de la puissance de calcul de la puce *Kinetis*, il reste donc là de grandes ressources pour ajouter des applications en parallèle. Si vous n'avez besoin que du décodage mp3, nettement moins exigeant en calculs, vous pouvez choisir un plus petit micro-contrôleur *Kinetis* et supprimer la SRAM dans l'application définitive.

Pour débiter dans le développement d'un lecteur de flux audio, le *Workbench Embedded IAR*, version 6.3 ou supérieure, pour ARM est un environnement recommandé. Le code source de l'application *AudioDemo*, rédigé en C#, est disponible gratuitement chez *Future Electronics*. La carte utilise le système d'exploitation en temps réel MQX pour lequel *MQX Software Solutions* de *Freescale* offre au concepteur tout ce dont il a besoin.

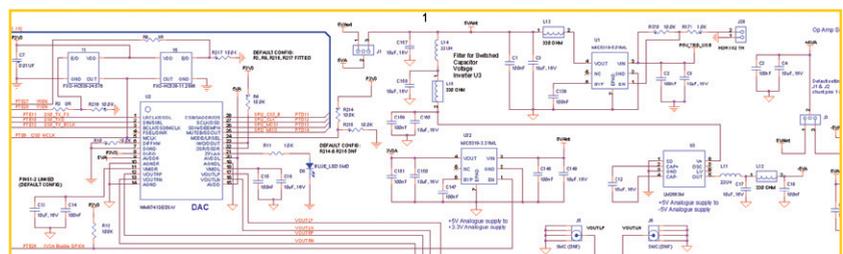
Pas mal de fonctions ne sont même pas mises en œuvre dans l'application *AudioDemo*, il reste donc un monde à explorer. En conclusion, nous pouvons affirmer que le *Streamer Audio* constitue, pour le concepteur, un moyen simple de réaliser un lecteur de flux. Avec la documentation et le logiciel offert, on peut développer complètement pareille application en un minimum de temps, éviter le stade du prototype pour le matériel et entreprendre directement le développement du logiciel pour arriver très vite à un système qui fonctionne parfaitement.

(120699 – version française : Robert Grignard)



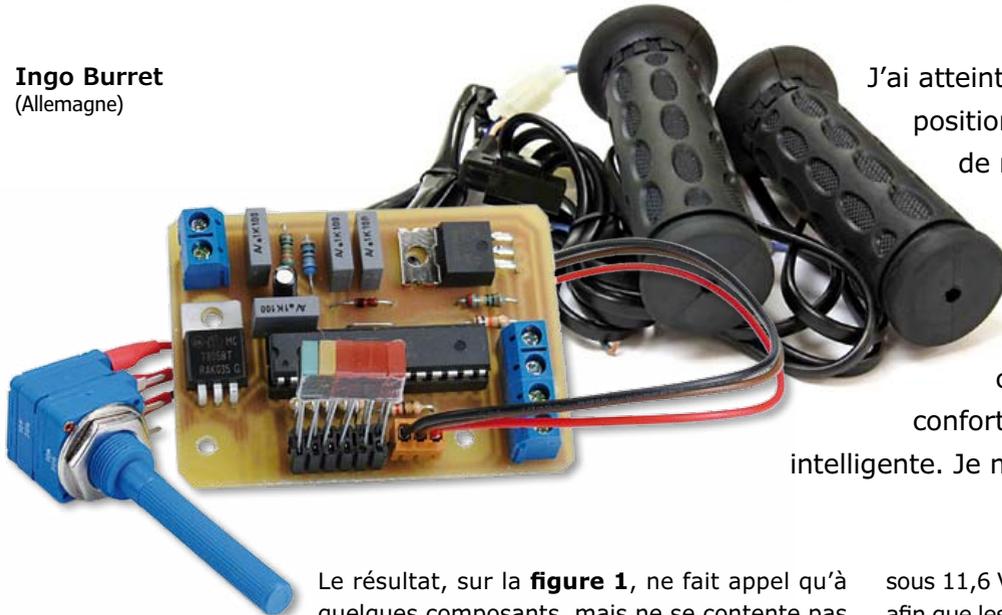
Liens

- [1] www.my-boardclub.com/future_blox/miniblox.php?blox=32
- [2] www.wolfsonmicro.com
- [3] <http://fr.wikipedia.org/wiki/FLAC>
- [4] <http://fr.wikipedia.org/wiki/Vorbis>



commande de guidon chauffant pour motos et gyropodes

Ingo Burret
(Allemagne)



J'ai atteint un âge où le confort « à deux positions » des poignées chauffantes de ma moto me paraît rudimentaire. D'autant plus que mes modestes connaissances en matière de microcontrôleurs m'ont incité à imaginer pour ce problème une solution (plus) confortable, (plus) élégante et (plus) intelligente. Je me suis donc mis au travail.

Le résultat, sur la **figure 1**, ne fait appel qu'à quelques composants, mais ne se contente pas de réchauffer bêtement les poignées. La fonction principale est assurée par un signal à modulation de largeur d'impulsion (MLI ou PWM) avec une fréquence de 1,8 kHz. Issu du port OC1A du microcontrôleur ATmega8, il pilote un MOSFET IRFZ34N. Ce transistor convient pour le chauffage de poignées classiques de 15 à 20 W telles qu'elles sont disponibles dans le commerce. Le signal PWM module l'intensité du courant fourni par la batterie sous 12 V et qui circule vers la masse à travers les poignées chauffées. Selon la largeur des impulsions du signal de commande, fixée par le motard à l'aide du potentiomètre P1, la production de chaleur sera donc dosée finement. La position du curseur de P1 est numérisée par le convertisseur analogique-numérique ADC0 et visualisée par quatre voyants à LED (PD5 à PD2). Au démarrage, le logiciel assure également que les poignées chauffent à plein régime pendant environ 30 s ; c'est ce qu'indique la LED PD1. La sixième LED est un simple témoin de mise sous tension.

À travers le diviseur de tension R9/R10 et le convertisseur A/N ADC1, le logiciel dans l'ATmega8 surveille également la tension aux bornes de la batterie. On sait qu'au ralenti la recharge d'une batterie est très faible. C'est pourquoi lorsque la tension de la batterie tombe en des-

sous 11,6 V, le signal MLI est inhibé par le logiciel afin que les poignées chauffantes ne consomment pas de courant. Dès que la tension de service aux bornes de la batterie remonte, le logiciel réactive le chauffage. Certains de ces paramètres peuvent être modifiés dans l'en-tête du programme.

Le circuit imprimé est compact (**fig.2**), l'assemblage facile, tous les composants sont traversants. Pour le contrôleur, prévoir un support, afin de pouvoir le reprogrammer facilement. Le petit circuit imprimé et le μ C sont proposés prêts à l'emploi, et pour les programmeurs il y a aussi les codes source C et hexadécimal [1].

Une moto n'est ni un atelier ni un salon ; l'électronique devra donc être mise à l'abri des projections d'eau. Ne mégotez pas sur le boîtier et vérifiez-en l'indice de protection international (IP) ; je recommande par exemple le RP1025C Hammond avec un IP de 65. Ces deux chiffres indiquent, sur une échelle de 0 à 8, la résistance l'un à l'eau (ici 5), l'autre aux corps étrangers (ici 6) [2]. Avec un 5, vous ne passerez pas dans la fosse des Mariannes, mais au moins votre boîtier sera protégé contre les projections d'eau. Ledit boîtier a un couvercle transparent : vous pouvez donc souder les LED (rectangulaires) directement sur le circuit imprimé. Pas de trous ! Étanchéité oblige, le circuit imprimé lui aussi devrait être fixé sans percer de trou dans le boîtier. Pour les

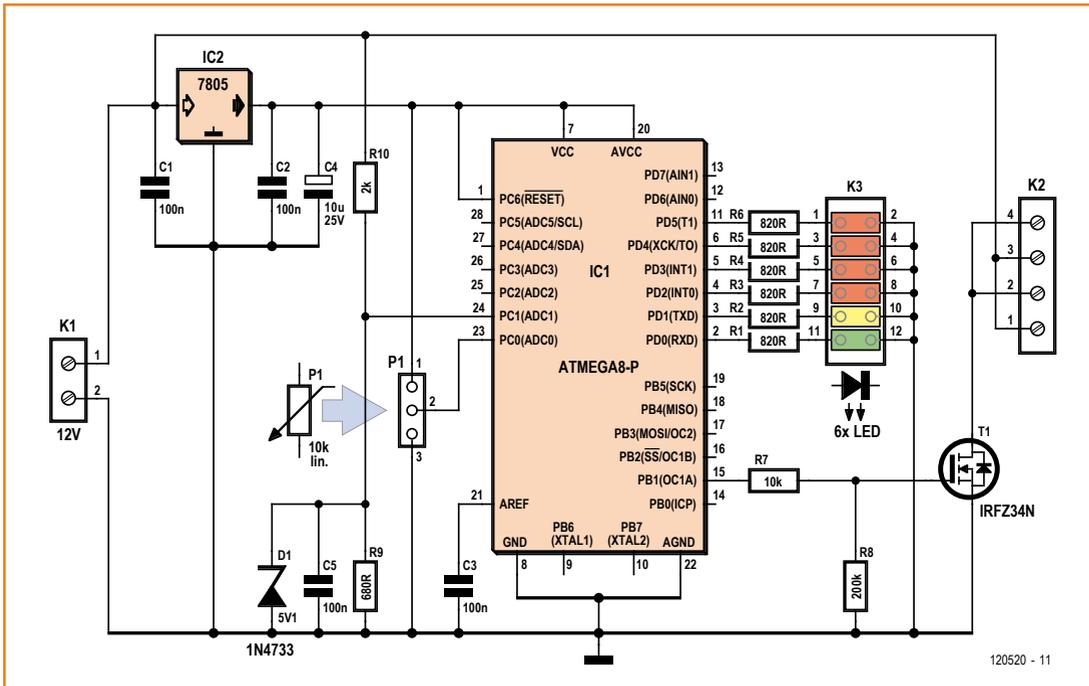


Figure 1.
Un microcontrôleur pour se chauffer les doigts.

câbles, je ne vois malheureusement guère d'autre possibilité que de percer. Utilisez des passe-fils dont vous renforcerez l'étanchéité avec du joint siliconé ou de la vaseline. Pour la traversée de l'axe du potentiomètre, il existe éventuellement des canaux étanches (pensez à enduire le joint de vaseline). Ce n'est là qu'une des possibilités de monter ma commande de chauffe-poignée. À vous de l'adapter à vos besoins et bonne route !
(120520)

Liens

- [1] www.elektor-magazine.fr/120520
- [2] <http://busse-yachtshop.de/info/ip-norm.html>

Liste des composants

Résistances :

- R1...R6 = 820 Ω
- R7 = 10 kΩ
- R8 = 200 kΩ, 1 %
- R9 = 680 Ω
- R10 = 2 kΩ, 1 %
- P1 = pot. 10 kΩ lin.

Condensateurs :

- C1,C2,C3,C5 = 100 nF/100V
- C4 = 10 μF/25 V

Semi-conducteurs :

- IC1 = ATmega8-16PU (programmé 120520-41)
- IC2 = 7805
- T1 = IRFZ34N
- D1 = zener 5V1, par ex. 1N4733A
- LED1 à LED6 = LED 3 mm rect. vert, jaune, rouge

Divers :

- Support DIL à 28 broches
- K1 = connecteur à 2 broches, 5 mm
- K2 = connecteur à 4 broches, 5 mm
- K3 = embase 2 x 6 points
- circuit imprimé 120520-1

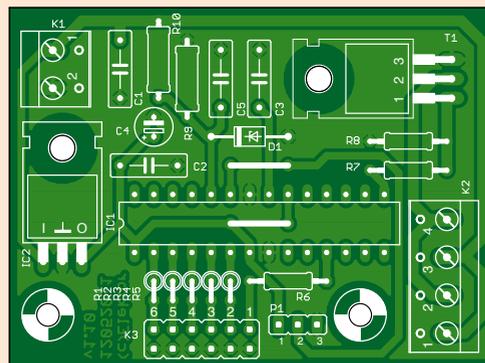


Figure 2.
Un circuit imprimé peu encombrant.

suivi de tension parcimonieux

Rolf Blijleven
(Pays-Bas)

L'électronique autonome sur cellules solaires, c'est possible, à condition de ne pas devoir travailler en permanence. Même la très faible récolte d'une cellule photovoltaïque, on peut l'engranger jusqu'à en avoir assez pour alimenter un circuit de temps à autre. Mais comment savoir qu'il y en a en suffisance ? Avec un moniteur de tension ! Le problème, c'est que lui aussi consomme de l'énergie. Il ne faudrait pas qu'il vienne marauder dans notre provision d'énergie, ou alors le moins possible. Ce sont les petits microampères qui font toute la différence !

En janvier de cette année-ci, nous avons discuté du principe de la récolte gratuite et des économies d'énergie. Le défi, c'est de faire fonctionner un circuit uniquement sur une petite source d'énergie,

sans alimentation externe ni pile. Dans le présent projet, la source est une petite cellule solaire à 5 € qui donne au maximum 1 V à 100 mA. La valeur maximale, une telle cellule ne la délivre que par une journée ensoleillée, quand la cellule est bien perpendiculaire aux rayons du soleil. Si le ciel est nuageux, elle ne fournit plus que quelques centaines de millivolts et quelques dizaines de microampères. Pourtant, il est possible, avec si peu d'énergie, de faire encore quelque chose d'utile. Pas en permanence, mais un certain nombre de fois par jour.

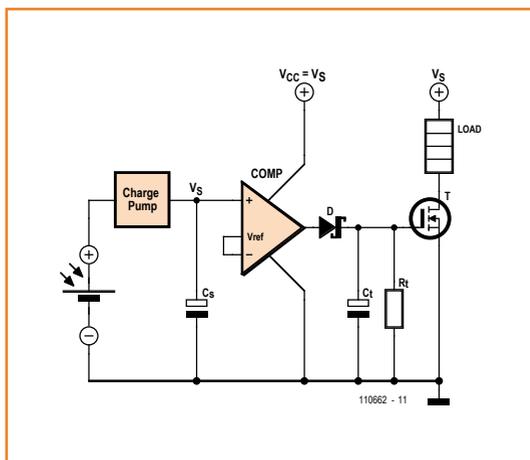


Figure 1.
Synoptique général du circuit de récolte d'énergie.

Le projet

Le diagramme fonctionnel est à la **figure 1**. La sortie de la cellule solaire est raccordée à un multiplicateur de tension qui charge le condensateur réservoir Cs. Le magazine Elektor de janvier 2011 a exposé en détail quelques-uns de ces multiplicateurs de tension, aussi nous contenterons-nous ici de le présenter sous forme de bloc fonctionnel. L'amplificateur dans le schéma sert de comparateur, il est alimenté par le condensateur réservoir. En pratique, il lui faut des résistances autour de lui. Quand l'entrée + atteint la tension de seuil, la sortie bascule au niveau haut, vers la tension d'alimentation Vs et du coup, le transistor devient passant. Après le retour à l'état d'origine de la sortie de l'amplificateur opérationnel, le transistor est maintenu en conduction par Rt et Ct pendant un peu de temps encore. Il en résulte que l'appareil reste alimenté jusqu'à ce que la provision d'énergie dans Cs s'épuise. Certains appellent parfois cette construction *solar engine* (un terme surfait pour un simple oscillateur à relaxation¹). On en trouve d'autres sur l'internet, par exemple sur [1].

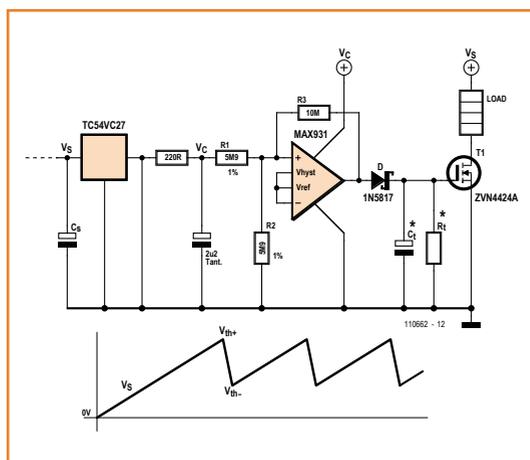


Figure 2.
Schéma d'un moniteur de tension parcimonieux. Tous les semi-conducteurs sont disponibles en CMS et en composants standard traversants.

Lors d'une matinée nuageuse, le courant de charge n'est que d'environ 20 à 30 μA et donc le condensateur réservoir se charge, mais très lentement. Il nous faut quand même vérifier tout le temps s'il y a assez d'énergie emmagasinée et pour cela, pas question d'utiliser un temporisateur, puisque nous n'avons pas de batterie. Avec un moniteur de tension MAX8212 de chez Maxim [2, 3] comme comparateur, les expériences ont bien marché, mais pas à notre entière satisfaction. Cette petite puce soustrait à peu près 6 μA dans sa résistance de réglage, facilement le quart du courant glané par temps nuageux.

Une solution pratique

La **figure 2** propose une meilleure solution. Le TC54VC27 est un détecteur de tension. Ce petit intégré à trois pattes consomme moins de 1 μA aussi longtemps que la tension de seuil de 2,7 V dont il est doté n'est pas atteinte, c'est-à-dire pendant la majeure partie du cycle de charge.

Au-dessus de ce seuil, le TC54 donne simplement en sortie ce qui se trouve à l'entrée. Cette transition doit se faire en douceur. En fait, la sortie du TC54 fournit aussi la tension pour le comparateur, un MAX931 de Maxim (courant de repos de 4 μA). S'il reçoit la tension brutalement, il se peut que sa sortie passe tout de suite à l'état haut, ce qui occasionnerait une brève conduction du transistor et un début de chute de Vs. Dans ces circonstances, Cs n'arriverait jamais à la tension voulue, ce que nous refusons évidemment. Aussi, le réseau RC de 220 Ω et 2,2 μF empêche-t-il que cela se produise. Le MAX931 reste hors service jusqu'à ce que la tension de seuil Vth+ soit atteinte, elle est déterminée par R1, R2 et R3 (soit 3,1 V avec les valeurs indiquées). Le décalage avec les 2,7 V semble petit, mais il est bien réel. Vous le remarquerez si vous y branchez un petit moteur. Le calcul de R1 et R2 est expliqué en long et en large dans le cahier de caractéristiques du MAX931 [4]. Entre 2,7 V et la tension de seuil, le moniteur consomme 5 μA tout au plus.

Il existe des TC54 avec différentes tensions de seuil intégrées entre 1,4 et 7,7 V, voyez le cahier de caractéristiques [5] à ce sujet. Et effectivement, nous en avons choisi un dont la tension de seuil se situe tout juste sous Vth+. Vous allez sans doute vous demander si un comparateur est vraiment nécessaire. La réponse est affirmative, parce que le transistor doit commuter de façon bien tranchée.

Dans ce cas-ci, le TC54 commence à conduire dans les environs de 2,75 V. Si nous accrochons le MOSFET directement derrière le TC54, il va commencer à conduire un peu, la tension sur Cs va diminuer, le TC54 va recommencer à bloquer et tout l'attirail n'arrivera jamais à fonctionner. Une résistance de couplage entre la sortie et l'entrée du TC54 n'y changerait rien, parce qu'elle ne ferait que consommer davantage sur Cs.

La constante de temps de Ct et Rt détermine la durée pendant laquelle le transistor reste passant. Le condensateur Cs ne doit pas se vider plus que nécessaire pour que l'appareil qui y est branché fasse son office. Supposons que l'appareil ensemble avec la résistance R_{dsON} du MOSFET représente 25 Ω et que Cs se compose de trois condensateurs électrolytiques de 4 700 μF , soit 14,1 mF au total. Alors, Cs se déchargerait complètement en 0,35 seconde. Mais si 0,2 s est un temps suffisant, cela veut dire que Cs conserve un tiers de sa charge, il sera d'autant plus vite rechargé pour la prochaine fois. Prenez alors pour Ct 2,2 μF et, pour Rt, 100 k Ω conviendront. La courbe de la tension en fonction du temps est reproduite dans le graphique.

Les différences sont minimes et pourtant bien perceptibles. Je n'avais pas envie de surveiller tout le temps un voltmètre pendant les essais du circuit. Aussi ai-je construit un carillon avec un petit moteur électrique de manière à connaître la périodicité à laquelle s'opère le plein d'énergie (et quel temps il fait dehors). Avec ce moniteur économe, le carillon résonne quelques fois par jour même par temps de pluie. Mais avec des appareils plus gourmands, il ne sonnait jamais par une météo pareille.

(110662 – version française : Robert Grignard)

Liens & références :

- [1] www.beam-wiki.org/wiki/Solar_engine
- [2] www.maxim-ic.com/datasheet/index.mvp/id/1273
- [3] www.iamwhen.com/archives/53-Herbert-1701-Species-B-Generation-1.html
- [4] www.maxim-ic.com/datasheet/index.mvp/id/1219
- [5] <http://ww1.microchip.com/downloads/en/DeviceDoc/21434h.pdf>



Figure 3.
Un carillon sonne chaque fois qu'il y a assez d'énergie récoltée. Pas besoin de rester planté devant un appareil de mesure.



¹Le principe de fonctionnement et des exemples d'application de cet oscillateur sont bien expliqués dans le volume 1 du **Traité de l'électronique** (Horowitz & Hill, Elektor) au chapitre 5.13.
www.elektor.fr/tt1

mon disque bosse dur dix LED pour en témoigner

Karsten Böhme (Allemagne)

La plupart des ordinateurs sont équipés d'une LED qui clignote lors des accès au disque dur. Ce témoin a beau être lumineux, il n'en reste pas moins flou. Que diriez-vous d'un afficheur à dix LED pour connaître avec précision l'activité de votre disque dur ?

La puissance des ordinateurs personnels et la vitesse de rotation des disques durs sont devenues impressionnantes. Nous pouvons aujourd'hui utiliser sur nos PC des applications autrefois réservées à des stations de travail spécialisées. Nous ne sommes toutefois jamais à l'abri d'un blocage de nos logiciels : difficile alors de savoir si le programme est bloqué ou simplement en train de solliciter le disque dur. Être informé de l'activité matérielle du disque dur serait dans ce cas utile.

Une solution logicielle aurait pu être envisagée, mais cet afficheur d'activité de disque dur à réaliser soi-même possède plusieurs avantages : il ne consomme pas de puissance de calcul supplémentaire, est indépendant du système d'exploitation et, abstraction faite du micrologiciel, n'est pas sujet aux risques de plantages et de bogues. Le marché des accessoires pour modification de boîtier est pléthorique, mais curieusement aucun revendeur n'offrait un tel afficheur... jusqu'à Elektor.

Dix LED à l'aide !

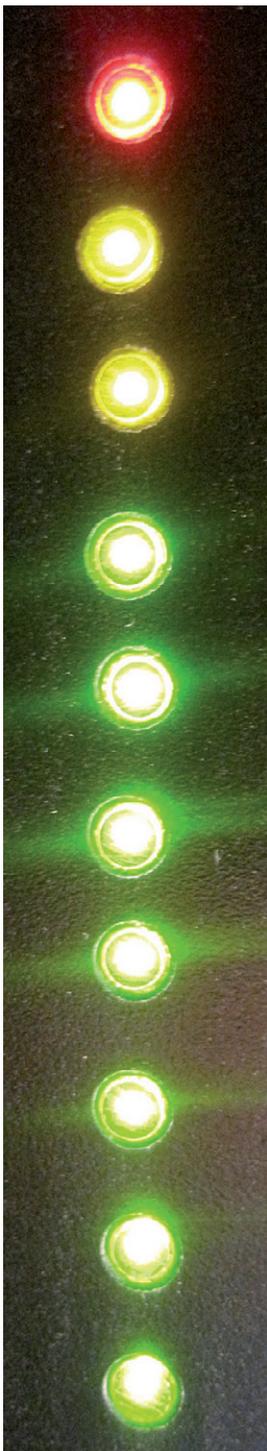
L'indication fournie par un afficheur typique à une seule LED est qualitative et grossière : un clignotement indique si le ou les disques durs internes

sont actifs ou non, c'est tout ! La solution présentée ici est de nature quantitative, claire, et réactive : l'intensité de l'activité du disque dur est indiquée par intervalles de 10 % et l'information actualisée cinq fois par seconde.

Mais comment est réalisé ce bargraphe ? Après tout, il n'existe ni sortie analogique ni interface numérique qui transmette les données d'accès au disque dur. Karsten a donc considéré que le scintillement de la LED commandé par la carte-mère était une sorte de signal modulé en largeur d'impulsion. L'astuce fonctionne étonnamment bien. Le programme prélève le signal lumineux de la LED, puis mesure quel a été le pourcentage de niveau haut pendant 200 ms. Les 10 LED sont allumées pour 100 % de niveau haut, éteintes pour 0 %. La photo de l'afficheur monté sur le PC de Karsten montre les 10 LED activées. Il a choisi des LED vertes pour les sept premières (intervalles de 1 à 70 %). Les deux LED jaunes indiquent un taux d'occupation appartenant à l'intervalle 71 à 90 %. La LED rouge s'allume lorsque le taux d'activité du disque dur est compris entre 91 et 100 %.

Décodeur et pilote de LED

Quel meilleur candidat qu'un microcontrôleur pour mettre en œuvre un tel principe de fonctionnement ? Le héros de la **figure 1** est un simple ATtiny2313 d'Atmel, accompagné d'un optocoupleur, de LED et de quelques autres composants. L'ensemble est dépouillé. Le signal de scintillement est prélevé sur la broche de la carte-mère à laquelle est relié le connecteur de la LED DD (disque dur) du boîtier. Le transfert des signaux

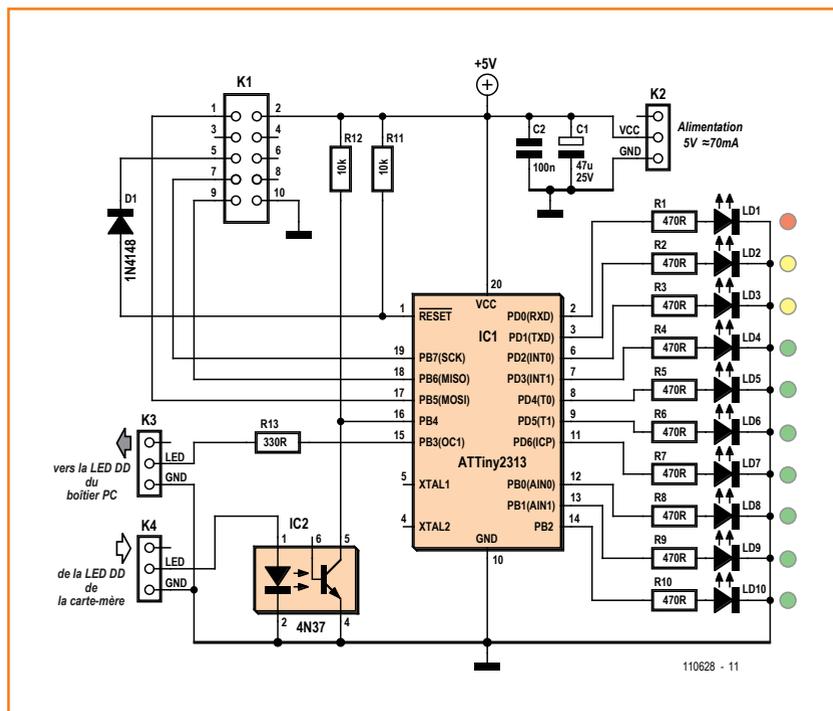


Caractéristiques

- Affichage sur 10 LED de l'activité du disque dur
- Affichage de l'activité du disque dur par intervalles de 10 %
- Immunité totale aux caprices de l'informatique (fonctionne sans programme PC)
- Module compact, à installer dans le boîtier d'un PC

est élégamment effectué par un optocoupleur ; ainsi, pas de problème d'adaptation de niveau. La LED du boîtier n'est pas mise au chômage : une sortie supplémentaire permet au microcontrôleur de la commander comme si elle était reliée directement à la carte-mère.

Le circuit IC1 pilote chacune des 10 LED via une simple résistance-talon de 470 Ω. Pour piloter encore plus de LED, il faut changer de microcontrôleur et adapter le programme. Dans ce cas veillez à ne dépasser ni le courant maximal autorisé pour chaque broche (ici 10 mA) ni le courant total admissible pour l'ensemble des broches (ici 60 mA). Utilisez des LED à faible courant si vous souhaitez un afficheur très lumineux (une forte luminosité peut toutefois être inconfortable). L'ensemble du circuit est alimenté par l'une des sources de tension 5 V disponibles (câble pour disques durs ou lecteur de disquettes). La barrette mâle à 10 contacts, typique de chez Atmel, sert à programmer le microcontrôleur.



Assemblage et installation

La platine conçue par l'auteur accueille les LED et l'électronique. Vous pourrez constater de visu en téléchargeant les fichiers de Karsten [1] que ses côtés composants (*PCB_top.jpg* [1]) et cuivre (*PCB_bot.jpg* [1]) restent aérés malgré la présence de composants CMS. L'implantation des composants ne présente pas de difficultés particulières. Karsten a prévu pour les résistances un boîtier 1206, facile à souder à la main.

Les LED se placent côté cuivre. N'exagérez pas l'espacement entre elles pour que le module garde un aspect compact. Percez dans le boîtier les trous qui accueilleront les LED. Le dessin à l'échelle 1:1 peut servir de gabarit de perçage. Servez-vous d'entretoises pour visser la platine. Vous pouvez aussi, comme l'auteur (voir *Prototype.jpg* en [1]), plier et souder les pattes des LED de façon à les monter à angle droit au bord de la platine, elle-même montée à angle droit derrière la façade.

Lors du flashage du micrologiciel, les bits de configuration doivent être positionnés pour que le générateur d'horloge interne soit utilisé avec une fréquence de 4 MHz. Sans ce paramétrage, le microcontrôleur tournerait plus lentement et la fréquence de l'actualisation en serait affectée. La capture d'écran de la **figure 2** montre la configuration des bits sous l'environnement de développement AVR-Studio 4.

Le micrologiciel a été écrit en C et compilé avec CodeVision. Le code est abondamment commenté, il sera facile à modifier. Code source C et fichier HEX peuvent être téléchargés gratuitement depuis la page de notre site associée à cet article [1]. Vous y trouverez également les dessins au format Eagle ainsi qu'une vidéo dans laquelle Karsten montre le fonctionnement de sa jauge lumineuse.

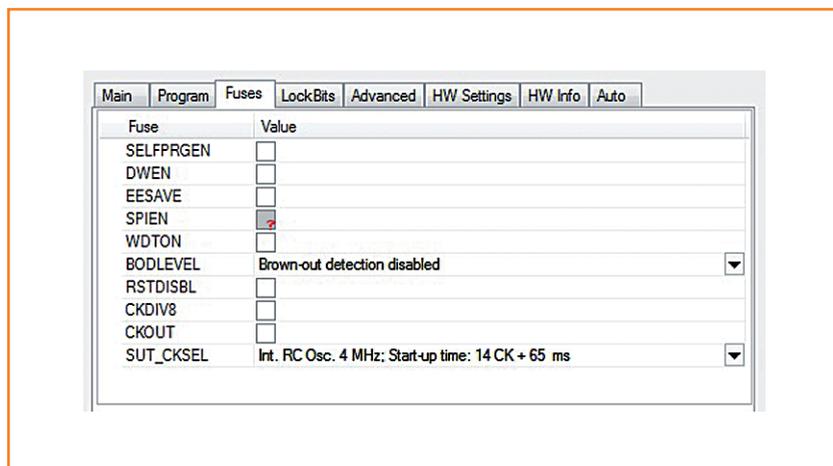
(110628)

Lien

[1] Vidéo, programme & dessins de l'auteur : www.elektor.fr/110628

Figure 1. Un schéma de principe limpide.

Figure 2. Configuration des bits sous AVR-Studio 4 pour une horloge cadencée à 4 MHz.



du code de pros

avec les diagrammes d'états-transitions

Peter Müller
(Allemagne)

À l'origine outils de conception pour le développement du matériel, les diagrammes d'états-transitions sont de plus en plus utilisés en génie logiciel. Ces schémas permettent en effet de construire des programmes robustes et peu sujets aux bogues.

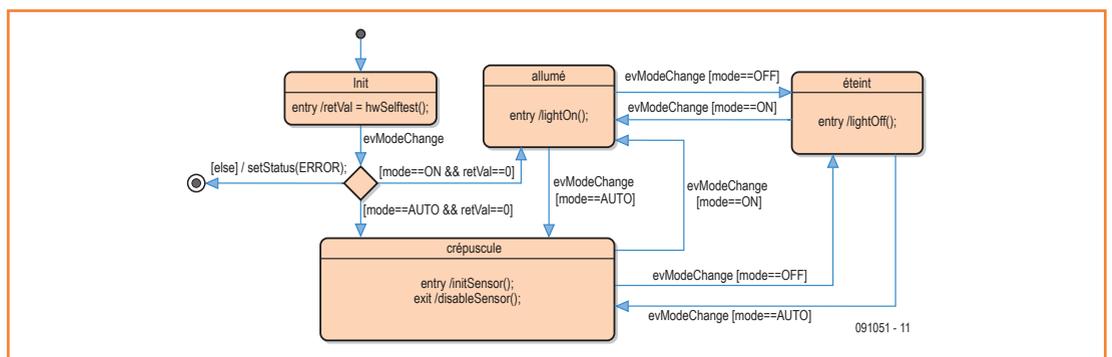


Figure 1.
Modèle simplifié d'un interrupteur crépusculaire.

De nombreux appareils sont programmés pour réagir à certains évènements internes ou extérieurs. Leur réponse n'est cependant pas toujours à sens unique, car elle peut dépendre de l'état en cours de l'appareil. Autrement dit l'historique d'un dispositif, c'est-à-dire l'ensemble des évènements passés qu'il a rencontrés, conditionne sa réponse. L'intérêt des diagrammes d'états-transitions est dès lors évident, puisque ces schémas décrivent d'un côté l'état d'un système, de l'autre les conditions qui auront pour effet un changement de son état.

Quelques exemples :

- Lorsqu'un usager appuie sur le bouton d'ouverture de porte d'un ascenseur, celui-ci ne réagit pas de la même façon selon qu'il est à l'arrêt ou entre deux étages.
- Un distributeur automatique de boissons attend qu'un choix ait été validé et que le montant correct ait été payé avant de délivrer la commande. La réponse est différente s'il n'a pas la boisson demandée ou pas assez de pièces pour rendre la monnaie.
- Un interrupteur crépusculaire automatique réagit à une présence en fonction de la lumi-

nosité et du réglage choisi (éclairage permanent, fête, etc.)

La liste pourrait être longue. Quoi qu'il en soit, il suffit de considérer quelques appareils comme des « automates à états finis » pour vite se convaincre de la puissance de la méthode.

Exemple

Prenons l'exemple d'un interrupteur crépusculaire automatique équipé d'un détecteur de mouvement, comme de nombreuses portes de maison en sont pourvues. Il réagit aux évènements extérieurs, p. ex. :

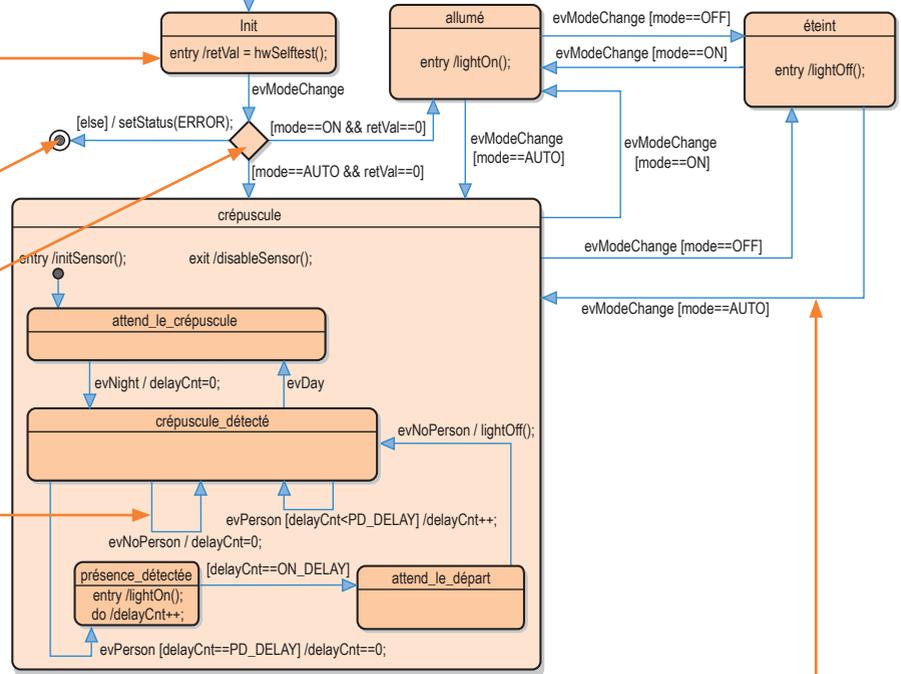
- choix du mode de fonctionnement (automatique, éclairage permanent, éteint) ;
- détection d'une présence ;
- passage jour/nuit, nuit/jour.

Il réagit cependant aussi aux évènements internes, p. ex. une certaine temporisation. La lumière s'allumera ou non en réponse à ces évènements. Certaines interactions sont impossibles à un moment donné : une présence ne déclenchera aucune réponse si l'interrupteur a détecté l'évènement « jour ». Il faudra pour cela attendre que la condition « obscurité » soit réalisée. Après

Facile à comprendre

Quelques points forts des diagrammes d'états-transitions : on se familiarise très vite avec eux, ils sont accessibles aux non-programmeurs, et ils forment un excellent support pour vérifier un programme ou discuter de sa conception avec un collègue ou client.

- État initial (init state) :** premier état actif pris par la machine après son initialisation.
- État :** un état décrit la situation ou condition dans laquelle se trouve un appareil ou composant logiciel. Un état peut posséder des sous-états.
- État final (final state) :** état qui ne traite plus aucun évènement.
- Point de décision :** un point de décision a une transition en entrée, et au moins deux en sortie. La transition sélectionnée dépend des conditions de garde.
- Transition interne :** transition qui n'entraîne pas de changement d'état ; *entry* et *exit* déclenchent des transitions internes.
- Transition :** une transition relie un état source et un état cible. Une transition peut être soumise à une condition de garde (*guard*) et/ou contenir une activité ; l'activité décrit l'action qui accompagne la transition.



sa mise sous tension, l'interrupteur crépusculaire est dans un état dit initial. Un diagramme d'états-transitions (en anglais *state chart* ou *state diagram*) décrit les états que peut prendre un système à partir de cet état initial. Un état est représenté par un rectangle aux angles arrondis. Une transition entre deux états est symbolisée par une flèche. La description textuelle au-dessus de la flèche précise l'évènement (*event*) qui déclenche la transition. Une condition de garde (*guard*) peut décider du passage ou non d'un état à un autre. La transition ne sera effective que si cette condition est vraie.

La **figure 1** montre le diagramme d'états-transitions simplifié d'un interrupteur crépusculaire. La flèche qui émane du petit cercle mène à l'état que prendra la machine après avoir été initialisée. L'appareil prend ici l'état *Init* après avoir été mis sous tension. L'état *Init* mène à un point de décision (représenté par un losange) qui peut conduire à trois états. Chaque choix possible est soumis à une condition de garde.

- Une action peut être exécutée lors d'une :
- entrée dans un état (*onentry*)

- sortie d'un état (*onExit*)
- poursuite d'un état (*do*).

Lors d'une transition, le code de sortie de l'état source puis le code d'entrée dans l'état cible sont exécutés s'il y a lieu. En entrant dans l'état *Init*, l'interrupteur effectue ainsi un autotest. S'il échoue, une erreur est signalée à l'utilisateur (branchement conditionnel *else* du point de décision). Le disque noir entouré d'un cercle symbolise l'état final. Aucune transition n'est possible à partir d'un état final, autrement dit un état final ignore tous les évènements.

Si l'autotest s'est terminé avec succès (*retVal == 0*), la machine prend l'état *crépuscule* ou *éteint*, selon la valeur de la variable *mode*.

Une transition peut aussi être liée à une action (donc à un certain code de programme) : dans l'exemple ci-dessus, la branche conditionnelle *else* du point de décision conduit à l'état final *ERROR*.

Affinement

Le diagramme d'état d'un automate fini peut être plat ou hiérarchisé. Celui de la figure 1 est plat, car aucun état ne comprend de sous-état. Si maintenant nous affinons le mode *crépuscule*,

Figure 2. Diagramme détaillé de l'interrupteur crépusculaire.

nous obtenons la machine hiérarchisée dont le diagramme d'état est reproduit sur la **figure 2**. Lorsqu'elle prend l'état *crépuscule*, la machine se branche sur l'état initial interne *attend_le_crépuscule*. Si l'évènement *evNight* est envoyé à la machine, elle prend l'état *crépuscule_défecté*. Si une présence a été détectée, la lumière reste allumée un certain temps, avant de s'éteindre à nouveau si plus aucun mouvement n'active le capteur.

Le diagramme d'état rend plus manifeste la logique du programme, structure logique qui aurait été autrement plus difficile à lire en parcourant du code. On voit par exemple que la lumière reste allumée pendant un certain temps, et qu'elle le reste encore jusqu'à ce qu'aucune présence ne soit plus détectée. Si cette condition n'avait pas été prévue, en tout état de cause la lumière devrait d'abord être éteinte. Lors de la transition *allumé à crépuscule*, la lumière est déjà allumée. Est-ce judicieux, ou bien manque-t-il encore une action ?

Exécution

Un automate fini réagit à des évènements. Le temps de réponse typique d'un système est de quelques millisecondes, mais peut aussi s'exprimer en secondes. Pour que la machine s'exécute correctement, on doit partir du principe qu'un évènement sera entièrement traité avant que le prochain ne se produise. Si rien ne le garantit, une file d'attente (*eventqueue*) peut servir de tampon.

Dans les diagrammes hiérarchisés, les évènements sont d'abord « soumis » à l'état actif le plus interne. Si l'évènement n'y est pas traité, il est remonté vers le niveau hiérarchique immédiatement supérieur, puis vers le suivant, et ainsi de suite jusqu'au plus haut. L'évènement *evModeChange* ne peut par exemple déclencher une transition de l'état *crépuscule* que s'il n'a pas déjà été « consommé » par un état plus interne (ce qui n'est pas le cas ici). L'idée générale étant que des états internes affinent un état de niveau supérieur, l'état le plus profond doit par conséquent pouvoir réagir le premier à un évènement.

Tableau d'états

Un diagramme d'états devient vite confus lorsque les états sont très nombreux. Les tableaux sont un moyen de représentation mieux adapté aux

situations complexes. Les états y sont rangés horizontalement et verticalement, l'intersection d'une ligne et d'une colonne contenant le déclencheur d'une transition.

La **figure 3** montre la table d'états équivalente au diagramme d'états de la figure 2. Même si la représentation n'est que partielle, la présence de cases vides est normale. Les transitions non définies sont par définition les évènements qui ont été écartés. Par exemple l'évènement *evPerson* n'est pas traité par l'état *attend_le_crépuscule*. Un tableau permet de passer en revue les points d'intersection de façon méthodique, de s'assurer que les évènements, actions et conditions de garde ont été choisis correctement, et face à une case vide de se demander si une transition n'a pas été oubliée.

Implantation

L'étape qui suit la conception et la vérification éventuelle d'un automate fini est bien sûr son implantation. Pour cela on utilise souvent une structure de contrôle de type *switch*.



Le document du lien [1] contient le listing d'un code qui implante la machine de la figure 2 avec une structure conditionnelle *switch*.

Implanter à la main un diagramme d'états complexe est toutefois laborieux. On oublie vite une transition, ou encore on place une action au mauvais endroit. La probabilité qu'une erreur se faufile dans le code augmente en particulier lorsque le diagramme est modifié après coup. Ce type d'erreur est souvent difficile à trouver.

Il existe heureusement des outils capables de générer automatiquement un code à partir d'un diagramme d'états. Les diagrammes d'états se prêtent généralement très bien à une conversion complète en code, un avantage appréciable puisque la cohérence entre l'automate et le code produit épargne un fastidieux travail de codage. Quelques règles sont cependant à respecter au moment de la conception si l'on veut que l'outil puisse produire un code exécutable. Quelques exemples :

- Les états doivent posséder des noms univoques.
- Les caractères qui composent le nom d'un état doivent être valides dans le langage du code produit (ex : pas d'espace en C, ni chiffre en début de nom).

	éteint	allumé	crépuscule	attend_le_crépuscule	crépuscule_détecté	présence_détectée	attend_le_départ	Init	FINAL_0
éteint		evModeChange [mode==OFF]	evModeChange [mode==OFF]						
allumé	evModeChange [mode==ON]		evModeChange [mode==ON]					evModeChange [mode==ON && retVal==0]	
crépuscule	evModeChange [mode==AUTO]	evModeChange [mode==AUTO]						evModeChange [mode==AUTO && retVal==0]	
attend_le_crépuscule					evDay				
crépuscule_détecté				evNight	evNoPerson evPerson [delayCnt<PD_DELAY]		evNoPerson		
présence_détectée					evPerson [delayCnt==PD_DELAY]				
attend_le_départ						[delayCnt==ON_DELAY]			
Init									
FINAL_0								evModeChange[else]	

- Les actions doivent contenir des appels de fonctions ou du code valides.
- Une condition de garde doit toujours renvoyer « vrai » ou « faux ».

Le générateur de code vérifie lui-même un certain nombre de ces règles. Il peut aussi procéder à d'autres validations de façon automatique, par exemple :

- Tous les états peuvent-ils être atteints, ou bien existe-t-il des voies sans issue ?
- Un état initial est-il défini partout là où il est nécessaire ?
- Existe-il un déclencheur pour chaque transition ?

Le suivi de ces règles permet au générateur de code de détecter rapidement et automatiquement de nombreuses incohérences. Ce type d'outil possède souvent d'autres fonctions d'aide au débogage et à la recherche d'erreurs.

Modélisation et débogage

Pouvoir exécuter sur ordinateur la représentation numérique d'un automate fini s'avère très utile. Les transitions d'état sont dans ce cas déclenchées par des événements modélisés ; on voit alors clairement quel code est exécuté, et si la machine se comporte comme attendu. Observer le comportement de l'automate en conditions réelles (dans l'appareil) est tout aussi utile. Pour cela on peut se servir d'une simple sortie texte qui affiche l'état en cours et les événements entrants.



Le document du lien [1] contient un exemple de fonction qui réalise un tel traçage. Un second exemple montre comment le petit robot mobile ASURO [2] a été doté d'une certaine intelligence.

Si cette introduction aux diagrammes d'état a déclenché en vous un état d'excitation, la transition vers un état de satisfaction sera déclenchée par un outil UML (*Unified Modeling Language* [3]). Mentionnons en particulier l'outil *open source* ArgoUML [4]. La version de démonstration Sinelabore [5] pourra quant à elle servir à vos premiers essais de production de code.

(091051 - version française : Hervé Moreau)

Liens

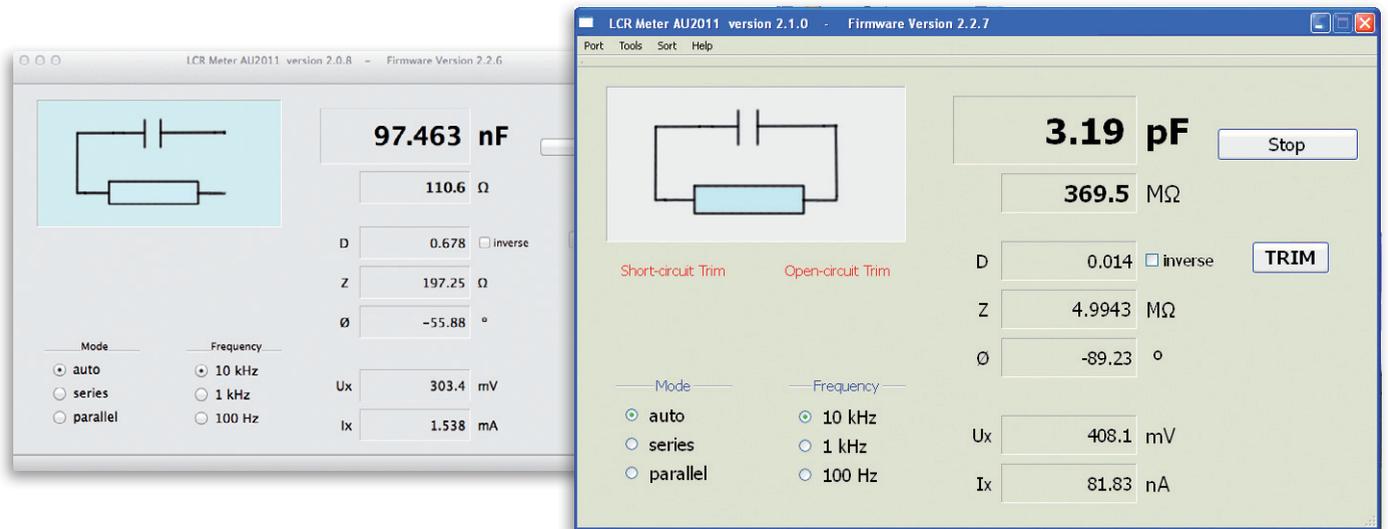
- [1] www.elektor-magazine.fr/091051
- [2] www.youtube.com/watch?v=pIpuR_LlwY4&feature=player_embedded
- [3] http://fr.wikipedia.org/wiki/Unified_Modeling_Language
- [4] <http://argouml.tigris.org>
- [5] www.sinelabore.com

Figure 3. Le diagramme d'état de la figure 2 représenté par tableau (vue partielle).

LCR-mètre 0,05 %

Le luxe de la précision accessible à tous

2^e partie



Jean-Jacques Aubry,
Ollioules

Dans ce second article, vous découvrirez :

- l'extension afficheur&clavier pour utiliser l'appareil sans PC
- les fonctions de l'appareil par ses menus et l'afficheur
- le logiciel, du moins certains de ses aspects les plus remarquables, car il est impossible d'en explorer ici toutes les arcanes (description, code source et exécutables intégralement disponibles sur notre site [1]).

Pour tirer profit de cette deuxième partie, il vaut mieux avoir lu la première et avoir sous les yeux le schéma de l'appareil.

Faute de place dans le premier article [1], nous n'avions pas encore présenté la deuxième carte, optionnelle, du LCR-mètre, qui en fait un appareil autonome. Elle n'est pourtant pas bien encombrante et nous en aurons vite fait le tour.

L'extension afficheur&clavier

Au début, ce LCR-mètre ne fonctionnait que connecté à un PC. Les avis sont partagés sur les restrictions que cela implique. Pour réconcilier les partisans de l'autonomie et ceux qui ne répugnent pas à dépendre d'un ordinateur, j'ai rajouté l'extension à la demande du labo d'Elektor. La tâche fut (relativement) aisée, car non seulement la version précédente de l'appareil,

alimentée sur le secteur, utilisait déjà un afficheur graphique de 128x64 pixels, mais d'autre part le code de gestion de cet afficheur et du mini-clavier associé était réutilisable à 90 %.

Le nombre réduit de lignes de port disponibles (5 au lieu de 13 dans la version antérieure) m'a contraint à abandonner la commande *parallèle* de l'afficheur au profit d'une commande *série*. De plus, il fallait pouvoir détecter automatiquement la présence de l'extension ainsi que le type d'alimentation (par le port USB du PC ou par une alimentation externe).

Le schéma de l'extension est modeste (**fig. 1**). L'afficheur graphique à cristaux liquides U1 de 128x64 pixels (*Displaytech*) est géré, en interne,

par un contrôleur *ST7565R* de *Sitronix*. Les condensateurs C1 à C9 servent pour la conversion de tension de la commande du LCD. L'éclairage arrière intégré est commandé par la carte principale par l'intermédiaire du transistor Q1 : il ne sera activé qu'en mode autonome. L'état de la broche 6 de J16 (LCD_RES) indique la présence (0) ou l'absence (1) de l'extension.

La LED D5, commandée par le transistor Q2 n'est qu'un clone de D6 du circuit principal : elle flashe à la fin de chaque séquence de mesure.

Le mini-clavier est constitué des cinq boutons K1 à K5 et des diodes de matricage D1 à D4. Trois lignes de port offrent 8 codes différents, dont 7 codes utiles si on exclut le code 111 (pas de bouton actionné). C'est peu, mais le logiciel du LCR-mètre fera la distinction entre appui bref et appui prolongé, ce qui augmentera les combinaisons possibles.

La liaison avec le circuit principal (par J16) est faite par un câble en nappe muni d'un connecteur femelle à cette extrémité, mais solidaire, côté extension, de J1, qui est un connecteur de transition soudé sur le circuit, pour des raisons d'encombrement vertical.

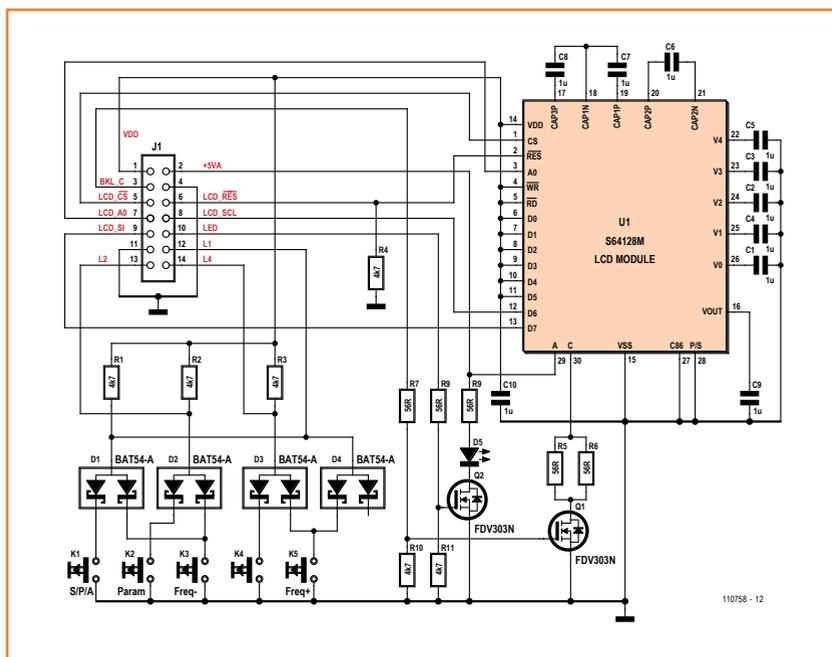
Les fonctions des poussoirs, sur lesquelles nous reviendrons bientôt en détail, sont :

	Appui bref	Appui prolongé
K1	S/P/A	Menus
K2	Param.	Q-D
K3	Freq-	Sorting
K4		RT
K5	Freq+	Trim

Cette fois nous avons fait le tour de la description du matériel. Il est temps de découvrir le fonctionnement du LCR-mètre, associé ou pas à un ordinateur. Les procédures de mise en route et d'alignement font l'objet de deux documents complémentaires, richement illustrés, téléchargeables depuis le site d'Elektor [3]. Nous n'entrerons pas ici dans le détail de tous les réglages ni dans tous les menus, mais en dirons assez pour donner au lecteur l'envie de franchir le cap et de se lancer dans la réalisation de cet appareil hors pair.

Mode PC

Si la tête de mesure, dont le schéma a été publié par Elektor le mois dernier, est reliée à un ordinateur par une liaison USB, le mode sélectionné par défaut est le mode *PC*. L'interface du pro-



gramme *AU2011* [1], installé sur le PC hôte, est en anglais par défaut, mais prend en compte des fichiers de traduction.

Comme l'appareil est équipé du circuit de communication USB-UART de FTDI (FT232R), il faut en installer le pilote. Cette opération est décrite dans le document *LCR-mètre | Mise en route* avec toute la procédure de calibration, soigneusement détaillée.

Lancement du programme et menus

Connecter le LCR-mètre au PC, lancer le programme utilisateur sur le PC et mettre sous tension le pont de mesure, puis suivre les instructions de la notice de mise en route. Se suivent ensuite les menus **Préférences, Outils, Trier (fig. 2)** dans le détail desquels nous n'entrerons pas ici.

Figure 1a. La carte d'extension comporte cinq boutons matricés et un afficheur graphique.

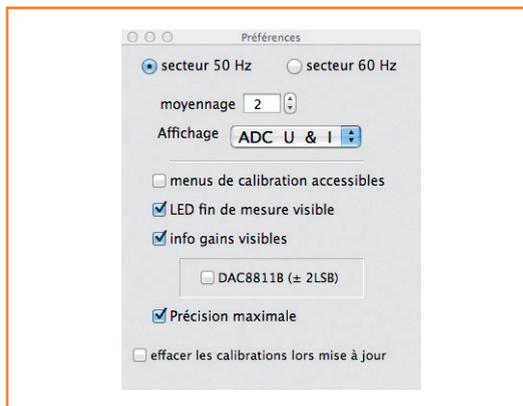


Figure 2a. Il faut commencer par certains choix incontournables dans les préférences.

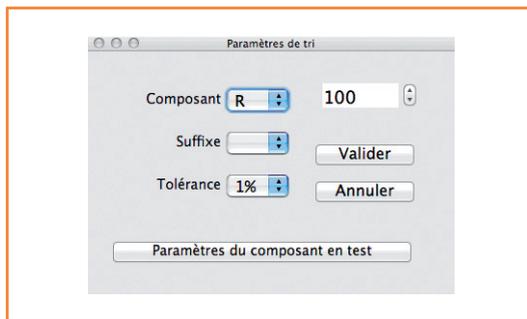


Figure 2b.
La fonction de tri demande un paramétrage préalable.

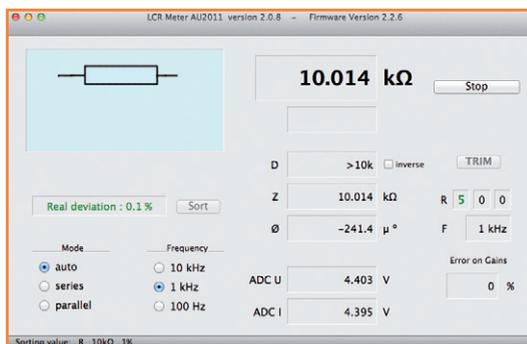


Figure 3.
La fenêtre principale, ici pendant un tri sur la résistance (valeur de consigne affichée en bas à gauche).

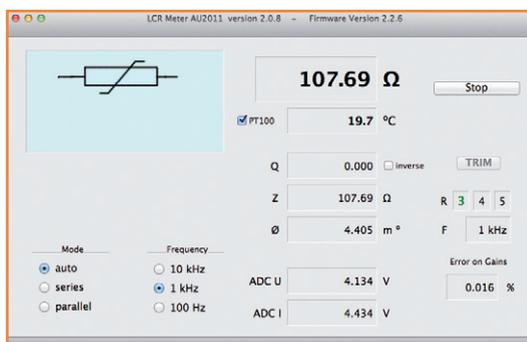


Figure 4.
Une sonde de platine permet une mesure précise de la température dans une gamme de -80 °C à +600 °C.

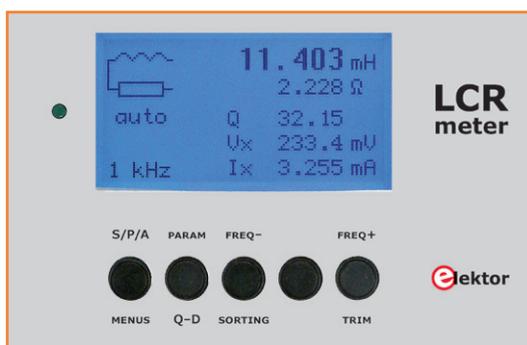


Figure 5.
Les boutons et l'affichage en mode autonome.

Intéressons-nous à l'**Interface utilisateur** (fig. 3). La fenêtre principale du programme affiche les résultats de la mesure, et quelques boutons permettent de modifier certains paramètres, ou de lancer certaines actions. En haut à gauche, un pictogramme affiche le circuit équivalent du composant en test. Quand le pointeur de la souris passe sur un champ d'affichage, une bulle

d'information apparaît sur ce qui est affiché dans ce champ.

À la première mise en route, le bouton **Ouvrir le port** lance la procédure de connexion au port série défini (et mémorisé) dans les *Préférences*. Une fois la connexion établie, il devient inactif. Les fois suivantes, la procédure sera automatique et ce bouton ne sera pas visible. Comme son nom l'indique, le bouton **Arrêter** ou **Démarrer** arrête ou démarre les mesures.

Bouton TRIM

Lance la compensation *OUVERT – FERMÉ* (*OPEN – SHORT*).

La compensation, qui consiste à mesurer les paramètres propres au dispositif de mesure (cordons, pince), est effectuée pour chaque fréquence. En cas d'erreur, la valeur n'est pas retenue. Tant que la compensation n'a pas été faite, un symbole rouge le signale.

Le bouton **Trier** lance la procédure de tri. La valeur de référence apparaît à gauche dans le bandeau inférieur de la fenêtre principale. L'écart réel est affiché en vert si le composant testé est dans les tolérances, en rouge s'il est en dehors. Les boutons **Mode** permettent la sélection du mode de représentation du composant en test : **auto** : la sélection entre série ou parallèle est effectuée automatiquement en fonction de l'impédance du composant en test.

série : force la représentation série.

parallèle : force la représentation parallèle.

La sélection de la fréquence de test est faite avec les boutons **Fréquence**. La plus basse est égale à deux fois celle du secteur, choisie dans les *Préférences*. Le premier champ de **Gain** indique la gamme de mesure :

- en vert pour les gammes 3 à 6 où l'erreur due à l'amplificateur principal est nulle.
- en magenta pour les gammes 2 et 7 (erreur due à l'amplificateur principal ±0,02 %)
- en rouge pour les gammes 1 et 8 (erreur due à l'amplificateur principal ±0,04 %)

Les deux autres champs donnent les valeurs, entre 0 et F, des paliers du CNA multiplicateur pour les mesures de la *tension* et du *courant*.

Si la case **Max** est visible (ce n'est pas le cas ici), la valeur de l'erreur sur le gain est calculée avec les valeurs typiques ou avec la valeur Max (du PGA103) selon que celle-ci est cochée ou non.

Les menus **Réglages** permettent d'effectuer des réglages internes : décalages, calibrations, etc. Les trois premiers ne nécessitent pas d'ouvrir le

Qu'y a-t-il de plus jouissif que de réaliser ses propres outils ?

couvercle. On se reportera au document *LCR-mètre | Mise en route*.

Mesure d'une sonde de platine PT100

On trouve beaucoup d'appareils qui proposent un affichage de la température, mais avec une précision de quelques degrés ! Il me paraissait intéressant de disposer d'une vraie référence. Une sonde de platine permet donc une mesure de la température dans une gamme de $-80\text{ }^{\circ}\text{C}$ à $+600\text{ }^{\circ}\text{C}$ et avec une précision de $\pm 0,3\text{ }^{\circ}\text{C}$ (classe B) ou $\pm 0,1\text{ }^{\circ}\text{C}$ (classe A).

Quand la valeur d'une résistance est comprise entre $70\ \Omega$ et $300\ \Omega$, que la fréquence est $\leq 1\text{ kHz}$ et que Q , le facteur de qualité, est pratiquement nul, alors une case marquée **PT100** apparaît à gauche du champ d'affichage du paramètre secondaire (**fig. 4**)

En la cochant, on fait apparaître la valeur de la température d'une sonde de platine PT100 ayant cette résistance. Le symbole devient celui d'une thermistance. Dans la plage de mesure utilisée ($-75\text{ }^{\circ}\text{C}$ à $+557\text{ }^{\circ}\text{C}$) la précision de la conversion résistance/température est $< 0,05\text{ }^{\circ}\text{C}$.

Mode autonome

Équipé de l'extension *afficheur & clavier*, notre LCR-mètre devient autonome. Plus besoin de PC (sauf pour faire les *réglages préliminaires*, impossibles en mode autonome). Dans ce mode, tous les textes affichés sont en anglais.

Les cinq boutons (**fig. 5**) permettent de sélectionner les fonctions de l'appareil et de configurer certains paramètres. Les touches ont différentes fonctions selon que l'on appuie brièvement ou plus de deux secondes (signalé par la LED verte). Les fonctions principales sont en noir (en haut sur mon proto), les fonctions secondaires (appui long) en bleu.

Dans les menus, la fonction (variable) des touches est indiquée en bas de l'afficheur à cristaux liquides.

Fonctions principales

S/P/A : Choix du modèle entre les circuits équivalents série ou parallèle pour le calcul des paramètres, ou le mode modèle automatique.

Param. : Affichage au choix :

- du module de l'impédance $|Z|$ et de l'angle de phase Φ .



Figure 6 – Fonction de tri (Sort)

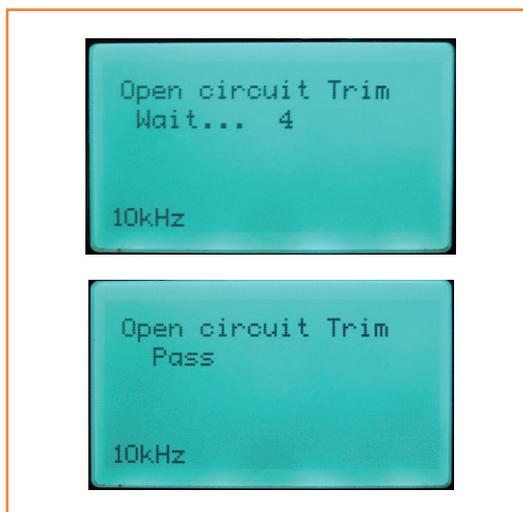


Figure 7 – Fonction de compensation (Trim)

- de la résistance série équivalente R_s et de la réactance série équivalente X_s .
- de la tension efficace V_x et du courant efficace I_x du composant.

Freq- & Freq+ : Choix de la fréquence de mesure.

Fonctions secondaires

Menus : Entrez dans le mode **Menu** en appuyant longtemps sur cette touche.

Q_D : Un appui long sur cette touche inverse l'affichage automatique de **Qualité** → **Dissipation** ou **Dissipation** → **Qualité**

Sorting : Il faut avoir auparavant défini les paramètres du tri par le menu *Sorting Parameters*. Un appui long sur cette touche permet d'activer ou de désactiver le mode **Tri**.

La valeur du composant en tri s'affiche avec son symbole (**fig. 6**).

Les paramètres de tri sont en affichage inversé. Un appui sur la touche [GO] lance la comparaison avec affichage transitoire de la tolérance réelle et du résultat (OK ou BAD !)

Trim : Un appui long sur cette touche permet de lancer la compensation *OUVERT - FERMÉ (OPEN - SHORT)* (**fig. 7**). Les compensations sont effectuées pour chaque fréquence.

Si $|Z|$ est $< 10 \Omega$, c'est la calibration *FERMÉ* qui est effectuée.

Si $|Z|$ est $> 100 \text{ k}\Omega$, c'est la calibration *OUVERT* qui est effectuée.

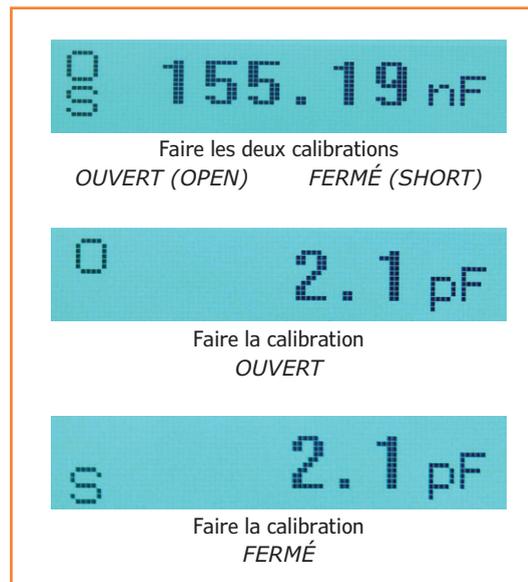


Figure 8.
Calibrations défaillantes



Figure 9.
Le choix des menus

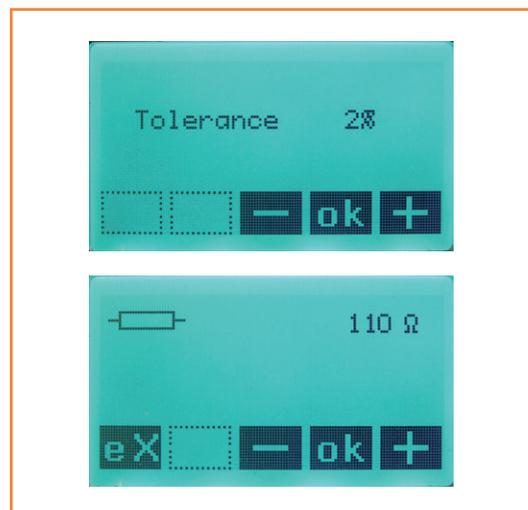


Figure 10.
Tolérance de tri et paramètre de tri principal

Un symbole clignotant apparaît devant la valeur du paramètre principal quand l'une ou les deux calibrations ne sont pas valides. Refaire la (les) calibration(s) défaillante(s) (**fig. 8**).

Touche sans nom : la fonction de ce bouton est expliquée dans le document *LCR-mètre | Mode d'emploi* [3].

Menus (fig. 9)

La navigation s'effectue à l'aide des touches \uparrow et \downarrow . Le symbole $>$ indique le menu qui sera sélectionné par [ok].

L'appui sur la touche [eX] (*exit*) permet de sortir du mode *menu*.

Sorting Parameters

Ce menu permet de choisir les paramètres (tolérance et valeur) utilisés lors du tri de composants. La valeur proposée sera celle du paramètre principal du composant de référence connecté **avant** d'entrer dans le mode menu.

La première étape permet de choisir la tolérance par les touches [-] et [+] puis validation par [ok]. (**fig. 10**)

L'étape suivante, en partant du **paramètre principal** du composant connecté, permet d'ajuster la valeur qui sera retenue comme valeur de tri.

Touches [-] et [+] puis validation par [ok]. (**fig. 10**)

Dans le mode d'emploi en ligne, vous découvrirez d'autres fonctions que nous ne détaillerons pas ici : le moyennage (*Averaging*), l'affichage de la gamme en cours (*Display Range*), le réglage du contraste (*Adjust Contrast*), l'éclairage (*Back Light*), la fréquence du réseau (*Line freq. 60/50Hz*). Voir aussi ci-dessus dans la section **Mode PC** le paragraphe consacré à la **Mesure d'une sonde de platine PT100**, fonction également disponible en mode **autonome**.

Les programmes

La performance globale du LCR-mètre (précision des mesures, facilité d'utilisation) est tributaire de la qualité des trois programmes utilisés :

le programme d'amorçage, ou *bootloader*, qui prend le contrôle lors de la mise sous tension du pont de mesure.

le logiciel résident ou *firmware*, la partie la plus importante du programme interne qui réalise tout le travail d'acquisition des mesures, de calcul de la DFT, etc. C'est aussi lui qui pilote l'afficheur en mode *autonome*.

le programme externe *AU2011* s'exécutant sur l'ordinateur connecté, et qui, en mode « PC », permet d'afficher les résultats, de lancer des com-

mandes, etc. Il dialogue avec le *bootloader* puis le *firmware* par envoi et réception de messages sous forme de chaînes de caractères prédéfinies.

Bootloader

Le programme d'amorçage, lancé automatiquement à la mise sous tension, réside en permanence en mémoire à partir de l'adresse 0x0000. Sa fonction principale est d'effectuer la mise à jour du programme principal (*firmware*) :

- si le cavalier J17 est présent (mise à jour

inconditionnelle, p. ex. dans le cas où le *firmware* est planté) ;

- lors d'un *software reset*, quand l'utilisateur demande cette mise à jour ;
- si le contrôle de l'intégrité du code du programme principal en EEPROM est négatif.

Cette mise à jour ne peut s'effectuer que **si l'appareil est en mode PC**, par le programme *AU2011*.

La première étape est l'effacement de la mémoire de programme (tous les octets prennent la valeur

Listing 1

```
void UART0_ISR(void) interrupt INTERRUPT_UART0
{
    char SFRPAGE_SAVE = SFRPAGE;
    SFRPAGE = UART0_PAGE;
    if (RI0 == 1) // if receive flag set, put the byte in UART_InputBuffer
    {
        RI0 = 0;
        Byte = SBUF0; // Read a character from UART0
        if ((Byte == '\0') || (Byte == '\n') || (Byte == '\r'))
        {

            RX_Ready = 1; // Reception complete
        }
        else
        if (UART_InputBuffer_Size < UART_IN_BUFFERSIZE - 1) // room needed for string terminal '\0'
        {

            UART_InputBuffer[UART_InputBuffer_Size] = Byte;

            UART_InputBuffer_Size++; // Update array's size
        }
        if (TI0 == 1)
        // if transmit flag set, send UART_OutputBuffer datas
        {
            TI0 = 0;
            if (UART_OutputBuffer_Size > 0) // If buffer not empty
            {

                Byte = UART_OutputBuffer[UART_Output_First];

                SBUF0 = Byte; // Transmit to UART0

                UART_Output_First++; // Update counter

                UART_OutputBuffer_Size--; // Decrease array size
            }
            else
            {

                UART_Output_First = 0;

                TX_Ready = 1; // Transmission complete
            }
        }
        SFRPAGE = SFRPAGE_SAVE; // restore SFRPAGE
    }
}
```

0xFF), puis vient la réception du fichier au format HEX d'Intel via la liaison USB-UART. Après vérification que l'opération s'est correctement déroulée, le nouveau *firmware* est lancé à l'aide d'un *pointeur de fonction* sur l'adresse de début du code du *firmware* :

```
((void (code *) (void)) PROG_BEGIN_FLASH_ADDR) () // -> jump to application code
```

La partie mise à jour du *firmware* est directement issue de la note d'application AN112 de *Silicon Labs*, avec cependant une modification nécessaire lors de la somme de vérification (*checksum*) de la mémoire occupée par le programme nouvellement installé. En effet, des *trous* (de valeur 0xFF) pouvant exister dans l'espace mémoire nouvellement écrit, cela peut entraîner une différence entre la taille réelle du programme et le calcul effectué par *adresse_fin - adresse_début* d'écriture !

Pour les autres tâches réalisées par le *bootloader*,

il faut se reporter au code source complet que nous vous offrons de télécharger depuis notre site [1].

Firmware

C'est le cœur (et la tête !) de l'appareil. Il réside en mémoire à partir de l'adresse 0x2000. Il effectue les mesures, répond aux commandes de l'utilisateur, affiche (mode *autonome*) ou envoie les résultats par la liaison USB (mode *PC*).

Seules quelques petites parties du code sont décrites ici, le code source complet et généreusement commenté du *firmware* se trouve sur notre site [1].

Dans le mode *PC*, le dialogue avec le programme *AU2011* s'effectue par échange de messages. Les chaînes de caractères reçues ou envoyées sont traitées par la routine d'interruption de l'UART0. Une particularité des plus intéressantes du μ C choisi est le stockage directement en XRAM

Listing 2

```
void TIMER2_ISR(void) interrupt INTERRUPT_TIMER2
{
    char SFRPAGE_SAVE = SFRPAGE;
    SFRPAGE = TMR2_PAGE;
    TF2 = 0; // Immediately reset Interrupt flag
    if (TMR2CF & 0x04) // if state of the output = 1
    {
        ET2 = 0; // Disable Timer2 Interrupt
        SFRPAGE = DMA0_PAGE; // Switch to DMA0 Page
        DMA0EN = 1; // Begin Executing DMA Ops (which will enable ADC0)
    }
    SFRPAGE = SFRPAGE_SAVE;
    // restore SFRPAGE
}

void DMA0_Acquire_Samples (void)
{
    char SFRPAGE_SAVE = SFRPAGE;
    // --- reset TF2 flag just prior enabling Timer2 interrupt
    SFRPAGE = TMR2_PAGE;
    TF2 = 0;
    ET2 = 1; // Enable Timer2 Interrupt
    SFRPAGE = DMA0_PAGE; // Switch to DMA0 Page
    // --- Timer2 interrupt enable DMA-ADC operation
    while (!DMA0INT); // wait for DMA operations are complete
    DMA0EN = 0; // Stop Executing DMA Ops
    DMA0INT = 0; // Clear DMA0INT bit
    SFRPAGE = SFRPAGE_SAVE; // restore SFRPAGE
}
```

Mesures comparatives avec un pont de mesure professionnel

Mesurer des inductances est une science en soi. Dans les feuilles de caractéristiques des selfs, on indique souvent le courant et la fréquence pour lesquels la valeur d'inductance est spécifiée. Si vous mesurez par exemple à 1 kHz au lieu des 10 kHz indiqués par le fabricant et avec un courant de mesure différent, vous obtiendrez une valeur d'inductance sensiblement différente. Ce qui n'est pas nécessairement catastrophique, puisqu'avec les selfs la tolérance est de toute manière de +/- 20 %. Au labo d'Elektor, j'ai comparé le comportement de notre nouveau LCR-mètre conçu par Jean-Jacques Aubry à celui d'un très gros appareil de mesure de labo : le *Hameg Programmable LCR Bridge 8118*. Le courant de mesure sur cet appareil de référence (à plus de 1500 €) est réglable ; lors d'une première mesure, son intensité était plus de dix fois supérieure à celle de notre petit appareil. Selon le matériau de l'inductance, cela peut

conduire à des différences considérables. Mais nous sommes restés largement dans la plage de +/-20 %. Une première mesure effectuée sur le *Hameg* pour une self de 100 µH/5 A donne 108,7 µH. En réduisant le courant de mesure et après avoir procédé par acquit de conscience à une nouvelle calibration, le même appareil indique alors 97,6 µH. Et maintenant vous vous demandez bien sûr combien pendant ce temps-là affiche le nouveau LCR-mètre d'Elektor : 96,8 µH. Pas mal, non ?...

Ton Giesberts, labo d'Elektor



(**D**irect **M**emory **A**ccess) des données issues des ADC0 et ADC1. L'interface DMA est programmable, entre autres, pour écrire les données à partir d'une adresse en XRAM et pour un certain nombre d'acquisitions de ces données. Cette programmation est effectuée une fois pour toutes, par quelques instructions au niveau de certains des registres. Il n'y a plus ensuite qu'à lancer la commande de départ et attendre le signal de fin : tout le travail d'acquisition des données et leur transfert en mémoire sont automatiques et superbement optimisés, ce qui permet d'atteindre une cadence d'acquisition de 1 méga-acquisitions par seconde (données sur 16 bits, soit deux mots de 8 bits) avec une horloge à 24 MHz !

Ce processus est ordonnancé par les interruptions demandées par :

Timer2 (fréquence du signal sinusoïdal) -> démarrage d'une séquence de N acquisitions par positionnement à 1 du drapeau *DMA0ENable*. N = nombre échantillons par cycle de *Timer2* x nombre de cycles de *Timer2*.

Timer3 (fréquence de *Timer2* x nombre échantillons par cycle de *Timer2* / 2) -> démarrage automatique d'une acquisition, le coefficient 2 étant dû au fait qu'il y a deux interruptions par cycle du *Timer3*.

La fin de séquence est signalée par le passage à 1 du drapeau *DMA0INT*. Il ne reste plus qu'à lire la XRAM et exploiter les données.

Nous avons vu dans le premier article qu'il était nécessaire de compenser le décalage des ampli-

ficateurs opérationnels U6 (*input_offset*) et U1 (*sinus_offset*). Comme nous disposons de deux DAC à 12 bits, ce réglage peut être automatisé. Il faut simplement débrancher les câbles de mesure pour ne pas perturber le processus par des signaux parasites (secteur entre autres). Comme tous les autres réglages de l'appareil, ceux-ci ne peuvent s'effectuer qu'en **mode PC** par le programme *AU2011*. Ils sont décrits dans le document *LCR-mètre | Mode d'emploi* téléchargeable gratuitement et dont la lecture est hautement recommandée [3].

Programme AU2011

Le logiciel *AU2011* tourne sur un ordinateur personnel (PC). Il est écrit en C++ et utilise les bibliothèques Qt et *qserialdevice/AbstractSerial* afin d'obtenir un exécutable pour PC sous *Windows*, *Mac OSX* ou *Linux* selon la cible de compilation. Le code source et des exécutables sont disponibles sur notre site [1].

Si vous voulez modifier et/ou compiler le code source, l'IDE à utiliser est *Qt Creator*. Des options de compilation conditionnelles (*#ifdef*) dans le code source permettent de gérer directement les différentes cibles. L'interface utilisateur (IU) est très légèrement différente selon la cible de compilation à cause, entre autres, des différences dans les polices de caractères ; aussi chaque cible a son propre fichier *AU2011_mainwindow.ui*. Le noyau de l'interface utilisateur est en anglais, mais il est conçu pour pouvoir utiliser des fichiers

Un maître atout du µC choisi est le stockage directement en XRAM (Direct Memory Access) des données issues des ADC0 et ADC1

de traduction. La version française (*AU2011_fr.qm*) existe déjà. Placé dans le même répertoire que le programme *AU2011*, accompagné du fichier *qt_fr.qm* (fichier de traduction propre à Qt qui se trouve, avec ceux d'autres langues, dans le répertoire *QtSDK/Desktop/Qt/4.8.0/gcc/translations*), il permet d'avoir tous les messages et l'IU en français **pour un système français**. Pour obtenir la même chose pour une autre langue **locale**, il faut :

- soit dans le fichier *AU2011_Projet.pro*, modifier la directive `TRANSLATIONS += AU2011_fr.ts` en *AU2011_xx.ts* où *xx* est le symbole de la langue locale (*de*, *da*, *cs* etc.) et recompiler pour produire ce nouveau fichier ;
- soit dupliquer le fichier *AU2011_fr.ts*, et renommer la copie en *AU2011_xx.ts*.

Ensuite il faut utiliser le programme *Linguist* pour effectuer la traduction de toutes les chaînes et enfin lancer la commande *lrelease* pour produire le fichier *AU2011_xx.qm*.

Reportez-vous à la documentation de Qt *Linguist* pour plus d'informations [2].

Progressivement de nouveaux fichiers de traduction seront proposés par nos lecteurs !

Rappelons que le dialogue entre le logiciel *AU2011* et le LCR-mètre s'effectue à travers la liaison USB, par émulation d'une liaison série grâce au pilote proposé par FTDI (115.200 bauds, 8 bits, pas de bit de parité, 1 bit de fin, sans contrôle de flux). Les commandes envoyées à l'appareil ou reçues par lui sont des chaînes de caractères définies dans le fichier en-tête *AU2011_mainwindow.h* [1]. Après l'ouverture du *port* de communication et l'envoi de la commande de synchronisation avec le *bootloader*, le logiciel attend de recevoir la première chaîne de caractères contenant le numéro de version du *firmware*, information qu'il affiche dans le titre de la fenêtre. Puis ce sera la chaîne formée avec les paramètres d'initialisation :

- fréquence du secteur 50 ou 60 Hz (**L5** ou **L6**)
- fréquence de test (**F1** ou **F2** ou **F3**) pour 100/120 Hz ou 1 kHz ou 10 kHz
- *Trim Short* effectué ou non (**S1** ou **S0**)
- *Trim Open* effectué ou non (**O1** ou **O0**)
- valeur du moyennage (**A1** à **A9**)

Le *firmware* entre alors dans sa boucle d'événement et surveille l'arrivée d'une commande de

l'UART (routine d'interruption). Si la demande d'effectuer les mesures a été reçue, les paramètres résultants sont envoyés au logiciel sous forme d'une longue chaîne de caractères formée avec les valeurs de :

« **Rs Xs Freq ranges Vpp Ipp ADC_Vpp ADC_Ipp** »

plus le caractère 'C' ou 'Z' selon que le composant est capacitif ou non.

À la réception de cette chaîne, lue par la routine d'interruption *slotRead()*, la fonction *ParseInputString()* la décompose et les fonctions d'affichage *Display_xxx()* remplissent les champs concernés.

Une routine importante parmi d'autres, *Convert_Value_to_String()*, est celle qui convertit une valeur numérique flottante en une chaîne de caractères avec mantisse et exposant sous forme de suffixe normalisé, par exemple **12,05 nF** pour un condensateur de valeur $1,205 \cdot 10^{-8}$ (en farads).

Ça sent la résine

Dans le prochain numéro, nous vous présenterons enfin les deux cartes, qui seront disponibles sous forme de modules testés, prêts à l'emploi. Pour nettement moins de 200 €, vous pourrez faire l'acquisition d'une merveille de précision qui ne quittera plus votre plan de travail. Car il n'y a rien de plus jouissif que de réaliser ses propres outils ! D'ici là, nous vous vous invitons à consulter la documentation en ligne qui vous donnera une vue à la fois générale et très détaillée de notre de pont de mesure automatique d'impédance.

(130022)

[1] Logiciel téléchargeable (bootloader, firmware et programme principal) :

www.elektor-magazine.fr/110758

[2] Qt Linguist

<http://goo.gl/fIYQh>

[ou]

<http://qt-project.org/doc/qt-4.8/linguist-translators.html>

[3] Documentation en ligne :

[LCR-mètre | Mise en route](#)

[LCR-mètre | Mode d'emploi](#)

www.elektor-magazine.fr/110758

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

open data : données portes ouvertes

Tessel Renzenbrink
(Elektor TTF)

Un groupe de *hackers*, de programmeurs, chercheurs et décideurs s'est réuni en vue de bidouiller les données de la Chambre basse hollandaise, à l'invitation de la présidente sortante. Le but de cette réunion était de préparer l'ouverture de la base de données parlementaire Parlis. Des données ouvertes (*open data*) sont par définition accessibles et utilisables sans restriction, ni commerciale, ni de droits d'auteur et proposées dans un format lisible par une machine.

Mieke van Heesewijk et Josien Pieterse ont fondé et dirigent une plateforme pour l'innovation démocratique, *Netwerk Democratie* (c'est-à-dire *réseau démocratie*). La fondation a organisé l'événement *Apps voor Democratie* [2] en collaboration avec *Hack de Overheid* (c'est-à-dire *hacker les pouvoirs publics*) et *Open State*. Mark Bastiaans est chercheur auprès de l'organisation néerlandaise pour la recherche scientifique appliquée. (TNO [3]). Avec quelques collègues, il a créé un programme autour des données parlementaires. Nous avons discuté des données ouvertes et de leur possible influence sur la démocratie du XXI^e s.

Dans cet article, le terme *hack* est à prendre au sens premier de « bidouillage ingénieux pour extraire le meilleur d'un système », une définition aux antipodes de la piraterie informatique à laquelle il a été associé.

Aussi vieux que l'internet

Tessel : *Quel âge ont les données ouvertes ?*

Mieke : Les hyperliens sont aussi « vieux » que l'internet. Mais à notre époque, ce sont les données ouvertes dans la démocratie qui retiennent le plus l'attention. Certains militent en faveur de leur éclosion, qu'ils soient informaticiens, ingénieurs ou activistes. Aujourd'hui, on peut se réjouir de voir les gouvernants et les sociétés participer à cet engouement croissant.

Josien : Il s'agit de données publiques, donc de données issues ou à destination de la sphère publique. L'idée, c'est qu'elles appartiennent à tous, puisque chacun a contribué à leur récolte. Ces données sont réputées ouvertes dès qu'un gouvernement ou une société les expose au grand jour, sans restriction d'usage. Le plus excitant

avec ces données, c'est qu'il est possible de les combiner pour apprendre, comprendre ou découvrir de nouvelles connexions par recoupement.

On ira tous au parlement

Josien : L'ouverture de la base de données parlementaire Parlis est un bel exemple de ce mécanisme. Elle contient des informations qui intéressent les citoyens concernés par les processus démocratiques en jeu, tels que le comportement électoral des partis, les motions et les questions parlementaires. Ces informations étaient déjà disponibles en PDF, un format difficile à exploiter tel quel par un logiciel sans une infrastructure adaptée, qui restait donc à créer.

Gerdi Verbeet, la présidente de la Chambre basse, a organisé une réunion entre *Netwerk Democratie* et d'autres associations, baptisée « Applications pour la démocratie » qui s'est tenue le 8 septembre 2012, dans l'enceinte du parlement. L'événement consistait en un *hackathon* entre programmeurs pour mettre au point des applications destinées à rendre accessibles les données de Parlis. Ces sessions étaient agrémentées d'ateliers sur les données ouvertes.

Mieke : Organiser un *hackathon* dans un parlement est une première mondiale. Observer des *hackers* en train de programmer à l'intérieur de la Chambre basse valait le détour. Au-delà de l'aspect symbolique, c'était une réelle démonstration d'ouverture. Les *hackers* sont vraiment très mal considérés aux Pays-Bas et il a fallu quelque peu jongler avec les angoisses de certains partis politiques, quitte à les tenir à l'écart pendant que les *hackers* éthiques travaillaient. Leur contribution revêt une importance capitale pour l'évolution de

la démocratie, vers davantage de transparence et de sécurité. Le *hackathon* de la Chambre basse marquait dès lors un tournant historique.

Hackathon

Mark : Avec quelques ingénieurs, nous avons cherché un moyen d'extraire les infos de la base Parlis et le *hackathon* nous a permis de rédiger un logiciel à cet usage. Notre outil explore diverses combinaisons entre les parlementaires de la Chambre basse, en fonction de critères tels que l'expérience ou le nombre de motions présentées et adoptées. Il résume en une seule page les activités parlementaires les plus significatives. La beauté du système réside dans son interopérabilité avec d'autres sources de données. Par exemple, nous avons connecté le programme à un moteur de recherche afin de montrer de quelles propositions on parle le plus dans les médias, par recoupement avec les termes employés.

Tessel : À quels critères doivent répondre les données ouvertes pour être exploitables par une machine ?

Mark : Tout dépend du programmeur. Tim Berners-Lee, inventeur du *World Wide Web*, a établi une sorte de classement à cinq étoiles. La première étoile porte sur les conditions minimales auxquelles doivent satisfaire les données ouvertes : des données déstructurées publiées selon une licence ouverte. Il attribue le maximum de cinq étoiles aux données qui sont annotées de manière à pouvoir être sémantiquement liées à d'autres sources de données. C'est ce qu'on appelle des données ouvertes liées. Sans surprise, les programmeurs préfèrent donc généralement les données cinq étoiles...

En théorie, on peut extraire assez d'informations à partir de données déstructurées pour les analyser automatiquement, mais c'est un processus complexe et imprécis. Un PDF numérisé est un bon exemple de données déstructurées. Il s'agit d'une image. Pour en extraire le contenu, on devra utiliser un logiciel de reconnaissance optique de caractères, puis passer le résultat au travers d'un analyseur textuel pour obtenir un document structuré. Bien que ces logiciels existent depuis longtemps, et parfois même en *open source*, la plupart des programmeurs choisiront en priorité des données déjà présentées au format informatique, comme un tableau Excel ou sous forme de texte séparé par des virgules. En tant que programmeur, j'accorde autant

d'importance à la signification des données qu'à la sémantique et aux relations claires entre elles. Pour construire notre logiciel, nous disposons d'un certain nombre de données brutes issues de la base Parlis. C'est une base de données relationnelle, c'est-à-dire que les tables sont reliées entre elles via un indexage commun. Comprendre ces liens et à quelles informations ils se rapportent demande déjà beaucoup de temps. Heureusement, la base de données était bien documentée, avec une cartographie précise des relations entre les différentes tables. À certains moments, notre travail s'apparentait presque à celui d'un détective. En effet, il n'y a virtuellement aucune limite au chaînage de données. C'est un processus infini.

Sécurité et vie privée

Tessel : Bien que beaucoup de personnes plaident depuis longtemps en faveur des données ouvertes, les gouvernements s'y sont toujours farouchement opposés. D'où vient cet actuel revirement d'opinion ?

Josien : Évidemment, ce n'est pas simple. Si les programmeurs assurent qu'il n'y a qu'à essayer d'ouvrir les données pour voir ce qui se passe, on ne peut pas procéder de manière aussi brutale à la Chambre basse. La prudence s'impose.

Mieke : Leur plus grande crainte est de laisser sortir des données « impropres ». C'est la grande question dans les institutions. On ne veut divulguer que des données propres et s'assurer que son informatique interne soit parfaitement en ordre avant d'envisager une quelconque ouverture. Mais aucune société ne peut atteindre cet objectif.

Josien : La vie privée compte aussi. Les courriers sont très intéressants, car ils permettent entre autres de décortiquer les mécanismes internes de lobbying. Mais alors, on devrait révéler les auteurs des courriers, ce qui serait contraire au respect de la vie privée. Par conséquent, ces informations ne sont pas divulguées, au détriment de l'intérêt qu'elles pourraient constituer.

Mieke : La protection des données privées et



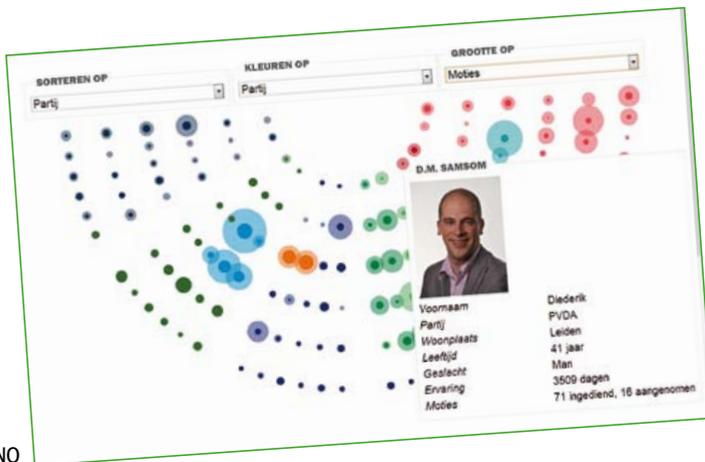
Mieke van Heeswijk



Josien Pieterse



Mark Bastiaans



L'application TNO

l'épuration des fichiers sont des problèmes bien connus et basés sur des arguments parfaitement légitimes.

Les temps changent

Mieke : Le succès des données ouvertes aux Pays-Bas a connu son plus grand coup d'accélérateur lorsque le ministère des Affaires économiques a décidé de se rallier au mouvement. Ils ont compris leur potentiel économique. La combinaison des fichiers permet l'émergence d'une foule d'applications qui profitent à l'innovation technologique et au marché de la connaissance.

TNT pour avoir le droit de s'en servir. Pourtant, à qui appartiennent réellement ces données ? Aujourd'hui, elles sont enfin disponibles gratuitement, une victoire pour l'association « Libérez les codes postaux ». [NdT : on saisit mieux l'enjeu, quand on sait que ces codes ne sont pas la simple immatriculation neutre d'adresses postales. Au Pays-Bas, le codage postal offre à qui sait les exploiter de considérables ressources commerciales potentielles.]

Josien : Je voudrais ajouter que Gerdie Verbeet a effectivement opéré un changement radical. Elle a toujours soigneusement veillé à proposer une politique claire et compréhensible par tous. Au début, son optique était assez traditionnelle puis, au fur et à mesure de ses échanges avec les partisans des données ouvertes, elle a compris le potentiel qu'elles offraient. Une démocratie transparente permet en effet aux citoyens de s'impliquer davantage. La clé de cette transparence réside dans la libre circulation de l'information. Elle a donc adapté sa politique en conséquence.

Tessel : Est-ce que le gouvernement est conquis ?

Mark : L'Europe commence à préparer le terrain au plan législatif à propos des données ouvertes.

un *hackathon* au parlement : première mondiale !

Les données parlementaires en font partie. On observe une évolution des mentalités au gouvernement. Alors que les budgets sont laminés, la collaboration entre citoyens, entrepreneurs et acteurs sociaux apparaît de plus en plus comme une bonne idée, presque une évidence. La société civile occupe aujourd'hui une position prépondérante dans diverses formes d'innovation et les gouvernements en prennent peu à peu conscience.

Il y a dix ans, on confiait beaucoup de données publiques à des sociétés privées. Le gouvernement considère à présent que cette situation est problématique, dans la mesure où les personnes qui ont contribué à l'acquisition de ces données n'y ont plus accès. Prenons l'exemple des codes postaux hollandais. Lorsque la Poste nationale a été privatisée pour devenir TNT, elle a reçu en dot la gestion des codes postaux. Sur internet, on trouve un grand nombre d'applications qui ont besoin de ces codes postaux. Leurs développeurs devaient donc s'acquitter d'une licence auprès de

À Bruxelles, ces projets théoriques sont toujours au stade embryonnaire. Une chose est claire : le changement devra venir du haut. Les détails ne sont pas encore réglés. On observe de nombreuses initiatives. Étonnamment, beaucoup de *hackers* et de techniciens affirment avec enthousiasme leur désir de participer au débat. Il y a donc une progression de part et d'autre, tant au niveau des acteurs de terrain que dans les hautes sphères politiques. Entre les deux, il reste beaucoup de travail. Ce changement doit s'opérer à l'échelle de la société tout entière. Un processus qui demande du temps et qui ne se contentera certainement pas d'un peu de code informatique par-ci et de quelques *hackers* par-là.

(120741 - version française : Robert Grignard)

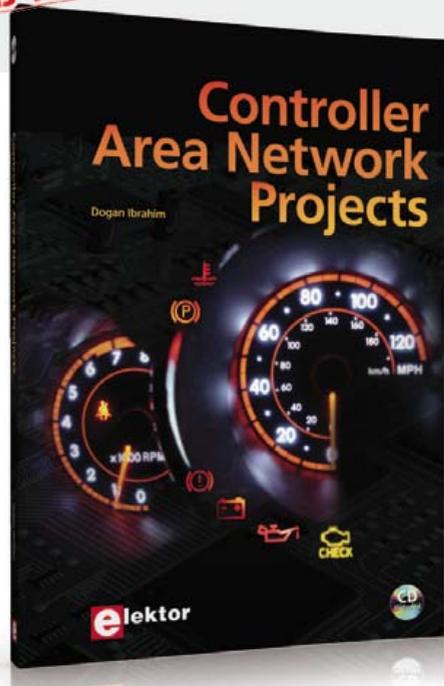
Liens

[1] <http://netdem.nl/>

[2] <http://appsvoordemocratie.nl/>

[3] www.tno.nl

CD-ROM INCLUS



Controller Area Network Projects

Ce livre (en anglais) s'adresse à tous ceux qui souhaitent en savoir (beaucoup) plus sur le bus CAN et ses applications : étudiants, ingénieurs en exercice, ou amateurs motivés.

Il donne les principes de base des réseaux CAN et montre comment élaborer des projets avec des microcontrôleurs sur le bus CAN.

Pour en profiter pleinement, il faut quelques connaissances de base en électronique. La compréhension du langage de programmation C est utile dans les derniers chapitres du livre ; connaître au moins un microcontrôleur de la série PIC sera un avantage, notamment quand vous passerez à la réalisation de projets à base de microcontrôleurs qui utilisent le bus CAN.

CD-ROM inclus

Ce livre comporte un CD-ROM gratuit qui contient une version de démonstration spéciale du compilateur mikroC.

ISBN 978-1-907920-04-2 • Langue : Anglais • 259 pages
34,50 €

 **elektor**

www.elektor.fr/arduino-buch

Nouvelle édition revue et augmentée par l'auteur



ISBN 978-2-86661-189-7
format: 17 x 23,5 cm
240 pages

44,50€

Ce livre est écrit par un spécialiste français de l'automatisation de l'éclairage de scène. Passionné à la fois de théâtre, de musique, de scène ET d'électronique, Benoît Bouchez consacre plus de la moitié de son ouvrage à la commande à distance, à l'automatisation, à l'utilisation des protocoles modernes tels DMX512 ou MIDI. Il en donne les éléments théoriques indispensables, puis les met aussitôt en pratique, à la portée de lecteurs désireux de comprendre et d'agir sur leurs installations.

Il présente également des protocoles encore plus récents comme RDM, ArtNet ou ACN. Conformément à la tradition des publications Elektor, les schémas des circuits électroniques sont dévoilés dans le livre, et dûment expliqués, les dessins des circuits imprimés peuvent être reproduits aisément. En outre le lecteur trouvera en téléchargement des programmes avec lesquels il pourra lui-même commander, tester, évaluer.

Le métier ne s'apprend qu'au contact de spécialistes, sur le terrain. L'expérience et le doigté ne se stockent pas dans des fichiers et ne s'impriment pas sur papier, mais ce livre contribuera à développer les talents d'éclairagiste de ceux qui le liront, amateurs ou professionnels. Il sera leur compagnon sur le chemin plein d'imprévus des innombrables possibilités offertes par les techniques numériques de commande de lumière.

 **elektor**

Logiciel, matériel et informations complémentaires sur
www.elektor.fr/dmx

des nouvelles du chaudron

Clemens Valens
(Elektor.Labs)

Elektor.Labs est en train de devenir le chaudron d'Elektor. Ça bouge, enfin ! Ça mijote bien. Il y flotte des arômes de plus en plus alléchants. Tout autour, il y a de la place pour tout le monde. Venez ! Je savais qu'il ne serait pas facile de changer les vieilles habitudes et qu'il faudrait du temps pour faire tomber les murs de notre donjon technologique. Ça se fissure. Une brèche est ouverte, à vous de vous y engouffrer.



Quelques questions fréquentes et leurs réponses

Q Si je poste un projet sur le site Elektor.Labs, est-ce qu'Elektor peut le publier dans le magazine sans m'avertir ?

A Certainement pas. Dès qu'un projet posté sur le site nous intéresse pour une publication imprimée, nous vous contactons. Vous en êtes l'auteur et recevez donc une proposition de contrat de publication en bonne et due forme. D'où l'importance de consulter votre courrier électronique à l'adresse utilisée pour vous identifier. Si nous ne pouvons pas vous joindre, votre projet ne débouchera sur rien.

Q Pourtant, j'ai découvert une photo de mon projet dans l'avant-première d'un numéro récent. C'est cool, mais ça m'a fait bizarre... car il n'y avait pas de contrat.

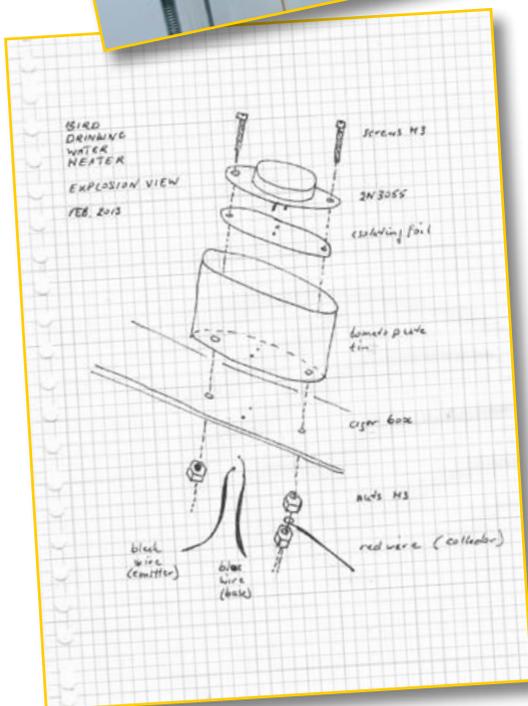
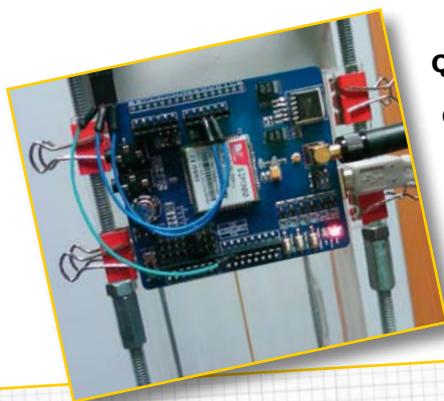
A Nous nous réservons le droit de parler de ce qui est mis en ligne sur Elektor.Labs afin de promouvoir ce site et son contenu. Vous voulez que votre projet rencontre une large audience, n'est-ce pas ? Nous aussi. Alors, nous le faisons savoir le plus fort et le plus largement possible. Chaque semaine, dans notre lettre d'information Elektor.Post nous évoquons aussi quelques montages dignes d'intérêt, ainsi que dans l'avant-première, tout à la fin du magazine imprimé. Mais jamais nous ne prenons de libertés avec le droit de nos auteurs.

Q J'ai remarqué la présence d'un curieux code à 6 chiffres dans le titre. C'est quoi ?

A La référence du projet dans la base de données d'Elektor qui, en cas de publication, deviendra le numéro de l'article. Sa présence indique que le projet a été retenu pour une publication éventuelle. Ça n'implique pas encore qu'il le sera effectivement. La plupart des auteurs dont le projet porte un tel numéro ont déjà été joints personnellement par Elektor. Inutile de rajouter un tel numéro vous-même.

Q Comment rendre mes projets plus populaires ?

A Donnez des informations claires, concises et complètes. Cela commence par le titre. Soignez les photos que vous mettrez en ligne. Écrivez en anglais. Vous



pouvez utiliser votre propre langue, mais en passant par l'anglais (utilisez *Google Traduction* !) vous multipliez vos chances. Ne craignez pas de publier vos schémas ou votre code source. Y a que ça qui intéresse les électroniciens. Ne passons-nous pas 99 % de notre temps à regarder des schémas que de toute manière nous n'utiliserons pas ? Mettez votre projet à jour de temps en temps, chaque fois il sera remis sur le haut de la pile et gagnera en visibilité sur la page d'accueil. Idem à chaque nouveau commentaire ou nouvelle contribution.

elektor labs

Q Ah oui, justement, c'est quoi la différence entre contribution et commentaire ?

A Une contribution, c'est l'adjonction d'un sujet de discussion supposé enrichir le projet, pointer une difficulté ou clarifier un point. Un commentaire, c'est un élément d'une discussion en cours, elle-même initiée par une contribution. C'est donc une matrice composée d'une liste (verticale) de sujets de discussions qui se développent chacune (horizontalement). Tout ça pour préserver la logique de l'information.

Q Pourquoi reçois-je des messages de mise à jour de mon projet alors que je n'ai rien changé ?

A L'administrateur d'un projet et ses membres reçoivent tous un message émis par le système à chaque mise à jour, chaque commentaire ou chaque contribution. Il arrive que des rédacteurs d'Elektor corrigent des fautes ou déplacent un projet... et hop, une nouvelle volée de messages de notification.

Q Que faire si je ne veux plus être administrateur de mon projet ?

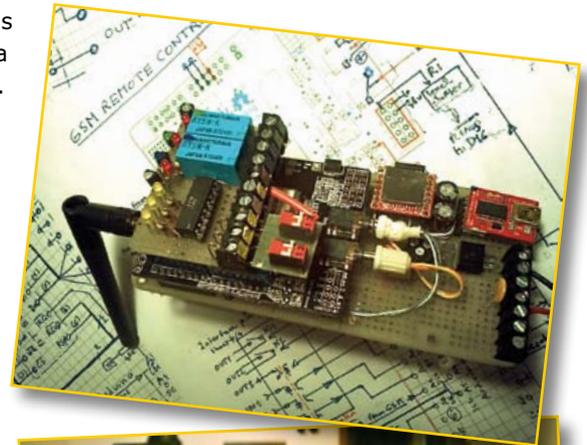
A Il suffit de demander à l'administrateur du site Elektor.Labs de vous libérer : labs@elektor.com.

Q Je viens de concevoir un projet fabuleux que je voudrais partager mais je ne suis pas membre de la communauté Elektor. Comme mon accès à Elektor.Labs se limite à la lecture, je ne peux donc rien publier dans Elektor ?

A Tous les projets intéressants ont leur place, soit dans le magazine soit sur le site. Il existe maintenant sur la page d'accueil une procédure particulière pour les internautes qui ne sont pas membres. Cliquez sur le bouton *Not-a-member?*, remplissez le formulaire et nous examinerons votre proposition. Si elle nous plaît, vous bénéficierez temporairement des privilèges des membres abonnés à la formule Greencard, ce qui vous permettra de mener à bien votre projet. Si la proposition est excellente, vous bénéficierez même d'une Goldcard et de ses privilèges, et vous recevrez le magazine Elektor gratuitement. Et, en plus, vous serez rémunéré pour la publication de votre article.

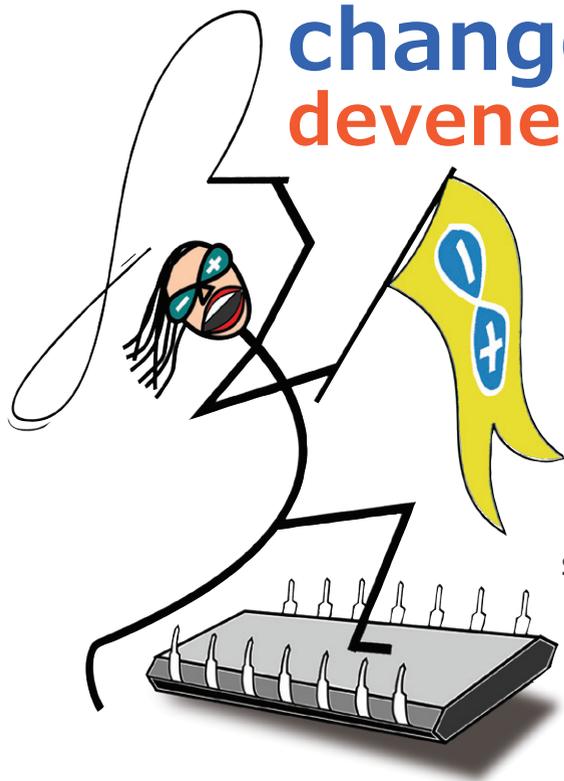
Q Comment je fais si j'ai d'autres questions?

A Envoyez un message à labs@elektor.com.

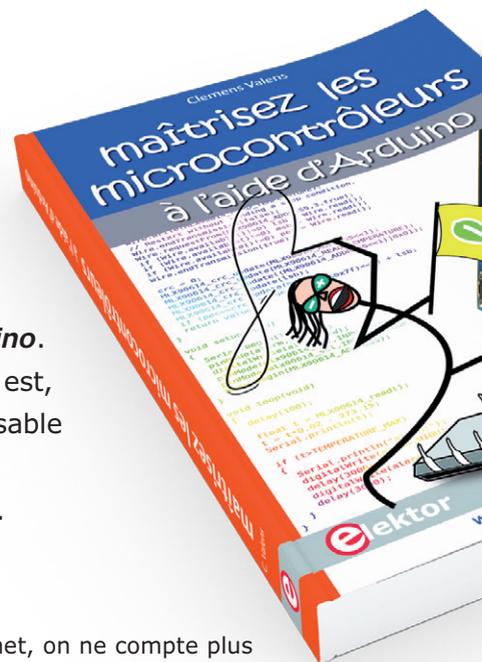


(130089)

changez de loisirs, devenez dompteur de puces



Un livre de pas moins de 352 pages vient de paraître sous le titre **maîtrisez les microcontrôleurs avec Arduino**. Sacré programme ! Son auteur est, entre autres activités, responsable technique du labo d'Elektor. Cela mérite des explications.



Mariline Thiebaut, Clemens Valens & Denis Meyer



DM à Mariline Thiebaut-Brodier, co-éditrice de l'ouvrage : *Il y a quelques mois, avant d'entreprendre le projet de publier le livre de Clemens, je vous avais demandé d'examiner le nombre et la nature des ouvrages déjà disponibles en français. En toute logique, nous aurions dû renoncer tant les platebandes Arduino semblaient irrémédiablement piétinées. Qu'est-ce qui vous a convaincue de prendre le risque quand même ?*

Mariline : Il y a beaucoup de livres en anglais sur *Arduino*, mais le barrage de la langue est un frein pour la plupart des débutants. En français, il n'y a que trois livres. L'un, écrit par un des co-inventeurs d'*Arduino*, se résume à une présentation de ce matériel (principes de base et premiers montages). Les deux autres livres s'intéressent essentiellement aux *shields* (modules qui viennent compléter les cartes *Arduino*), cela suppose donc que le lecteur connaisse déjà bien ces cartes elles-mêmes.

L'approche de Clemens est celle d'un véritable cours sur la programmation des microcontrôleurs, adossé à *Arduino*. Il ne laisse rien dans l'ombre. Il aurait pu prendre n'importe quel autre μC , un programmeur et un autre environnement de développement, mais *Arduino* lui permet de mettre en pratique aussitôt les notions expliquées.

Sur l'internet, on ne compte plus les sites avec des montages à base d'*Arduino*. Même si vous arrivez à les reproduire, vous ne saurez pas comment les adapter à vos besoins ni comment les corriger s'ils ne fonctionnent pas, ce qui est parfois le cas. Le cours de Clemens permet justement d'acquies les connaissances nécessaires pour bien comprendre ce qui se passe dans les cartes *Arduino*. C'est un cours sérieux...

DM : *Je vous interromps... je lis dans la table des matières : Boucles f'Or et les trois (t)ours et Petite arnaque entre amis, c'est sérieux ça ?*

Mariline : Oui, j'allais en parler. C'est de l'humour, sérieux et pédagogique. Les réalisations proposées par Clemens sortent des sentiers battus, on ne les oublie pas. On a envie de les comprendre et d'apprendre à en faire autant.

DM à Clemens : *Vous vous êtes bien amusé. Pourtant, un premier livre est plutôt une épreuve, quand on ignore si on est parti pour 150 ou 350 pages et que le volume de la matière semble augmenter à mesure qu'on la prémâche pour le lecteur.*

Clemens : Au début je pensais que ça allait être assez facile, qu'il suffisait de commencer à écrire,

« La plate-forme *Arduino* est idéale pour introduire graduellement les concepts de base des microcontrôleurs ; on ne s'y perd pas dans la mise en place d'un environnement de développement complexe avec une pléthore de fonctions et d'options. »

j'avais des dizaines de sujets en tête.

Or, très vite je me suis aperçu que pour arriver à quelque chose de correct, il fallait être rigoureux dans les explica-

tions. Ce qui est évident pour l'auteur ne l'est pas forcément pour le lecteur. Il est nécessaire de simplifier et d'expliquer un maximum.

Du coup, le nombre de pages a commencé à augmenter de façon spectaculaire, d'autant que je n'ai voulu ni simplifier les montages ni abandonner certains sujets. Réfléchir à la façon de présenter les matières m'a beaucoup plu. J'ai eu beaucoup de satisfaction aussi à tout illustrer moi-même. Ça a été plus de travail que je ne l'imaginais, mais je l'ai fait avec beaucoup de plaisir jusqu'au bout.

DM : *Un premier livre en français pour quelqu'un comme vous qui auriez pu l'écrire plus facilement en néerlandais ou en anglais, vos « autres » langues maternelles, chapeau.*

Clemens : Si j'avais écrit en néerlandais ou en anglais, j'aurais gagné deux ou trois mois. La relecture par les rédacteurs aurait été plus rapide et il y aurait eu moins de corrections. Mais, il n'y aurait pas eu ce défi qui m'a stimulé. Depuis que je vis en France, je tente de m'intégrer au mieux. J'ai même acquis la nationalité française, un défi d'ordre administratif. Mon accent ne s'évaporera probablement jamais, je ne suis pas un bon perroquet, mais en lecture et en écriture j'ai l'ambition de devenir aussi bon qu'un Français de souche. Écrire ce livre en français s'inscrit dans cette ambition. Il reste des faiblesses et des zones d'incompréhension, mon français reste un français d'étranger, mais grâce à Mariline, qui a patiemment débusqué chaque faute, chaque impropiété, le français dans ce livre est

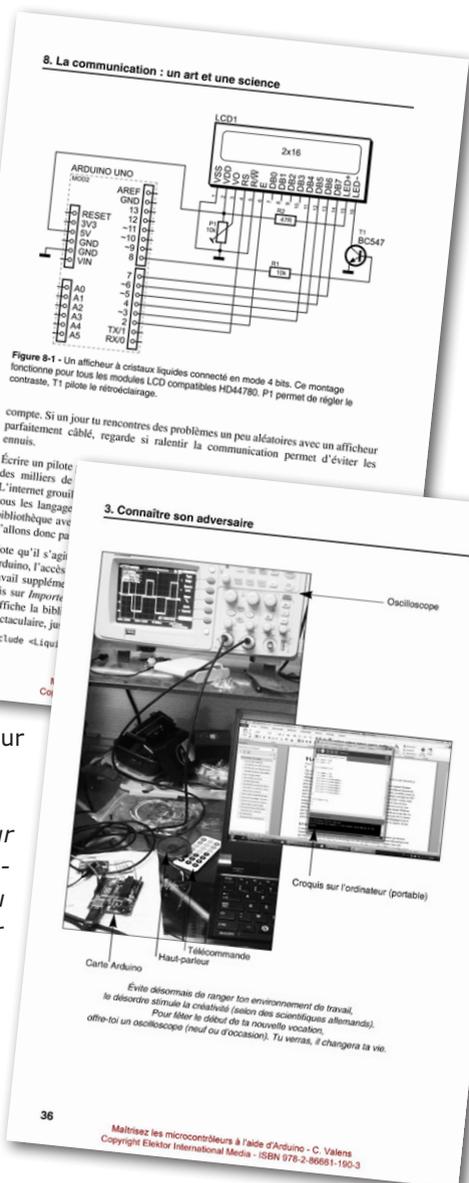
impeccable. Grâce à cette attention et au délai supplémentaire, le contenu du livre a pu mûrir, ce qui m'a permis d'améliorer les montages et leurs explications.

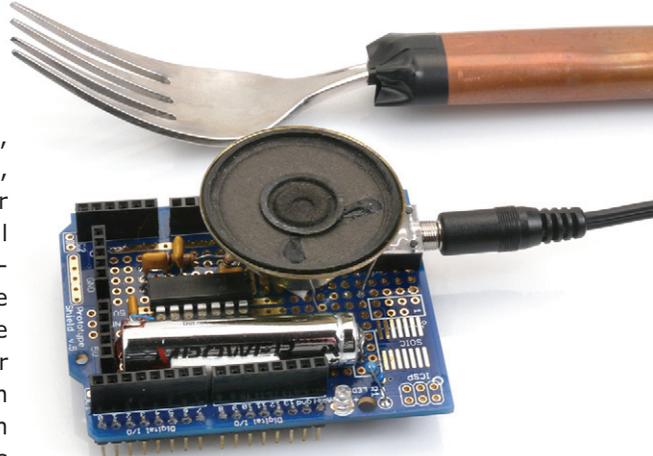
DM : *C'est du beau travail, bravo à tous les deux. Aviez-vous un modèle à suivre ou peut-être au contraire un anti-modèle qui vous motivait « à tout faire sauf ça » ?*

Clemens : Aucun modèle. Ce que je ne voulais en aucun cas, c'était un livre « à la mode *Arduino* », simplifié à l'extrême avec des dessins en 3D qui, à mon avis, n'expliquent finalement pas grand-chose. Je ne voulais pas non plus d'un livre d'applications pour *Arduino*, il y en a déjà plein. On peut faire beaucoup avec *Arduino*, mais ce n'est pas une bonne idée de vouloir tout faire avec cette plate-forme. Pour autant que je sache, il n'y a pas non plus de livre du style « **30 projets pour marteau** ». Mon but n'est pas d'utiliser à tout prix *Arduino*, mais d'utiliser *Arduino* pour atteindre un but : le mien est d'expliquer le micro, pas de mettre en avant la plate-forme *Arduino*. J'ai aussi voulu expliquer certains sujets qu'on ne traite pas souvent, mais auxquels on se heurte toujours dans la vraie vie de développeur d'applications à μC .

DM : *Il me semble que votre avis sur Arduino a changé au fil de votre progression. Je caricature un peu, mais au début, vous pensiez plutôt à l'intituler « maîtriser les microcontrôleurs malgré Arduino » et maintenant c'est bel et bien « à l'aide d'Arduino ».*

Clemens : Je pense et je dis que tout ce qu'on peut faire avec *Arduino*, on





10.6 L'interruption de trop

la mémoire est à moitié vide... excellent !

Figure 10-1 -
La programmeur optimiste.

interruptions s'il est vraiment nécessaire que tout continue tout le temps. Par contre, si tu fais cela, tu libères encore un autre mauvais génie de la boîte de Pandore des interruptions : le débordement de la pile (*stack overflow* en anglais).

10.6.1 La pile
Le micro possède une mémoire spéciale, appelée la pile (*stack*), qui joue un peu le rôle de médiateur entre programme principal et fonctions ou routines d'interruption. Elle est utilisée pour passer des données ; la dernière utilisée pour passer des données locales et quand le micro avait prévu une routine d'interruption ou routine normale lui, sur la pile, il était au moment où le micro avait prévu une fonction ou routine normale. Celle-ci aura assez de place dans le programme, appelée une troisième, toutes les adresses de données sont également stockées.

9.2 Réaliser un collier de secondes

lieu d'envoyer les durées des bits sur le port série, elles sont maintenant stockées dans la variable *dt*. Si la valeur de *dt* est supérieure à 50 ms, mais inférieure à 150 ms, nous pouvons supposer que nous avons affaire à un niveau logique 0 et à un caractère « 0 » est enregistré dans la trame *bits*. Si *dt* dépasse les 150 ms, mais est inférieure à 250 ms, la valeur du bit reçu est de 1 et un caractère « 1 » est ajouté à *bits*. Enfin, si *dt* dépasse les 1 500 ms, nous pouvons supposer que nous avons détecté l'absence du bit n° 59 qui indique la fin de la trame. La trame est envoyée sur le port série avant d'être vidée pour créer de la place pour la réception de la prochaine.

Grâce à cette méthode, la polarité des bits n'a aucune importance et le croquis fonctionnera avec tous les signaux, inversés ou pas. Comme tu l'as remarqué, j'ai choisi une marge de ±50 ms pour les deux durées de bit et même plus pour la fin de trame. C'est peut-être trop et pour plus de certitude tu pourras diminuer les marges.

Les trames envoyées sur le port série ressemblent à ceci :

```
00100000
00101001001110000100100001001010011001100000010010010000
0111000001111000010011000010010011001100110000010010010000
000010101010101001001001001001001100110000010010010000
```

Elles contiennent toutes 59 bits, sauf la première qui correspond à un fragment de fin de trame, puisque l'algorithme n'était pas encore synchronisé avec le flux de bits.

La prochaine étape est de décoder la trame. Son format est bien documenté sur l'internet, voilà ce que ça donne pour la première trame de 59 bits :

Bit(s)	Description	Reçu(s)	Décodé(s)
0 à 14	Réserve	0010100100111000	-
15	Émetteur de réserve	0	0
16	Basculement émetteur au début de la prochaine heure.	0	0
17, 18	Décalage par rapport à l'heure UTC.	10	2
19	Ajout d'une seconde intercalaire à la fin de l'heure.	0	0
20	Début de la transmission des informations horaires.	1	1

Tableau 9-1 - La signification des bits dans la trame horaire DCF77.

Maitrisez les microcontrôleurs à l'aide d'Arduino - C. Valens
Copyright Elektor International Media - ISBN 978-2-606611-190-3

peut le faire mieux sans. Et pourtant, je me suis aperçu que, petit à petit, je me suis mis, moi aussi, à utiliser la plate-forme *Arduino* comme outil pour mon propre travail. Il est tellement facile de programmer une carte *Arduino* qu'il est souvent plus simple d'essayer une idée rapidement sur une telle carte que d'extraire d'un placard le programmeur pour un micro quelconque avec ses outils de programmation dont on n'arrive pas à retenir les particularités. La carte *Arduino* ne quitte plus mon bureau et je l'utilise parfois même comme calculatrice : c'est plus rapide avec *Arduino* qu'avec Excel ou Windows. D'ailleurs, je viens de terminer un projet pour un micro PIC de *Microchip* que

j'ai prototypé sur une carte *Arduino* avec son micro AVR d'*Atmel*.

DM : Autrement dit, *Arduino* acquiert un caractère d'outil universel, un couteau suisse ou le multimètre de la pratique moderne de l'électronique ? En quoi votre livre contribuera-t-il à cette évolution ?

Clemens : *Arduino* est un outil remarquablement pratique pour par exemple se familiariser avec un composant à inter-

comme fin en soi. Elektor explique régulièrement comment utiliser un transistor, alors pourquoi pas faire pareil pour le micro ? C'est un composant incontournable. Avec *Arduino* je pense que nous pourrions faire de belles choses.

DM : La concurrence est féroce. Les cartes à μC se succèdent et nous submergent, vous-même êtes l'auteur d'une telle carte, mais dans votre livre vous vous en tenez à *Arduino* d'un bout à l'autre. Et pourtant, votre ambition est de mener votre lecteur à la maîtrise des μC « en général ». Ne serait-ce pas un paradoxe ?

Clemens : Non, pas du tout. On a souvent tendance à réduire *Arduino* à sa carte, au seul matériel. Or, il s'agit d'une véritable plate-forme de développement, consistant en une carte à μC , un environnement de programmation simple, mais efficace et des bibliothèques pratiques. C'est un

« Tout ce qu'on peut faire avec *Arduino*, on peut le faire mieux sans. »

face numérique, sérielle, I²C, SPI ou autre encore avant de le mettre en œuvre. Les entrées analogiques sont utiles quand on veut mesurer plusieurs tensions en même temps. Pas besoin de posséder cinq multimètres que de toute façon on n'arrive pas à lire tous d'un coup et qui en plus s'éteignent au mauvais moment.

C'est d'ailleurs précisément ce qu'espèrent les concepteurs de la plate-forme : qu'à terme *Arduino* devienne incontournable dans la panoplie de l'électronicien, à côté du multimètre, de l'oscilloscope et du fer à souder. Dans mon livre, j'utilise beaucoup la carte pour préparer le terrain avant de mettre au point une application. Elektor devrait aussi montrer davantage comment utiliser le micro en tant qu'outil et moins

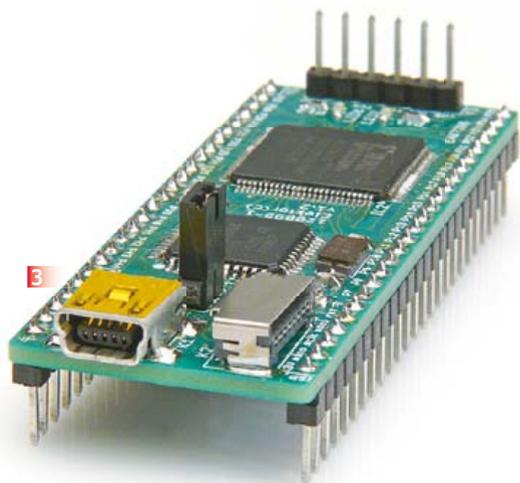
tout et sa force est là. Une telle plate-forme est idéale pour introduire graduellement les concepts de base des μC ; on ne s'y perd pas dans la mise en place d'un environnement de développement complexe avec une pléthore de fonctions et d'options. *Arduino* nous épargne cette couche de complexité assez stérile et nous invite à nous concentrer sur le micro lui-même.

Il y a déjà tellement de choses à apprendre.

130095

Attention, si vous cliquez sur ce lien, votre vie va changer :

www.elektor.fr/arduino



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 sérieuse 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 Boucle d'oreille à LED

La fantaisie et l'imagination sont des ressources précieuses aussi en électronique. Voici une petite création qui vous permettra de faire la fête... en boucle, et d'épater vos ami(e)s, et plus si affinités.

Caractéristiques :

- boucle d'oreille pilotée par microcontrôleur, avec motifs lumineux programmables
- carte ronde et légère, avec 8 LED symétriques
- très basse consommation (12 mW avec les LED multiplexées)

Réf 120607-91

3 Et l'homme créa sa puce

Sans aucun doute l'un des composants les plus universels, mais aussi des plus complexes dans l'électronique actuelle, c'est le **FPGA**, dédale de portes et d'autres briques logiques et numériques, pour composer, sur une seule puce, les circuits intégrés dont vous rêvez et dont vous avez besoin.

Le laboratoire Elektor a conçu une nouvelle carte d'expérimentation pour FPGA, pour faciliter la vie des électroniciens désireux de se familiariser avec cette famille de logique programmable aussi excitante qu'elle est exigeante. Cette carte servira aussi de support à une série d'articles qui expliqueront comment s'y prendre, et proposeront par exemple la réalisation d'un projet ISE pour programmer cette carte de FPGA, ou montreront comment hiérarchiser le projet à

l'aide de composants développés soi-même : un compteur-décompteur avec affichage sur deux chiffres à 7 segments...

Réf 120099-91

4 Éclairage annulaire à LED

L'éclairage annulaire est une source de lumière utilisée avec les *webcams*, microscopes et appareils photo. Il se monte autour de l'objectif pour fournir un éclairage uniforme quand le sujet est très près. Les sources annulaires sont utilisées pour le portrait ou la photo rapprochée, ou pour éclairer des objets sous une loupe binoculaire. Assemblez vous-même un éclairage annulaire puissant à partir d'un kit de composants.

Astuce : Avant de confirmer votre commande sur le site d'ElektorPCBService, vous devriez considérer la possibilité d'y joindre l'excellent livre (en anglais) *Mastering Surface Mount Technology*, le deuxième de la série *LabWorX* d'Elektor. Ce livre vous emmène dans un cours accéléré sur les techniques, trucs et savoir-faire en vue d'introduire les composants montés en surface (CMS) dans vos méthodes de travail. Visitez www.elektor.fr/labworx pour plus d'information.

Réf 129003-71



1



2



3

kit Audion *Radiomann* (vers 1956)

Au rayon des jouets éducatifs

Peter Beil
(Allemagne)

Vous rappelez-vous le temps enfui de votre jeunesse, quand vous faisiez vos premiers pas dans le monde de l'électronique ? Pour moi – et pour beaucoup d'autres, je suppose –, la radio était un objet de fascination absolue au début des années 1950. Construire soi-même un récepteur était pratiquement impossible parce que, au moins dans mon pays, les fournisseurs de composants qui vendaient par correspondance étaient rares et distants.

Cette lacune du marché s'est trouvée comblée par le kit nommé *Experimenter Set* proposé par l'éditeur *Kosmos* de Stuttgart, qui a ouvert pour les jeunes gens curieux et les ingénieurs en herbe une porte sur l'électronique de l'époque. Sous des dehors un peu austères (figure 1), ce matériel permettait tout de même pas moins de 80 expériences.

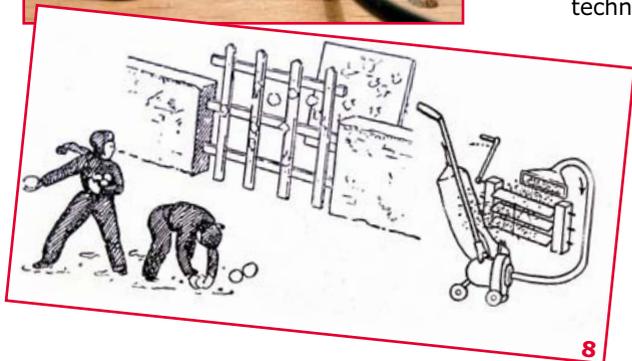
Il n'est pas inutile de rappeler que la technologie des semi-conducteurs était encore balbutiante à l'époque et que le tube à vide dominait la majorité des circuits électroniques. Les tubes permettaient non seulement l'amplification, mais aussi le calcul, la régulation ou le redres-

sement. La télévision existait déjà, mais pour les amateurs et *a fortiori* pour les jeunes garçons cela ne signifiait encore rien de pratique.

Le kit *Radiomann* proposait une méthode presque ludique pour faire connaissance avec la théorie ondulatoire, les fréquences audio et radio ou la technologie des tubes à vide (fig. 2). Tout se faisait d'une façon merveilleusement simple : le condensateur était un simple morceau de plastique avec une feuille de tôle collée de part et d'autre (fig. 3) et on faisait soi-même une résistance d'un gros trait de crayon sur un morceau de carton (fig. 4). J'ai vérifié depuis la valeur de cette « résistance » : elle était presque exactement de 2 MΩ. Elle était peinte à l'encre de Chine



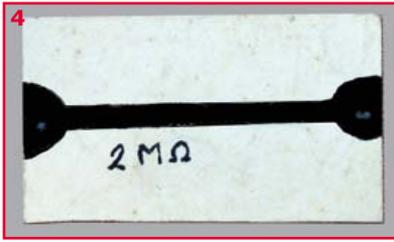
7



8



9

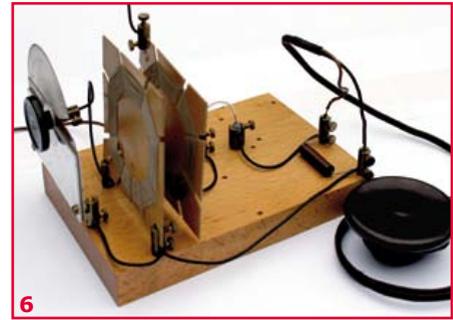
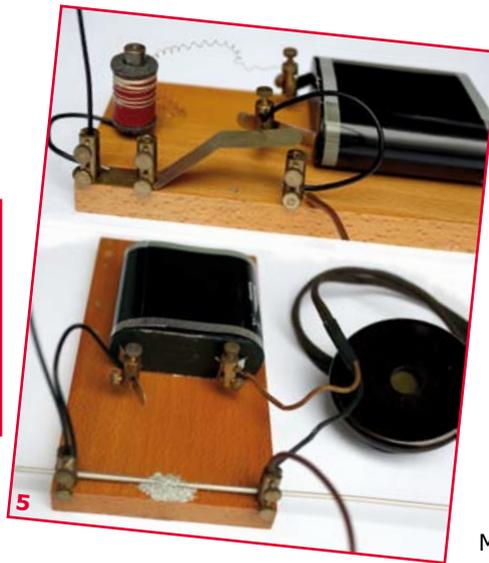


noire et « calibrée » avec un trait de crayon par-dessus.

Les expériences étaient montées sur une planche de bois avec des bornes en laiton qu'on enfichait dans les trous. On apprenait de cette façon avec l'exemple du « cohéreur » ce qu'était l'induction et on établissait sa première liaison « sans fil » (fig. 5). Pour ceux qui ne connaissent pas le cohéreur, c'était un simple indicateur de la présence d'ondes électromagnétiques, constitué de deux électrodes et de limaille de fer dans un tube de verre.

En ce temps-là, le « détecteur à cristal » avec rétroaction positive s'est imposé comme le « récepteur » le plus approprié (fig. 6). La diode au germanium était encore inconnue et on construisait un « semi-conducteur » adéquat avec un cristal de galène (sulfure de plomb existant à l'état naturel) et un morceau de fil très pointu, avec la difficulté inévitable de trouver la position d'une jonction optimale (fig. 7). À proximité d'un émetteur sur petites ondes, vous pouviez même attaquer directement un petit haut-parleur avec ce détecteur !

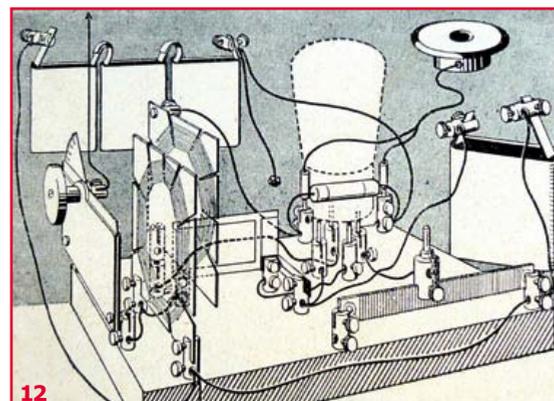
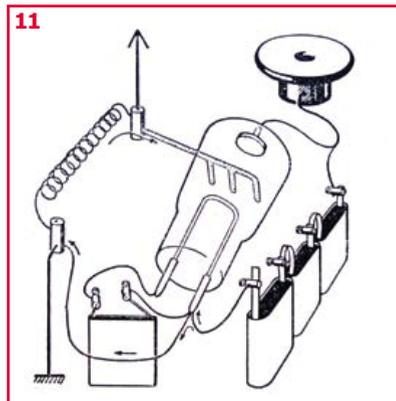
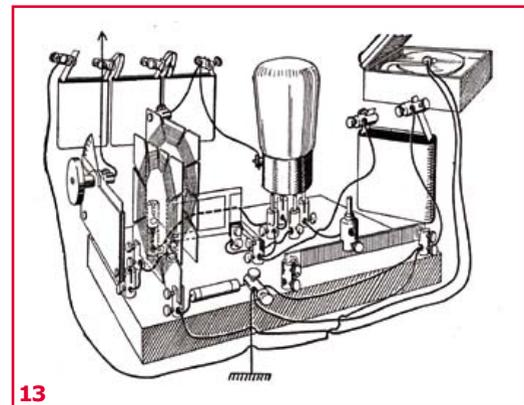
L'initiation aux secrets du tube à vide passait par des exercices parfaitement compréhensibles, même pour de jeunes garçons. Ainsi, la fonction de l'anode (ou plaque) était comparée à celle d'un aspirateur, le flux d'électrons à des boules de neige lancées à travers un grillage ou un store vénitien (fig. 8).



Malheureusement, mon tube à vide

a disparu au cours de ces soixante dernières années. Tout ce qui en reste est l'illustration du couvercle de la boîte (fig. 9). C'était à l'origine le type RE074d de *Telefunken*, un tube dit à grille à charge d'espace. Une fois devenu indisponible dans ces années d'après-guerre, il a été remplacé par le type DM300 conçu spécialement (fig. 10). Comme ce n'était plus un tube à charge d'espace, on le convertissait astucieusement en une tétrode normale en inversant les connexions des grilles. Dans ces applications, il fonctionnait sans aucun problème, malgré la tension continue de chauffage du filament (défavorable techniquement).

Le tube était d'une conception inhabituelle, avec une tension de chauffage de 3,5 à 4 V et une double grille. La tension de plaque avait une valeur inoffensive de 12 à 20 V. Signalons que ce n'était possible que parce que la deuxième grille était portée au potentiel de la plaque. Tout le montage était alimenté par des piles plates de lampe de poche de 4,5 V, courantes à l'époque (zinc-charbon, CEI 3R) (fig. 11). Le bouquet des expériences était



la construction d'un « audion » à réaction qui offrait une assez bonne sélectivité (**fig. 12**). En raccordant le tourne-disque domestique (si vous en aviez un) au récepteur pris « à l'envers », on pouvait émettre à destination du récepteur de radiodiffusion sur une distance incroyable d'environ trois mètres (**fig. 13**). Manifestement, cela résultait d'une certaine espièglerie, parce que cette expérience ne faisait pas partie des variantes proposées.

La personne qui a conçu cet *experimenter set* et d'autres était le Dr Wilhelm Froehlich, enseignant dans une école proche du lac de Constance. Son objectif initial était de donner aux élèves un accès à la technologie par des expériences simples de physique en classe. Pendant les années soixante, le *Radiomann* a été mis à jour dans une certaine mesure avec une pentode EF89 et un transistor. Par la suite, Burkhard Kainka, auteur bien connu et contributeur d'Elektor, a adapté le kit à des technologies plus perfectionnées.

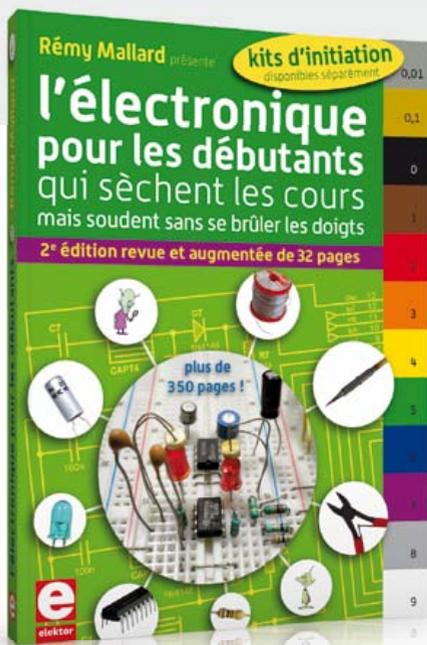
La version décrite ici date de 1956 environ. Il coûtait l'équivalent de 5,4 € (46 € aujourd'hui), plus 2,20 € (18,40 € aujourd'hui) pour le tube, qui était vendu séparément. À l'époque, ça représentait une somme considérable que je n'ai pu réunir qu'au prix de patients sacrifices, heureusement récompensés par de nombreuses heures de plaisir. La maison d'édition *Franckh-Kosmos* est toujours à la pointe dans le matériel d'enseignement et d'expérimentation, puisqu'elle propose une plate-forme à microcontrôleur *Arduino*. Il reste sur le ouëbe beaucoup d'informations sur le vieux *Radiomann*. Tapez « Kosmos Radiomann » dans votre moteur de recherche favori, vous serez étonné.

(120650 - version française : Jean-Paul Brodier)

Rétronique est une rubrique mensuelle sur les pages glorieuses et jaunies de l'électronique, avec occasionnellement des montages de légende décrits dans Elektor. Si vous avez des suggestions de sujets à traiter, merci de les télégraphier à redaction@elektor.fr

Dessins originaux reproduits avec l'aimable autorisation de l'éditeur Franckh-Kosmos à Stuttgart. Photos de l'auteur.

Publicité



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

deux kits d'initiation disponibles

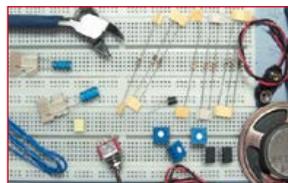
Voici le cadeau idéal pour partager votre passion de l'électronique avec vos enfants, petits-enfants, neveux... et autres *geeks*

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 !



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 €

elektor

Offre spéciale : livre + deux kits = 81,50 € au lieu de 87,50 €
Informations complémentaires et commande :
www.elektor.fr/debut

Hexadoku casse-tête pour électroniciens

Cette grille contient-elle un poisson d'avril ? Qui le sait... Le seul moyen de le savoir est de la déchiffrer, ce qui pour un hexadoku revient à la *chiffrer*. Remplissez la grille selon les règles, envoyez-nous votre solution, avec ou sans poisson, et vous gagnerez peut-être l'un des quatre chèques-cadeaux mis en jeu par Elektor.

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} mai 2013** :
Elektor c/o Regus Roissy CDG – Le Dôme – 1, rue de La Haye
BP 12910 – 95731 Roissy CDG
Courriel : hexadoku@elektor.fr

Les gagnants

La solution de la grille du numéro de janvier (415-6) est : **02518**

Le gagnant des **100€ à valoir sur des circuits imprimés Eurocircuits** est **Jean-Claude Pouillet** (France).

Les **3 chèques-cadeaux Elektor** d'une valeur de **50 €** chacun vont à :

Mike Corteling (Australie), Hans Pfister (Allemagne) et Georgios Mpeinas (Grèce).

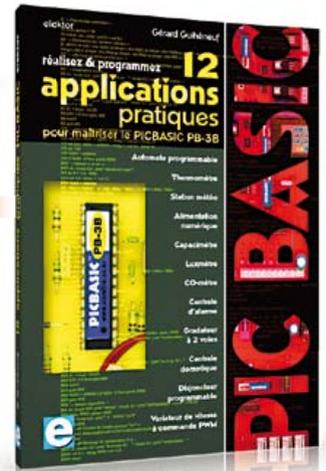
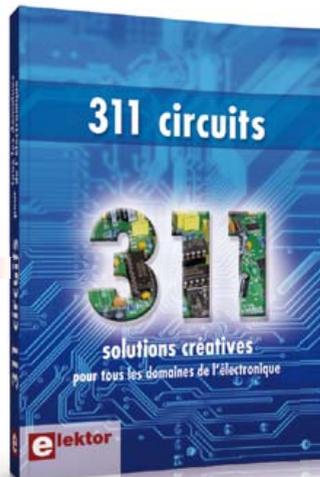
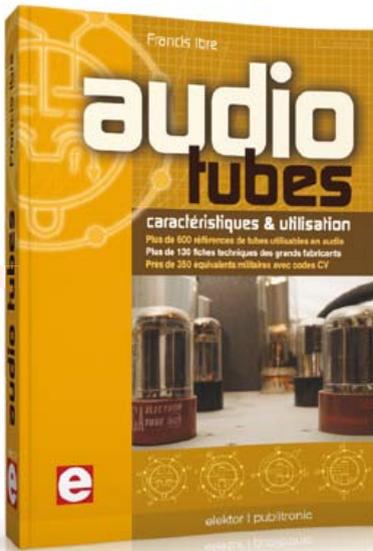
Bravo à tous et félicitations aux gagnants !

9				E	7	A			8						
5	B	3					8	C	6		0	2			
A	3	0	8	4	7		2		E					C	
	7			0	5	2		3	9	B	4	A	D		
		E			F	9									
	9		5	2	B	0		A	C		7	E		4	
	1	F			7	E				A	B			3	
	A		9				F				0	1			
	2		A				B				6	4			
		5	7			8	6				1	C		D	
		F		2	7	9	C		5	4		B	3		0
			C			4	A								
		E			6	C	A		0	5	4	D	1	8	
0		C	6	D	E		8			F		2			B
1		A		F						7	D	0		6	E
4					2		8		1				F		

F	A	C	4	D	0	2	5	1	8	B	7	9	3	E	6
9	5	D	E	6	7	1	8	2	0	A	3	F	B	4	C
2	6	B	1	3	9	4	A	D	C	E	F	0	7	5	8
0	7	3	8	E	B	F	C	4	5	6	9	1	D	2	A
1	C	F	A	2	4	7	9	3	6	D	8	B	E	0	5
6	0	E	9	5	1	D	F	B	7	4	C	2	A	8	3
B	2	5	3	8	A	6	E	9	1	F	0	C	4	D	7
8	D	4	7	0	3	C	B	A	2	5	E	6	F	1	9
3	B	1	2	C	D	9	7	E	F	8	A	5	0	6	4
4	E	6	C	F	2	8	3	0	9	7	5	D	1	A	B
A	9	0	5	4	E	B	1	6	D	3	2	7	8	C	F
7	F	8	D	A	5	0	6	C	B	1	4	E	9	3	2
5	1	9	6	7	C	3	4	F	A	0	D	8	2	B	E
C	4	7	B	9	8	A	0	5	E	2	1	3	6	F	D
D	8	A	F	1	6	E	2	7	3	C	B	4	5	9	0
E	3	2	0	B	F	5	D	8	4	9	6	A	C	7	1

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.

•Elektor e-CHOPPE



Caractéristiques & utilisation

1 Audio tubes

- 9 tableaux descriptifs, par types, pour choisir parmi plus de 600 tubes utilisables en audio (préampli, puissance, redresseur)
- Tous les équivalents connus des tubes usuels, y compris les très nombreux tubes anglais CV
- Codes CV de près de 300 équivalents militaires en regard des références connues
- Plus de 130 fiches techniques des grands fabricants
- Valeurs maximales et nominales pour différents modes de fonctionnement
- Recommandations pour interpréter les fiches techniques et en tirer pleinement parti (test, mesure, remplacement des tubes)
- Exemples concrets de calculs, de schémas et d'illustrations

pages - ISBN 978-2-86661-174-3 - 60,00 €

Kit complet avec circuit imprimé, afficheur et microcontrôleur

2 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 €

Solutions créatives

pour tous les domaines de l'électronique

3 311 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 « 300 circuits » (301... 302... 303... 304... 305... 306... 307... 308... 309... 310... 311 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. Ils forment un véritable catalogue d'idées, de trouvailles et d'astuces. C'est une source d'inspiration inépuisable, et à partir de laquelle chacun élaborera ses propres variantes qu'il combinera ensuite à sa guise avec d'autres circuits. Tous les domaines familiers et usuels de l'électronique sont abordés :

alimentations, régulateurs et chargeurs • audio & vidéo • communication • hautes fréquences • informatique • jeux & modélisme • maison & automobile • mesure & test • processeur & contrôleur • robots et leurs accessoires.

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

Incontournables microcontrôleurs, d'accord. Insurmontables microcontrôleurs, non !

4 12 applications pratiques pour maîtriser le PICBASIC PB-3B

Si avant d'utiliser un tel composant il faut apprendre l'assembleur ou le langage C, l'amateur ou l'électronicien débutant risquent de décrocher bien avant le stade des premières satisfactions, celui à partir duquel tout devient possible. Grâce à la simplicité des microcontrôleurs PICBASIC programmables en langage BASIC, l'électronique numérique programmable est désormais à la portée de tous. Ces douze applications pratiques du microcontrôleur PICBASIC PB-3B couvrent des domaines variés : la domotique, la protection des biens, la mesure, l'automatisation et l'électronique de puissance.

pages - ISBN 978-2-86661-166-8 - 43,50 €



5

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

5 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 !

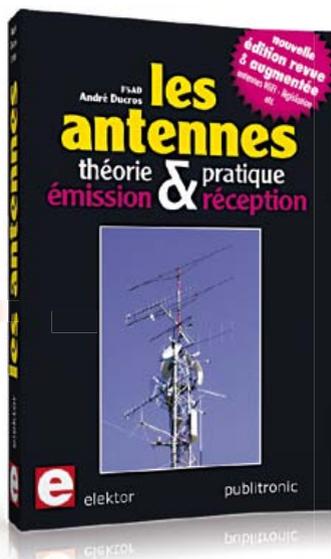
Carte Linux Elektor (montée et testée)

Réf. : 120026-91 • 64,95 €

théorie & pratique – émission & réception

6 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 : antennes filaires, antennes à gain, antennes THF, antennes courtes, antennes à large bande et multibandes, antennes de réception. La dernière partie est consacrée aux ultimes réglages :



6

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 €

Tous les articles d'ELEKTOR de l'année 2012

7 Elektor DVD 2012

Ce DVD-ROM réunit tous les articles d'ELEKTOR, le mensuel d'électronique et de micro-informatique appliquées, parus au cours de l'année 2012. Il contient non seulement le texte des articles ainsi que les schémas, mais aussi tous les dessins des circuits imprimés, sous forme de fichiers à haute résolution. Ceci permet à l'utilisateur de modifier à sa guise les dessins existants à l'aide d'un programme adéquat. Dès lors, rien ne s'oppose plus à l'exportation des documents vers un autre format à la convenance de l'utilisateur.

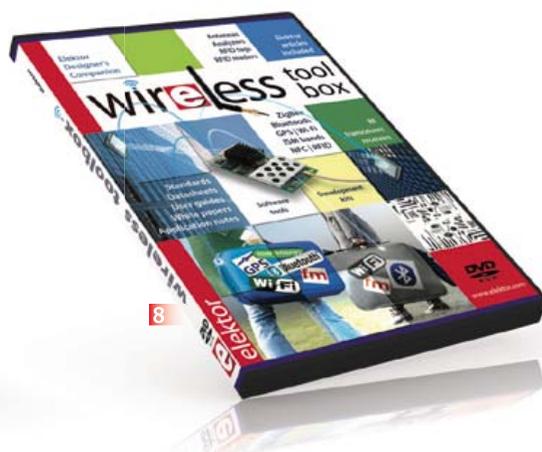
Une fonction de recherche rapide dans la table des matières sera très utile aux étudiants qui utiliseront ce DVD-ROM dans le cadre de leurs études.

Ce DVD-ROM contient aussi les versions anglaise, américaine, allemande, espagnole et néerlandaise des articles de l'édition française.

ISBN 978-90-5381-273-0 • 27,50 €



7



8

plus de 90 articles d'Elektor en français en bonus
8 DVD-ROM Wireless Toolbox

Sur ce DVD-ROM vous trouverez des documents techniques et des outils pour libérer vos propres systèmes électroniques de leurs fils. Selon la distance à couvrir, le choix est vaste : quelques cm avec la communication en champ proche (NFC) et l'identification par radiofréquences (RFID), des dizaines de mètres avec Bluetooth, Wi-Fi et ZigBee, des milliers de km avec la réception de données par GPS. Le plus difficile est de s'y retrouver dans la jungle des normes. D'où l'utilité de ce DVD, avec sa collection de normes, de protocoles standard et propriétaires (p. ex. MiWi de Microchip, SimpliciTI de Texas Instruments), sa revue des plages de fréquences utilisées (ISM p. ex.) et bien davantage.

ISBN 978-90-5381-268-6 – 32,50 €

Informations complémentaires et gamme complète sur :

www.elektor.fr/e-choppe

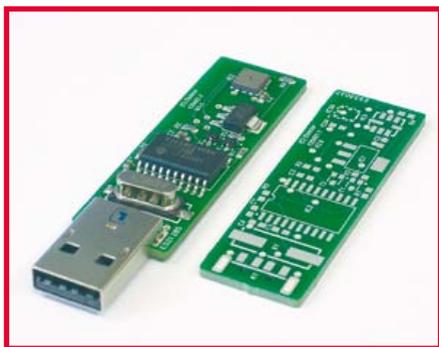
Elektor/Publitrone SARL

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

•bientôt dans Elektor



barostick

Vous avez déjà vu un baromètre sur une clé USB ? On en trouve, mais ce n'est pas facile. Ce serait pourtant un accessoire intéressant. Il n'en fallait pas plus pour nous donner envie de publier une telle réalisation. Le circuit à PIC pourra être équipé d'un capteur de pression BMP180 ou BMP085 de *Bosch Sensortec*.



initiation à Python

Pour commander la micro-électronique couplée à un PC, on fait de plus en plus souvent appel à Python pour programmer le versant PC du logiciel. C'est pourquoi Elektor proposera plusieurs articles d'initiation pour vous inciter à utiliser cet interpréteur, pour lequel il existe des bibliothèques variées, riches et robustes.



bibliothèque de micrologiciels embarqués

Cette *embedded firmware library* de conception modulaire doit nous faciliter la programmation des microcontrôleurs. Le grand atout d'une telle bibliothèque est de rendre possible le portage du code, c'est-à-dire de le libérer d'une configuration matérielle précise. En principe elle convient aussi sous Linux embarqué.

Informations préliminaires non contractuelles.
Le numéro de mai paraîtra le 27 avril.

ECD7

NOUVELLE EDITION

Base de composants d'ELEKTOR

Cet ensemble consiste en une quadruple banque de données (circuits intégrés, transistors, diodes et optocoupleurs) complétée par neuf applications satellites, au nombre desquelles on trouvera notamment de quoi calculer la valeur de la résistance associée à une diode zener, à un régulateur, à un diviseur, ou un multivibrateur astable, mais aussi le code de couleur de la résistance et de l'inductance. Avec ce CD-ROM, vous disposez donc de données fiables sur plus de 7.800 circuits entiers ; plus de 35.600 transistors, FET, thyristors et triacs ; environ 25.000 diodes et plus de 1.800 optocoupleurs. Le clou, c'est que vous allez pouvoir rajouter dans la base de données ce qui y manque encore, car elle est interactive ! Ainsi chaque utilisateur pourra lui-même rajouter des composants, en modifier les caractéristiques déjà enregistrées ou les compléter.

ISBN 978-90-5381-298-3 • 29,50 €

elektor Pour commander en ligne : www.elektor.fr/ecd7



elektor labs

est ouvert
24 heures sur 24
7 jours sur 7

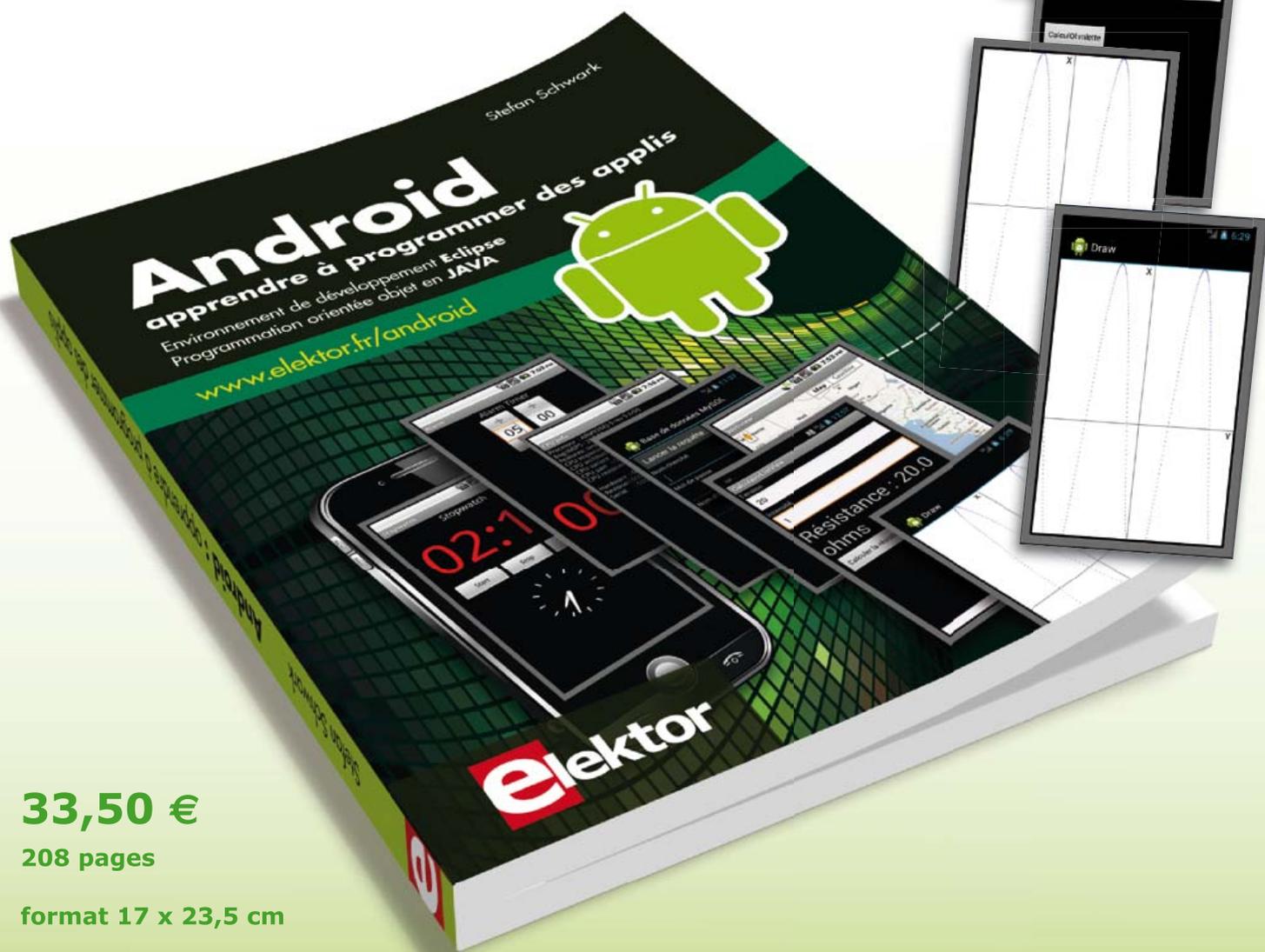
Participez à l'élaboration des
projets sur
www.elektor-labs.com



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

Pour tout type de mesure, faites confiance au leader



Les matériels d'acquisition de données de NI sont réputés pour être dignes de confiance, et faire référence en matière de performances et de précision en particulier. Les systèmes NI CompactDAQ peuvent intégrer plus de 50 modules de mesure spécialisés et un très large éventail d'E/S. Conjugée aux capacités d'analyse et de traitement du signal du logiciel NI LabVIEW, cette plate-forme met à votre disposition la puissance de la conception graphique de systèmes pour vous permettre de composer plus rapidement tout système de mesure.

LabVIEW offre de puissantes fonctions d'analyse pour les signaux et données provenant de capteurs, simplifie la création d'interfaces utilisateur et vous permet de programmer comme vous pensez : graphiquement.



>> Boostez votre productivité sur ni.com/measurements-platform/f

01 57 66 24 24

NATIONAL INSTRUMENTS France • 2 rue Hennape – 92735 Nanterre Cedex, France • Tél. : (01) 57 66 24 24 • Fax : (01) 57 66 24 14 • Société de droit américain – capital social 1.000,00 dollars • US – 11500 N Mopac Expwy, Austin-Texas USA – 10056236 – 344 497 649 • RCS Nanterre – SIRET B 344 497 649 00022 – APE 516J - N.I.I. FR 57344497649

©2012 National Instruments. Tous droits réservés. LabVIEW, National Instruments, NI, ni.com, et NI CompactDAQ 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. 08567

**NATIONAL
INSTRUMENTS™**