

92 pages : 8 pages supplémentaires !

avril 2015 | n° 442
www.elektor-magazine.fr

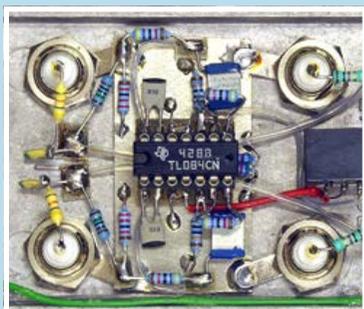
Elektor

DÉCOUVRIR • CRÉER • PARTAGER



CODEBENDER

ARDUINO sur l'internet



détecteur de cliquetis pour moteurs



ROBOT aérateur PLATINO régule l'humidité



convertisseur USB ↔ série multiprotocole avec FT232H

dans ce numéro :
3 projets elektor labs
6 modules & CI
1 cours de programmation
4 projets de lecteurs
1 banc d'essai et bien davantage

ARM pour néophytes 3^e p. : l'art des temporisateurs • **DesignSpark :** cartes multicouche • **analyseur UART/RS232** • **drôle de composant :** gyrateur intégré TCA580 • **galvanomètre à aiguille** commandé par ATtiny2313 • **étiquettes antivol pour contrôle d'accès** sous Bascom • **module Bluetooth Low Energy e-BoB BL600 2^e partie** • **Arduino en émetteur AM** • **pilote de LED** bon marché à haut rendement • **l'Internet des Objets ? c'est WunderBar !** • **rétronique**

ISSN 0181-7450

(B) 7,90 € • (CH) 13,20 FS • (CAN) 9,75 CAD • DOM surface 8,20 e • DOM avion 10,20 €
TOM surface 940 XPF • TOM avion 1620 XPF • Réu/A 9,70 € • Cal/S 950 CFP • Cal/A 1560 CFP • And 7,20 €

L 19624 -442- F: 7,80 €-RD



400 000 références pour vos réalisations techniques avec Conrad

Mesure

Outils
Soudage

Kits et
Modules

Composants
Électroniques

Automatisme

Pile / batterie
alimentation

La robotique est à l'honneur du 1er au 30 avril chez Conrad !

Robotique (n.f.) :
Rêve d'enfant,
projet d'avenir !



Partenariats Conrad

Conrad, distributeur exclusif du kit Wunderbar

Le kit de développement Wunderbar a décroché la première place du Grand Challenge Internet Of Things organisé par Cisco Systems, pour son innovation et son design.



Conrad, partenaire officiel des Trophées de la Robotique 2015.

Rendez-vous les 2 et 3 mai
à Saint-Quentin !



Offre Spéciale Conrad

-20%⁽¹⁾

sur **TOUTE** votre commande

+

0€⁽²⁾

Livraison **GRATUITE** de votre colis

code avantage particuliers sur www.conrad.fr **ELEK** - code avantage professionnels sur www.conradpro.fr **ELEKPRO**

Offre de bienvenue réservée aux 500 premiers clients.

CONRAD
Partenaire de toutes vos idées

(1) 20% sur toute commande. Offre uniquement valable en France métropolitaine. Hors articles ayant un prix barré, signalés « non remisable » ou marqués d'un astérisque (*), hors librairie et offres spéciales, hors Eco-part. Uniquement en vente à distance. Une seule offre par entreprise / foyer. (2) Forfait livraison standard offert sous forme de remise répartie sur l'ensemble de vos articles à partir de 79€ d'achats HT sur www.conradpro.fr / 49€ d'achats TTC sur www.conrad.fr. Hors supplément de livraison. Offre uniquement valable en France métropolitaine. Offre non cumulable.

DROITS D'AUTEUR :

© 2015 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'œuvre 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.

Imprimé aux Pays-Bas
par Senefelder Misset - Doetinchem
Distribué en France par M.L.P.
et en Belgique par A.M.P.



avis de recherche

Vous êtes passionné d'électronique, de micro-informatique et de communication ?

C'est peut-être de vous que l'équipe française d'Elektor a besoin pour renforcer sa présence dans le monde francophone. Lu par les professionnels passionnés et par les amateurs chevronnés, Elektor est la source d'inspiration par excellence de tous les électroniciens curieux de l'évolution vertigineuse de leur discipline.

Pour se développer en France, Elektor recherche des électroniciens créatifs et enthousiastes, susceptibles de coopérer à distance avec Elektor depuis leur lieu de résidence actuel en tant que **producteurs et éditeurs**.

Vous peut-être ? Vos quatre tâches essentielles pourraient être :

- chercher des auteurs de livres et d'articles sur les développements de l'électronique et, plus généralement, des auteurs de projets électroniques publiables ;
- établir le contact avec des entreprises, des institutions scolaires, universitaires ou de formation professionnelle, pour envisager avec elles des coproductions possibles ;
- développer la diffusion des publications d'Elektor auprès de nouveaux lecteurs et de membres ;
- recruter, superviser et assister des concepteurs et des auteurs francophones pour le site www.elektor-labs.com.

Cette liste n'est pas restrictive, car ce sont vos talents que nous souhaitons voir mûrir. Cette activité conviendrait p. ex. à des enseignants ouverts à de nouveaux défis, à des étudiants fraîchement diplômés et avides de mettre en application leurs connaissances, mais pourrait intéresser aussi des vendeurs/acheteurs formés dans l'industrie de l'électronique et plus généralement toute personne passionnée par l'électronique disposée à s'engager sous la nouvelle bannière d'Elektor : **découvrir, créer, partager**

Votre français est impeccable et vous êtes à l'aise en anglais, en allemand ou en néerlandais. Vous sentez que la cause d'Elektor pourrait devenir la vôtre ?

Adressez-moi votre candidature avec CV sans tarder : d.meyer@elektor.fr

Denis Meyer

Notre équipe

Rédacteur en chef :	Denis Meyer (redaction@elektor.fr)
Rédaction internationale :	Harry Baggen, Jan Buiting, Jaime González-Arintero, Jens Nickel
Laboratoire :	Thijs Beckers, Ton Giesberts, Luc Lemmens, Clemens Valens (responsable), Jan Visser
Coordination :	Hedwig Hennekens
Ont coopéré à ce numéro :	Robert Grignard, Hervé Moreau, Kevin Petit, Guy Raedersdorf, Mariline Thiebaut-Brodier
Service de la clientèle :	Cindy Tyssen
Graphistes :	Giel Dols, Mart Schroijen, Patrick Wielders
Elektor en ligne :	Daniëlle Mertens

- 6 **Elektor : votre guide de connexion(s)**
- 30 **Rési&Transi ne feront plus de bulles**
- 32 **pilote de LED bon marché à haut rendement**
0,5 A/33 V
- 44 **e-choppe : notre boutique en ligne**
- 88 **Elektor World**
Le monde d'Elektor - Elektor dans le monde
- 90 **hexadoku**
votre bain parfumé à la fleur d'hexadécimal !

DÉCOUVRIR CRÉER PARTAGER

- 8 **découvrir :**
choses bêtes et bêtes choses
- 9 **trucs & astuces entre lecteurs**
- 10 **microcontrôleur ARM pour néophytes (3^e)**
pour passer de 8 bits à 32 bits
- 18 **DesignSpark 19e (et dernier) jour :**
les cartes multicouche
- 20 **l'Internet des Objets ?**
c'est WunderBar !
- 23 **gyrateur intégré TCA580**
drôle de composant n°14
- 24 **Codebender**
Arduino sur l'internet
- 28 **détecteur de cliquetis**

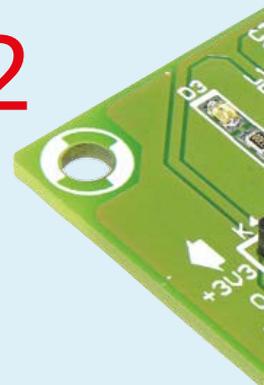
DÉCOUVRIR CRÉER PARTAGER

- 35 **bienvenue dans la section des créateurs**
- 38 **analyseur UART/RS232**
tenir à l'œil le flux sériel
- 50 **robot aérateur de cave**
Platino régule l'humidité
- 56 **galvanomètre à aiguille**
commandé par microcontrôleur
- 59 **étiquettes antivol**
pour contrôle d'accès sous Bascom

analyseur UART/RS-232

pour tenir à l'œil le flux sériel

Pour l'échange de données entre appareils, on privilégie l'UART. Cette interface est d'usage facile et quantité de puces de convertisseurs peuvent s'y associer. Mais si, en cours de développement ou de test, vous apercevez que quelque chose cloche dans la communication, vous pouvez vous attendre à un dépannage épineux. Sauf si vous avez notre analyseur sériel UART qui vous simplifie la vie en enregistrant les octets dans les deux sens à la milliseconde près. Des raccordements variés et une séparation galvanique lui permettent de s'insinuer sans risque dans toute liaison sérielle à niveaux TTL ou RS232.



34

Arduino sur l'Internet CODEBENDER

L'environnement de développement (IDE) de Massimo Banzi n'est pas du goût de tout le monde. En 2012, un groupe de développeurs grecs créa - surtout pour le plaisir - un IDE Arduino exécutable dans un navigateur. Au fil du temps il s'est révélé être une alternative sérieuse.

24

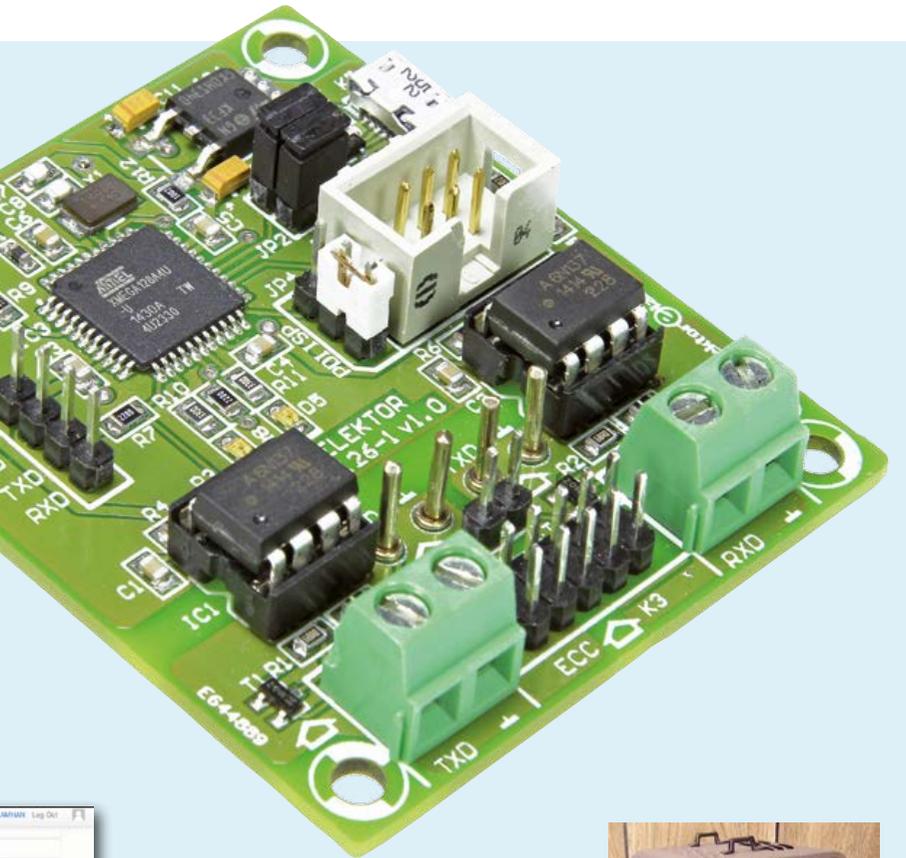


Arduino en émetteur sur P.O. par induction

Les radios à tubes possèdent un charme particulier. Non seulement elles sont de toute beauté, mais elles ont un son hors pair et, de plus, rappellent souvent des souvenirs agréables. D'où le nombre de fans qui collectionnent, réparent ou restaurent avec passion ces antiquités. Malheureusement, de moins en moins de stations (radio) émettent sur les bandes de MA (Modulation d'Amplitude) classiques.

67

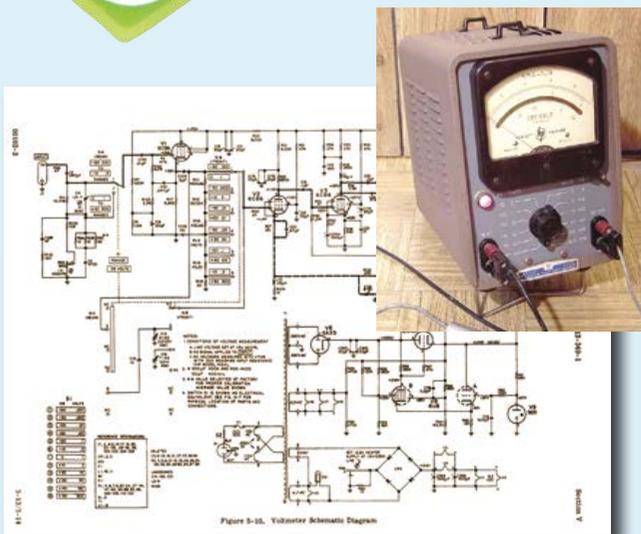




- 62 **e-BoB BL600 2^e partie**
Éditer, compiler, transférer un programme avec le module *Bluetooth Low Energy*
- 67 **Arduino en émetteur AM sur P.O.**
par induction
- 70 **convertisseur USB ↔ série**
multiprotocole avec FT232H

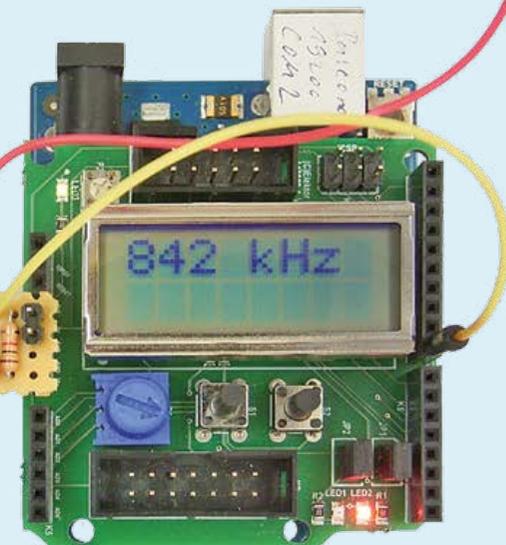
DÉCOUVRIR **CRÉER** **PARTAGER**

- 77 **l'horloge comme métaphore du partage**
- 78 **Rétronique**
restauration d'un voltmètre à tubes (2)
- 82 **épluchage de pomme**
good Job
- 85 **LABorama**
Big Brother pourra jouer du tambour pour Spock
- 86 **mieux mesurer à l'oscilloscope**
des bases aux astuces



Rétronique restauration d'un voltmètre à tubes

78



bientôt sur ces pages

Lecture d'étiquettes NFC
Expérimentations avec la communication en champ rapproché à l'aide d'une puce NFC de NXP (NTAG I2C) et d'un ATmega328P. Pour programmer et lire les étiquettes, il suffit d'un téléphone tactile compatible NFC.

Sonnette de porte d'entrée programmable
Les deux parties de ce circuit (le bouton et la sonnette proprement dite) communiquent sans fil. Des interrupteurs DIP permettent de choisir une des mélodies disponibles.

Mesure de résistance avec Arduino
Circuit expérimental pour mesurer des résistances entre 1 kΩ et 1 MΩ à l'aide d'un Arduino et de l'interface Elektor. Associé à un capteur d'humidité, c'est une application intéressante.

Sous réserve de modification.
Le numéro de mai paraîtra le 25 avril.

Elektor : votre guide

Elektor, c'est bien plus qu'un simple magazine. C'est une communauté d'électroniciens, du débutant au professionnel, désireux d'apprendre, de concevoir et de partager une électronique qui frappe.

56

pays

246497

membres actifs

10

experts &



elektor.post

L'hebdo d'Elektor

Les signaux du changement fusent autour de nous. Avec son rythme hebdomadaire, la lettre électronique elektor.post permet de les suivre à une allure soutenue. Une semaine sur deux, un montage inédit !

www.elektor.com/newsletter



communauté Elektor

Devenez membre, Green ou Gold

Pro ou débutant, rejoignez la communauté. C'est le plus sûr moyen de ne rien rater, ni en électronique classique, ni en techniques embarquées modernes. Vos atouts : accès direct à elektor.labs, forums, lettres d'information hebdomadaires, projets inédits bimensuels, offres exceptionnelles, archives, moteurs de recherche. Les formules Green et Gold donnent droit à de nombreux avantages : **GREEN**, c'est le magazine sous forme numérique, sans papier. **GOLD**, c'est la formule complète avec la version imprimée.

www.elektor.com/memberships



elektor.TV

Pour y voir plus clair

L'image vidéo filmée sans façon est devenue un rival stimulant pour le texte typographié et mis en page ! Les anecdotes visuelles ne manquent pas dans la vie d'un labo d'électronique, surtout quand ça commence à fumer. Et souvent trois plans filmés remplacent efficacement de longs discours. Regardez elektor.tv !

www.youtube.com/user/ElektorIM



Elektor PCB Service

Des cartes à la carte

Adieu perchlore de fer, bienvenue aux magnifiques circuits imprimés du service de gravure à la demande d'Elektor. Très professionnel et très rapide, très avantageux et tellement plus commode !

www.elektorpcbsevice.com



elektor.labs

découvrir, créer & partager

Au cœur de la matrice, **elektor.labs** (le labo d'Elektor) est ouvert et accessible à tous, c'est l'incubateur où éclosent les circuits. Petits et grands, analogiques ou numériques, d'avant-garde ou nostalgiques, ils y sont tous transformés en matière première raffinée, prête à l'emploi, testée et documentée pour vos propres créations.

www.elektor-labs.com



elektor.academy

À cheval sur la courbe d'apprentissage

Webinaires, séminaires, cours, présentations, ateliers, lectures, formation en entreprise sont quelques-unes des méthodes pédagogiques utilisées par Elektor pour diffuser la connaissance de l'électronique à tous les niveaux aussi bien parmi les professionnels passionnés que pour les amateurs motivés.

www.elektor-academy.com

de connexion(s)



auteurs publications visiteurs (mois) date de référence

elektor.magazine
Plus de 500 pages d'électronique inédite chaque année

Le magazine est le vaisseau amiral, affrété tous les mois par la rédaction internationale d'Elektor pour vous embarquer vers des contrées électroniques nouvelles. Chaque édition, sur papier ou en format numérique, ne se contente pas de rester à la hauteur des précédentes, mais cherche à les surpasser.

www.elektor-magazine.com

e-choppe Elektor en ligne
Votre panier d'achats pour l'électronique

Le magazine et le labo d'Elektor proposent, en coopération avec des partenaires choisis, des produits et des services de haut niveau. Notre e-choppe, véritable caverne d'Ali Baba, est ouverte toute l'année sans interruption pour les électroniciens du monde entier.

www.elektor.fr

3 formules pour rester connecté avec Elektor !

livres et DVD Elektor

La puissance de l'information

Elektor aborde tous les domaines de l'électronique : de la programmation des 8 bits aux ARM, des antennes aux diodes zener, des µC aux tubes... Nos ouvrages font référence et autorité dans le monde entier, aussi bien pour les techniques classiques que pour les innovations les plus récentes. Lire pour (mieux) comprendre.

www.elektor.fr

Formule GREEN 92,50 € par an	Formule GOLD 127,50 € par an	Formule gratuite !
10 x magazine imprimé	10 x magazine imprimé	10 x magazine imprimé
10 x magazine numérique	10 x magazine numérique	10 x magazine numérique
l'accès à l'archive d'Elektor	l'accès à l'archive d'Elektor	l'accès à l'archive d'Elektor
10 % de remise dans l'e-choppe	10 % de remise dans l'e-choppe	10 % de remise dans l'e-choppe
l'accès à Elektor.labs	l'accès à Elektor.labs	l'accès à Elektor.labs
un DVD annuel	un DVD annuel	un DVD annuel
des offres exclusives	des offres exclusives	des offres exclusives
26 nouveaux projets inédits	26 nouveaux projets inédits	26 nouveaux projets inédits
www.elektor.fr/formule-greencard	www.elektor.fr/formule-goldcard	www.elektor.fr/inscription


Jens Nickel

choses bêtes et bêtes choses

Rédiger un article pour Elektor implique d'inlassables vérifications, aussi celle du bon fonctionnement de la carte décrite. Pour cela j'ai donc emprunté à mon collègue Ton du labo **l'enregistreur de données UART/RS-232** que je décris dans ce numéro.

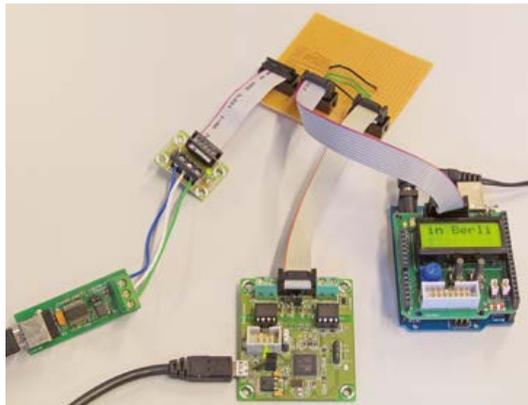
Déconvenue en reliant le prototype à mon ordinateur : Windows 7 affiche bizarrement : *Périphérique inconnu* au lieu de détecter si le pilote nécessaire a été installé ou pas sur mon ordinateur. Dans Google je tape « USB CDC » (la classe CDC établit une communication série entre un microcontrôleur

et un PC via USB) ainsi que quelques mots-clés exprimant des problèmes possibles avec Xmega et Windows 7. Il n'en sort rien hélas qui éclaire ma lanterne.

Penaud, je rapporte la carte à Ton en lui jurant que – cette fois-ci – je n'avais pas tripatouillé le circuit. Il est vrai que peu de temps auparavant, j'avais relié au mauvais connecteur l'alimentation d'un module capteur de température, mais oublions cette lamentable histoire...

Quand mon indulgent collègue relie la carte et le câble à son ordinateur, je vois par dessus son épaule s'afficher le même *Périphérique inconnu*. Un ange passe. Après quelques secondes d'intense réflexion, l'un de nous deux, je ne dirai pas lequel, a eu l'idée de tenter le coup avec un autre câble USB. Et tout est rentré dans l'ordre. Bah, pas grand-chose en fin de compte, mais une leçon utile : **les gens malins pensent aussi aux pannes les plus idiotes**. Pas de micrologiciel capricieux ici, ni de « problème Atmel », encore moins de concours de circonstances fâcheux avec Windows, mon câble USB était tout bonnement foutu.

Pionnier chez Elektor pour tout ce qui concerne les systèmes modulaires, je voulais tester avec une liaison ECC l'enregistreur dont je parle ci-dessus et qui marche donc très bien. J'avais prévu d'écrire un encadré à ce sujet.



Sur mon bureau, traînent en permanence une carte Arduino Uno et un *shield* d'extension auquel est relié (via le connecteur ECC) un convertisseur RS-485. L'ensemble communique avec mon PC par RS-485 et un convertisseur RS-485/USB.

Mes collègues avaient entre-temps soudé une petite carte de répartition en Y pour les signaux ECC, ce qui m'a permis d'insérer l'enregistreur entre le *shield* et le convertisseur RS-485 (voir photo). J'ai alors installé mon programme de commande, celui qui me permet par exemple d'allumer ou d'éteindre des LED depuis un ému-

lateur de terminal au moyen de simples commandes ASCII. J'ai ensuite ouvert une seconde instance d'HTerm, configuré l'enregistreur sur le bon débit, puis suis revenu au premier terminal pour envoyer ma commande (`L 0 0 +<CR>`) à la carte Uno. Dans la seconde fenêtre, j'ai reçu comme seul écho :

```
OK_{cr}
```

Autrement dit la réponse de la carte Arduino à l'envoi de l'instruction pour commuter la broche. Rien de la commande ASCII elle-même. Pourquoi l'enregistreur ne fonctionnait-il que dans un sens ? Une mauvaise soudure sur la carte de répartition ? J'ai inversé le sens du connecteur ECC sur l'enregistreur pour voir si cela changeait quelque chose. J'ai alors vu apparaître la commande du PC (`L_0_0_{CR}`) mais plus la réponse d'Arduino. Quelque chose clochait au niveau de l'enregistreur. Une soudure sèche ? En examinant la carte de près, j'ai fini par découvrir l'absence du cavalier pour relier les masses des deux signaux. Le schéma de l'enregistreur est explicite : la masse de l'ECC n'est relayée que sur un seul canal.

Et dire que je suis le signataire d'un avertissement écrit pour la série *Le bus arrive*, article 5, sous le titre *Grounding for Dummies* ?

Dummiie reprendra encore, tiens !

(150019 - version française : Hervé Moreau) ◀

trucs & astuces entre lecteurs

compilés par **Burkhard Kainka**

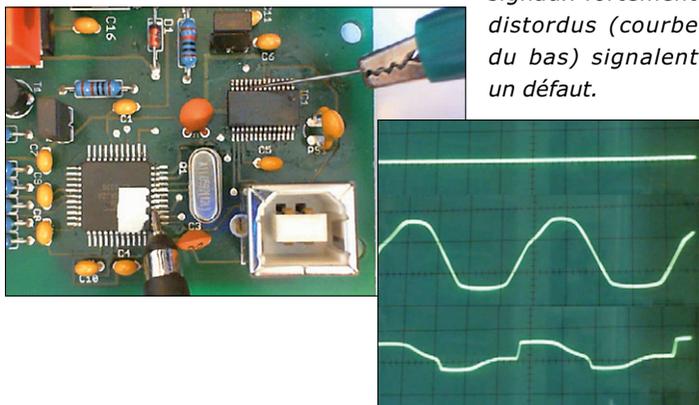
Voici de nouvelles solutions futées pour faciliter votre vie d'électronicien.



Oscilloscope en testeur de continuité

Burkhard Kainka

Dépanner des cartes dotées de CMS juste avec un ohmmètre n'est pas facile. Voici une nouvelle approche qui fait appel à un générateur sinusoïdal et un oscilloscope. Avec la sonde de l'oscillo, on touche la broche dont on souhaite vérifier la connexion. On passe la pointe du générateur de signal sur les broches d'un autre circuit intégré avec lequel une connexion devrait exister. Même du coin de l'œil, on distingue aussitôt une absence totale de connexion d'une bonne ou d'une mauvaise connexion. L'oscillogramme du milieu indique une bonne connexion : même si les alternances inférieures de ce sinus, qui provient de diodes de protection CMOS, sont cabossées, c'est bon signe. Attention : avec cette méthode, la masse doit être connectée correctement. Des signaux fortement distordus (courbe du bas) signalent un défaut.



résistance additionnelle. Il fonctionne par le biais de la résistance de la peau ! Avec un doigt, on touche le pôle positif de la pile et l'anode de la diode, tandis que l'on maintient la cathode de la LED directement sur le pôle négatif de la pile. La photo illustre ce « tour de main ». Dans le cas d'une résistance épidermique de l'ordre de 1 M Ω , le courant de LED atteint quelques μ A.

Commuter un port AVR

Truc d'Andreas Riedenauer (Ineltek Mitte GmbH)

Lorsqu'on écrit dans un registre de type PIN, la broche de port correspondante commute (toggle). Cette action est très rapide. La capture d'écran montre un test en Bascom avec l'ATmega328 de l'Arduino Uno : ça marche ! Idem avec l'ATtiny13, vérification avec un programme de clignotant.

```
'LEDblink.BAS
-----
$regfile = "m324pdef.dat"           'ATmega328p
$crystal = 16000000                 '16 MHz

-----
Config PORTB = Output

'13.2.2 Toggling the Pin
'Writing a logic one to PINxn toggles the value of PORTxn.
'Independent on the value of DDRxn.
'Note That The Sbi Instruction Can Be Used To Toggle One Single Bit In A Port.

Do
  PINB 5 = 1                         'B5 toggeln
  Waitas 500
Loop

Do
  PORTB 5 = 1                         'LED on
  Waitas 500                          '500 ms
  PORTB 5 = 0                         'LED off
  Waitas 500                          '500 ms
Loop
```

Testeur de pile simple

D'après une idée d'Eckhard Koch

Les LED vertes les plus récentes fonctionnent à des courants inférieurs au micro-ampère (μ A). Cela ouvre de nouvelles perspectives. Un testeur de pile peut ainsi se passer de toute



Ce n'est qu'après de longues recherches qu'on le découvre dans la feuille de caractéristiques (ATtiny13 par ex.) : **The Port Input Pins I/O location is read only, while the Data Register and the Data Direction Register are read/write. However, writing a logic one to a bit in the PINx Register, will result in a toggle in the corresponding bit in the Data Register.** ◀

(150027)

Vous avez une solution intelligente pour un problème vraiment délicat... Vous mettez en œuvre de façon non conventionnelle un composant ou un outil... Vous savez comment résoudre un problème plus facilement ou mieux qu'avec la solution actuelle... Écrivez-nous – chaque astuce publiée vous rapportera 40 € !



L'art des temporisateurs



µC ARM pour néophytes pour passer de 8 bits à 32 bits

3^e partie

Viacheslav Gromov (Allemagne)

Même et surtout dans le monde numérique, le *timing* est d'une importance capitale. C'est pourquoi cet article portera sur les temporisateurs (*timer*) du SAM D20, du chien de garde (TCG) à l'horloge en temps réel (RTC en anglais). Nous aborderons d'autres sujets tels que la production du signal d'horloge et les interruptions externes, en utilisant, comme toujours, de petits exemples pratiques de code. Ainsi vous ne perdrez pas votre temps ;-)

Dans l'article précédent, SERCOM, l'unité de communication a été évoquée, mais nous n'en avons pas encore essayé certaines interfaces. Avant de poursuivre dans cette direction, il est opportun de jeter un coup d'œil appuyé aux différents temporisateurs du microcontrôleur (µC). Ceux-ci seront abondamment mis à contribution dans les prochains articles.

Les listages des projets ne sont pas reproduits in extenso dans la revue, mais vous retrouverez les parties essentielles du code relatif à cet article dans le fichier « Listings » téléchargeable en [1]. Les projets Atmel Studio complets se trouvent dans l'archive Zip. Le matériel nécessaire pour cet article est le même que celui de l'article du mois dernier, à ceci près que les boutons et les LED sont connectés un peu différemment à la carte. Il suffira de relier le circuit de la **figure 1** à la carte SAM D20 Xplained Pro et déjà nous pourrons nous y mettre !

Le TCG, la production du signal d'horloge et le CIE

Le Temporisateur Chien de Garde (TCG = WDT pour *Watchdog Timer* en anglais) est le temporisateur le plus simple du SAM D20. Il nous permet une entrée matière sans douleur comme le montre la **figure 2**, bien qu'il offre plus de fonctions que le TCG d'un µC 8 bits ordinaire. Un TCG a pour tâche de redémarrer un programme si ce dernier ne s'exécute pas dans le temps imparti (plantage). On voit sur le synoptique qu'il existe un registre de comptage *COUNT*, capable ici de compter jusqu'à 32768, et au-dessous, d'autres registres pour la comparaison de la valeur actuelle du compteur. En cas de concordance, cela se traduira par une EWI (*Early Warning Interrupt* = Interruption de Pré-Alerte, IPA), voire une réinitialisation (*Reset*) du microcontrôleur. Il faut donc que le programme réinitialise toujours le TCG à temps pour éviter un débordement du compteur et l'activation d'un redémarrage (*Reset*).

Dans le cas du SAM D20, le TCG connaît deux modes :

- En mode normal, on paramètre une valeur de débordement comprise entre 8 et 16384, pour laquelle il y a réinitialisation et remise à zéro du compteur. On peut, dans la même plage, paramétrer l'IPA qui délivre une interruption d'alerte juste avant le débordement.
- En mode Fenêtre, le programme ne peut réinitialiser le compteur qu'à l'intérieur d'une fenêtre temporelle bien définie. Si le programme réinitialise le compteur du TCG hors de la fenêtre temporelle, on a redémarrage du µC. On pourra choisir, pour la fenêtre temporelle « ouverte », un état de compteur entre 8 et 16384. La fenêtre fermée commence à partir de cet état de compteur ; elle aussi peut durer entre 8 et 16384 cycles de compteur.

Le TCG est cadencé via un multiplexeur par l'un des huit générateurs du contrôleur GCLK (*Generic CLock* = Horloge Générique), voir le synoptique de la **figure 3**. Il comporte huit générateurs qui peuvent être eux cadencés par divers oscillateurs internes et externes, voire par d'autres générateurs. Chacun des générateurs contient, entre autres, un prédiviseur (*prescaler*) pour diviser l'horloge de l'oscillateur.

Le CIE (*Contrôleur d'Interruption Externe*, EIC pour *External Interrupt Controller*) de la **figure 4** de notre SAM D20 travaille sur seize broches. Le contrôleur filtre numériquement le signal présent sur les broches sélectionnées et l'évalue. Il peut donc, en fonction de la configuration, flanc descendant ou flanc montant sur la broche concernée, déclencher des interruptions et, par exemple, tirer le µC de son sommeil. En plus d'autres options, on peut aussi détecter une impulsion.

Le CIE travaille avec une autre broche encore, capable de délivrer des interruptions non masquées qui peuvent elles, sous la houlette du NVIC (*Nested Vectored Interrupt Controller*) même (en fonction des priorités) interrompre d'autres interruptions.

Premiers pas avec le TCG

Essayons le TCG en mode normal. Grâce à sa structure assez simple, il convient parfaitement pour le premier projet de temporisateur. Pas de panique, les bibliothèques bien faites de l'ASF (Cf. article précédent) rendent facile la commande d'un temporisateur. Le programme *First Project with WDT* de l'archive Zip est aisé à comprendre. Tout en haut, après la déclaration des variables et des prototypes de fonction, on trouve trois fonctions Callback. La première *watchdog_early_warning_callback* est appelée lors de l'IPA du TCG et ne fait rien de plus que commuter, par le biais d'une instruction de port, la LED0 sur la carte.

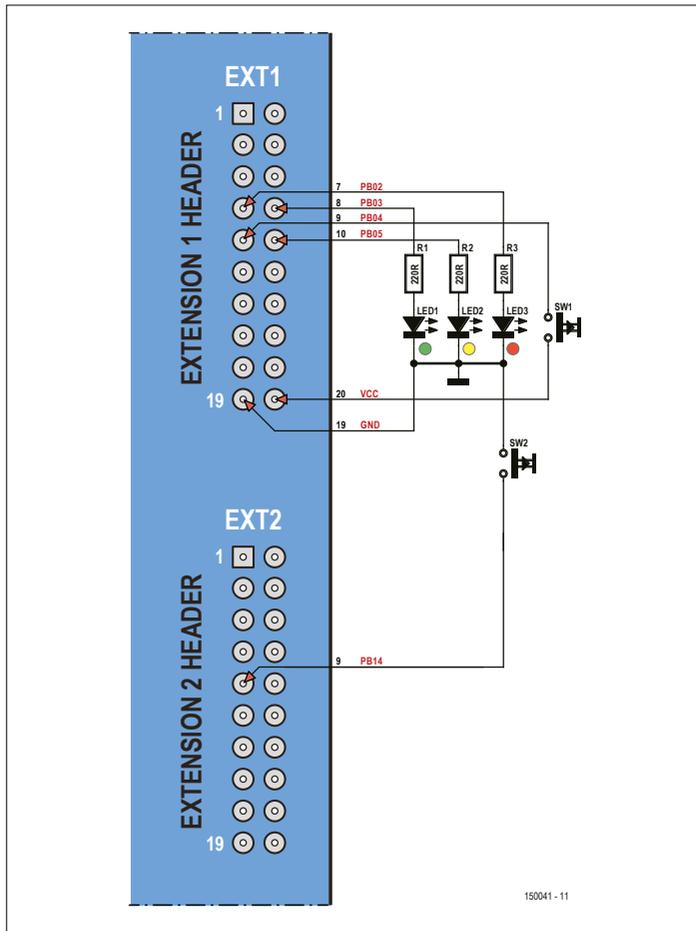


Figure 1. Le circuit a légèrement changé par rapport à sa version de l'article du mois dernier.

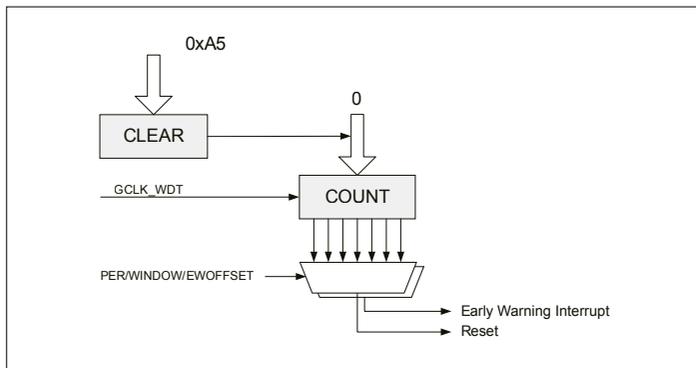


Figure 2. Synoptique du temporisateur de chien de garde (source des captures d'écran et synoptiques : Atmel).

Le bouton SW1 sur la broche d'interruption 4 appelle la fonction *extint4_detection_callback*. La variable booléenne *i* s'y voit attribuer la valeur 1. La dernière fonction Callback *extint14_detection_callback* est appelée par le bouton SW2 (broche d'interruption 14). Elle donne la valeur 0 à *i*. On a ensuite les deux fonctions de configuration importantes du TCG et du CIE que l'on retrouve dans le listage 1.

La configuration du TCG suit le modèle connu : on commence par créer une structure de configuration, ici *config_wdt*, initialisée à l'aide des paramètres *wdt_get_config_defaults(&config_wdt)*. On trouve ensuite des paramètres spécifiques des variables de la structure de configuration. La variable *always_on* est mise à *false* (si elle avait été mise à *true*, le TCG, étant auparavant toujours actif et ne pouvant qu'être remis à zéro, n'aurait pu être configuré qu'à la prochaine remise en fonction). On a ensuite, par la variable de structure *clock_source*, sélection de la source d'horloge (GCLK Generator 4) et paramétrage de la valeur de débordement ou de préalerte à 8192 et 4096 cycles d'horloge respectivement. Comme la valeur de la préalerte est exactement la moitié de la valeur de débordement, la LED clignotera régulièrement. À la fin de cette fonction, la commande *wdt_set_config(&config_wdt)* transmet les valeurs de la structure de configuration au TCG. Selon le même modèle, la fonction suivante paramètre le CIE. On obtient une structure de configuration appelée *config_extint_chan* dans laquelle on charge les paramètres. On a ensuite deux blocs de code quasi similaires. Dans le premier, les variables de la structure de configuration règlent les caractéristiques de la broche PB04 (raccordée au bouton SW1) : type, multiplexeur, résistance de polarisation et critère de détection. Comme SW1 est actif au niveau haut, une résistance de polarisation au niveau bas assure l'état bas du repos. Le critère de détection doit être un flanc montant (bas/haut). À la fin de ce bloc, il faut à nouveau, par le biais de *extint_chan_set_config(4, &config_extint_chan)*, transmettre au CIE les paramètres prévus. Il suffit de transférer à cette fonction le numéro de l'interruption externe (4 ici) et un pointeur vers la structure de configuration. Le second bloc de code de configuration du CIE a la même structure, mais concerne la broche PB14 (interruption externe 14) reliée au bouton SW2. Comme cette broche fonctionne dans l'autre sens, active au niveau bas, il faut prévoir une résistance de polarisation au niveau haut et le CIE doit réagir non pas à un front montant, mais sur front descendant.

On trouve ensuite les fonctions appelées qu'une seule fois dans le programme ainsi que les fonctions d'interruption (Callback) qui enregistrent et activent respectivement le TCG et le CIE. La première fonction appelée *configure_wdt_callbacks* ne contient que deux lignes de code :

```
wdt_register_callback(watchdog_early_warning_callback, WDT_CALLBACK_EARLY_WARNING);
wdt_enable_callback(WDT_CALLBACK_EARLY_WARNING);
```

La première instruction, qui enregistre la fonction Callback *watchdog_early_warning_callback*, ne requiert, comme second paramètre, que l'événement devant provoquer l'appel de la fonction. La seconde instruction qui active la fonction de Callback enregistrée auparavant ne comporte que la nature de l'événement.

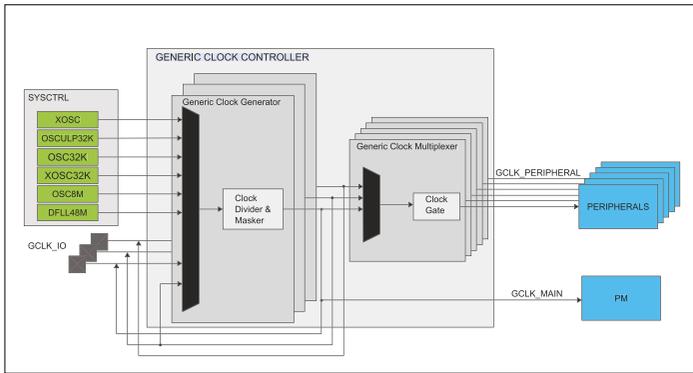


Figure 3. Synoptique du contrôleur GCLK (*Generic Clock*).

La fonction d'enregistrement et d'activation des Callbacks du CIE est similaire, mais traite, elle, deux interruptions (EXTINT4 et EXTINT14). Les instructions qu'elle contient sont semblables à celles de la dernière fonction, il leur faut le numéro de l'interruption externe souhaitée. Le fichier *Main* ne fait rien de plus que d'appeler les fonctions de configuration et mettre LED0 au niveau bas. La boucle sans fin se limite à des interrogations

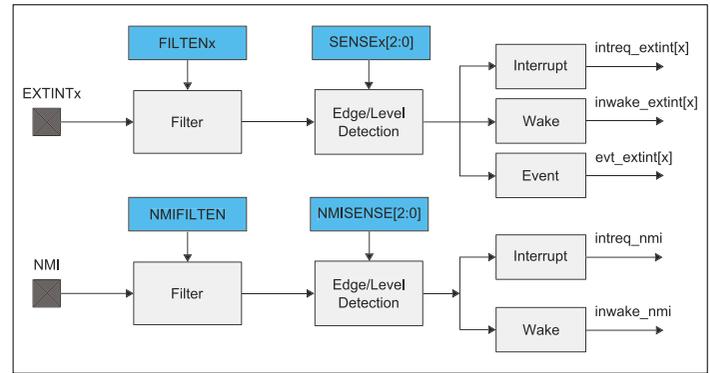


Figure 4. Ce synoptique est celui d'un CIE (Contrôleur d'Interruption Externe, EIC pour *External Interrupt Controller* en anglais).

pour savoir si $i = 1$. Si oui, on a, par le biais de `wdt_reset_count()`, remise à zéro du TCG.

Avant de compiler ce programme, jetez un coup d'œil au fichier `conf_clocks.h` (sous `src/config`). Il n'est pas difficile de modifier les paramètres d'horloge utilisés. Pour ce projet, on a modifié les paramètres du générateur GLCK 4 chargé du cadencement

Listage 1. Les fonctions de configuration du WDT et de l'EIC.

```
void configure_wdt(void)
{
    struct wdt_conf config_wdt;
    wdt_get_config_defaults(&config_wdt);
    config_wdt.always_on = false;
    config_wdt.clock_source      = GCLK_GENERATOR_4;
    config_wdt.timeout_period   = WDT_PERIOD_8192CLK;
    config_wdt.early_warning_period = WDT_PERIOD_4096CLK;
    wdt_set_config(&config_wdt);
}

void configure_extint_channels(void)
{
    struct extint_chan_conf config_extint_chan;
    extint_chan_get_config_defaults(&config_extint_chan);

    config_extint_chan.gpio_pin = PIN_PB04A_EIC_EXTINT4;
    config_extint_chan.gpio_pin_mux = MUX_PA04A_EIC_EXTINT4;
    config_extint_chan.gpio_pin_pull = EXTINT_PULL_DOWN;
    config_extint_chan.detection_criteria = EXTINT_DETECT_RISING;
    extint_chan_set_config(4, &config_extint_chan);
    config_extint_chan.gpio_pin = PIN_PB14A_EIC_EXTINT14;
    config_extint_chan.gpio_pin_mux = MUX_PA14A_EIC_EXTINT14;
    config_extint_chan.gpio_pin_pull = EXTINT_PULL_UP;
    config_extint_chan.detection_criteria = EXTINT_DETECT_FALLING;
    extint_chan_set_config(14, &config_extint_chan);
}
```

Listage 2. Les paramètres du générateur GLCK numéro 4.

```
# define CONF_CLOCK_GCLK_4_ENABLE      true
# define CONF_CLOCK_GCLK_4_RUN_IN_STANDBY false
# define CONF_CLOCK_GCLK_4_CLOCK_SOURCE SYSTEM_CLOCK_SOURCE_OSC8M
# define CONF_CLOCK_GCLK_4_PRESCALER   1024
# define CONF_CLOCK_GCLK_4_OUTPUT_ENABLE false
```

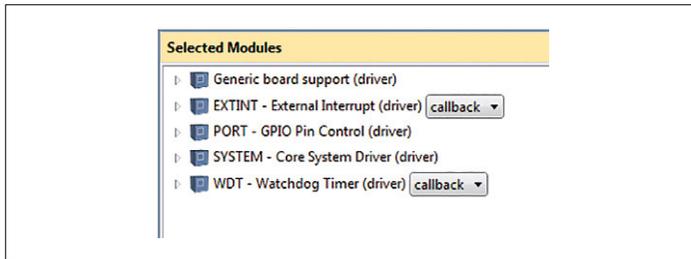


Figure 5. Toutes les bibliothèques requises par le projet de chien de garde.

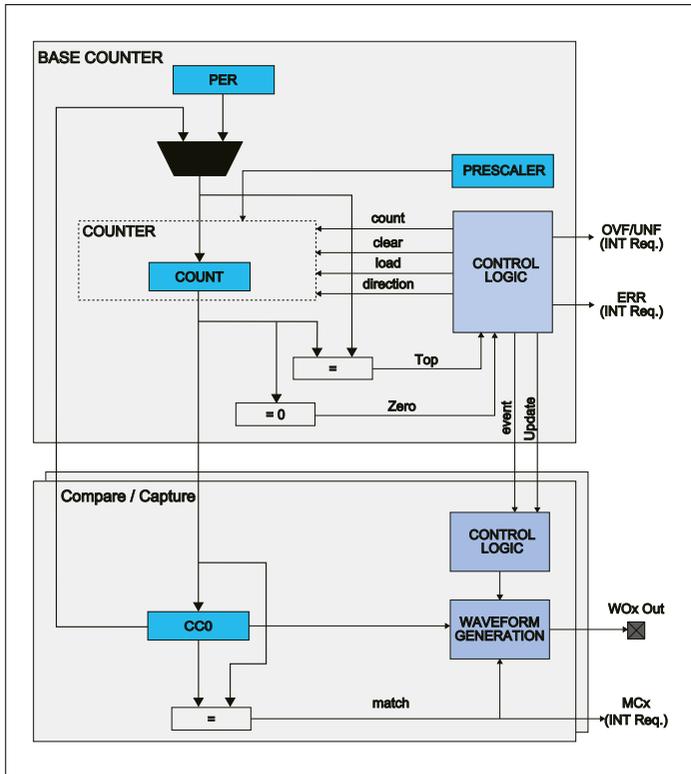


Figure 6. Voici la structure d'un temporisateur/compteur (*Timer/Counter*).

du TCG (**listage 2**). Le prédiviseur est paramétré à 1024, de façon à ce que le compteur du TCG incrémente le plus lentement possible. `CONF_CLOCK_GCLK_4_ENABLE` a été mis à `true` pour activer le générateur 4. Le listage montre aussi quel oscillateur utilise ce générateur, à savoir l'oscillateur-maître de 8 MHz qui cadence aussi le μC . Assurez-vous que toutes les bibliothèques requises ont bien été liées dans le projet par le biais de l'Assistant ASF (**fig. 5**).

Maintenant, compilez ce programme avec Studio et transférez-le vers la carte (*Start without Debugging*) : la LED0 sur la carte clignote, vu que l'IPA l'active d'abord et que la réinitialisation due au débordement du TCG la rappelle au niveau bas (LED

éteinte). Si vous appuyez sur SW1, une interruption externe met `i` à 1. Le TCG est maintenant réinitialisé en permanence dans la boucle sans fin de sorte qu'il ne peut plus déclencher d'interruptions et la LED garde son dernier état. Une action sur SW2 a pour effet de remettre la variable `i` à 0 par le biais d'une interruption externe. Le TCG n'est plus réinitialisé et la LED0 se remet à clignoter [2][3].

Le temporisateur/compteur

Notre SAM D20 comporte huit temporisateurs/compteurs (TC, *Timer/Counter*) standard polyvalents. Comme expliqué dans le premier article, il est possible de configurer un temporisateur en temporisateur 8 bits, 16 bits ou d'en combiner deux pour en faire un temporisateur 32 bits. Comme sur les microcontrôleurs 8 bits, un tel temporisateur multifonction peut compter des cycles d'horloge (et donc déterminer un intervalle de temps) et déclencher une interruption soit en cas de débordement, soit suite à une comparaison avec un registre de comparaison. Il est également possible de délivrer un signal MLI (PWM) aux broches correspondantes. Les TC du SAM D20 en sont capables eux aussi, mais peuvent faire bien plus. La **figure 6** montre la structure simplifiée d'un TC du μC Cortex-M0+. Le bloc compteur (en haut) est doté de plusieurs unités de comparaison (en bas). Lorsque la valeur du compteur est égale à celle du registre de comparaison, soit le niveau des sorties ou des canaux (par exemple W00 et W01) associés au temporisateur change, soit une interruption est déclenchée. Le temporisateur est bien entendu en état de compter et de décompter, de se réinitialiser lui-même en cas de débordement. Ces blocs de comparaison jouent un rôle essentiel dans la flexibilité du temporisateur. Ce dernier peut fournir différents types de signaux MLI, mesurer soit des durées soit des cycles d'horloge, compter des événements, produire différents signaux rectangulaires et bien plus encore. Les temporisateurs peuvent aussi fonctionner lorsque le μC est en mode sommeil ou pendant le débogage. Le synoptique montre que le temporisateur possède son propre prédiviseur et peut dériver son horloge de n'importe quel générateur GCLK.

Le premier projet de temporisateur

Avec le circuit ci-dessus, testons le temporisateur en mode comptage. Le projet commute les LED lorsque le compteur atteint la valeur des registres de comparaison ou qu'il déborde. Ouvrez le projet *The first project with the counter* de l'archive Zip. Dans le fichier *Main*, vous verrez que le programme commence par la définition des constantes symboliques pour les broches reliées aux trois LED de la plaque d'essais. On trouve ensuite les prototypes de fonction avant la création d'une structure de type `tc_module` appelée `tc_instance` permettant, ultérieurement, d'appeler le TC.

On découvre ensuite la première fonction (déjà connue) de

Liens

[1] www.elektor-magazine.fr/150041

[2] www.atmel.com/Images/Atmel-42124-SAM-D20-Watchdog-Driver-WDT_Application-Note_AT03264.pdf

[3] www.atmel.com/Images/Atmel-42112-SAM-D20-External-Interrupt-Driver-EXTINT_Application-Note_AT03246.pdf

[4] www.atmel.com/Images/Atmel-42123-SAM-D20-D21-Timer-Counter-Driver-TC_Application-Note_AT03263.pdf

Listage 3. La fonction de configuration du TC.

```

void configure_tc(void)
{
    struct tc_config config_tc;
    tc_get_config_defaults(&config_tc);

    config_tc.counter_size = TC_COUNTER_SIZE_8BIT;    config_tc.clock_source = GCLK_GENERATOR_1;
    config_tc.clock_prescaler = TC_CLOCK_PRESCALER_DIV1024;    config_tc.counter_8_bit.period = 99;
    config_tc.counter_8_bit.compare_capture_channel[0] = 33;    config_tc.counter_8_bit.compare_capture_
    channel[1] = 66;

    tc_init(&tc_instance, TC0, &config_tc);    tc_enable(&tc_instance);
}

```

configuration en sortie des trois broches des LED. Suivent les trois fonctions Callback du temporisateur : `tc_callback_on_channel0`, `tc_callback_on_channel1` et `tc_callback_on_overflow`. Conformément à leur désignation, la première fonction est appelée en cas de concordance du compteur TC avec le registre de comparaison 0, la seconde en cas de concordance avec le registre de comparaison 1 et la troisième en cas de débordement du compteur TC. Chacune de ces trois fonctions commute, par le biais de `port_pin_toggle_output_level(LEDx)` l'une des trois LED, le x représente G(reen), Y(ellow) ou R(ed).

Aux fonctions Callback succède la fonction de configuration du TC (voir **listage 3**). Après la création et l'initialisation par paramètres on commence, dans la structure de configuration, par paramétrer le mode (8 bits), la source d'horloge (générateur GCLK 1) et la valeur du prédiviseur (1024). Il ne reste plus qu'à paramétrer les valeurs des deux registres de comparaison et du débordement. Nous veillons à des intervalles égaux (33, 66, 99), de sorte que le motif de clignotement des LED soit uniforme lui aussi. Les paramètres prévus sont transmis au temporisateur/

compteur0 par `tc_init(&tc_instance, TC0, &config_tc)`; pour finir, `tc_enable(&tc_instance)` active le TC.

La fonction `configure_tc_callbacks` ne fait qu'enregistrer les fonctions Callback pour divers événements afin de les activer ensuite. Le reste du fichier *Main* se résume à l'appel des fonctions de configuration une seule fois. La boucle sans fin reste vide. Dans le fichier *conf_clocks.h*, il faut activer le générateur GCLK 1 par `CONF_CLOCK_GCLK_1_ENABLE = true` et mettre en fonction l'oscillateur 32 kHz externe qui fournit son horloge au dit générateur par un `true` pour `CONF_CLOCK_XOSC32K_ENABLE`. Bien entendu, les bibliothèques nécessaires doivent avoir été liées par le biais de l'Assistant ASF (**fig. 7**).

Compilez le projet et vous verrez les LED s'allumer l'une après l'autre avant d'être éteintes.

Produire des signaux MLI

Abordons brièvement la production de signaux MLI (**Modulation en Largeur d'Impulsion**, PWM pour **Pulse Width Modulation** en anglais). Comme une variante de Callback ne fait guère l'affaire ici, nous choisissons la variante « interrogée »

Listage 4. La fonction de configuration du TC pour une production de MLI.

```

void configure_tc(void)
{
    struct tc_config config_tc;

    tc_get_config_defaults(&config_tc);

    config_tc.counter_size    = TC_COUNTER_SIZE_16BIT;
    config_tc.wave_generation = TC_WAVE_GENERATION_NORMAL_PWM;

    config_tc.counter_16_bit.compare_capture_channel[0] = 0xFFFF;
    config_tc.pwm_channel[0].enabled = true;
    config_tc.pwm_channel[0].pin_out = EXT1_PWM_0_PIN;
    config_tc.pwm_channel[0].pin_mux = EXT1_PWM_0_MUX;
    config_tc.counter_16_bit.compare_capture_channel[1] = 0xFFFF;
    config_tc.pwm_channel[1].enabled = true;
    config_tc.pwm_channel[1].pin_out = EXT1_PWM_1_PIN;
    config_tc.pwm_channel[1].pin_mux = EXT1_PWM_1_MUX;

    tc_init(&tc_instance, EXT1_PWM_MODULE, &config_tc);

    tc_enable(&tc_instance);
}

```

(*polled*) de la bibliothèque TC. On utilise également la bibliothèque Delay (**fig. 8**). Le programme doit mettre en gradation simultanée les LED verte et rouge prises aux sorties (WO0/WO1) de TC6. La fonction de configuration du TC est reproduite dans le **listage 4**. Après la création et l'initialisation de la structure de configuration, on sélectionne la taille du compteur (16 bits) et le mode du TC (MLI normale). Puis on paramètre les deux canaux. Les deux ont une valeur de débordement de 0xFFFF (65535) et sont reliés chacun à la broche de sortie correspondante. Il ne reste plus qu'à charger les paramètres dans TC6 (qui se cache sous EXT1_PWM_MODULE) par `tc_init(&tc_instance, EXT1_PWM_MODULE, &config_tc)` et à les activer par `tc_enable(&tc_instance)`. `tc_instance`

permettra d'appeler TC6.

Dans la boucle sans fin de la fonction *Main*, les deux registres de comparaison des canaux MLI ne cessent, par le biais de l'instruction `tc_set_compare_value(&tc_instance, TC_COMPARE_CAPTURE_CHANNEL_0, Wert)`, d'être modifiés dans une boucle For. On a incrémentation de l'un des registres et décrémentation de l'autre. Plus la valeur de comparaison est élevée, plus le rapport cyclique, et donc la luminosité de la LED le sont. 65535 et 0 sont respectivement les valeurs maximale et minimale de ce registre de comparaison. La seconde boucle For dans la boucle sans fin fait exactement l'inverse – la luminosité de LED verte diminue progressivement alors que celle de la LED rouge augmente. En l'absence de tout autre para-

Le compteur en temps réel (CTR)

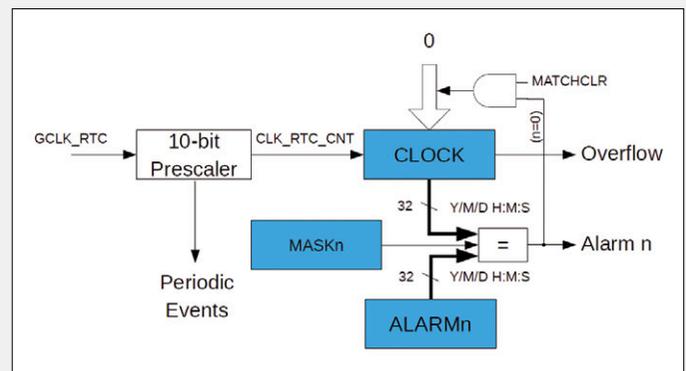
Le contrôleur principal de la carte SAM D20 Xplained Pro intègre, comme la plupart des microcontrôleurs 32 bits, un compteur en temps réel (RTC), ce qui n'est pas le cas des μC 8 bits.

Ce CTR est un compteur 32 bits avec un prédiviseur 10 bits pouvant dériver son horloge de différentes sources. Il peut travailler soit en mode Compteur pour compter sur de très longues durées, soit comme un CTR classique, en mode Calendrier pour délivrer la date et l'heure. En mode Compteur, il y a, tout comme avec un TC, comparaison de la valeur actuelle avec une valeur de comparaison. En mode Calendrier, il est possible, à une date et à une heure (format 24 ou 12 h) données, de déclencher une interruption (*RTC Alarm*). Si le CTR doit fonctionner en mode Calendrier, il doit être cadencé au rythme de 1 Hz.

Deux programmes simples de l'archive Zip, *The first project with RTC Counter* et *The first project with the RTC calendar mode*, montrent comment utiliser les deux modes du CTR, avec le circuit proposé plus haut.

Le premier programme est un clignotant piloté par le CTR : les LED verte et rouge passent, en alternance, au jaune. On commence par configurer les broches de commande des LED de la plaque d'essais en sortie et mettre PB05 au niveau haut. Il ne reste plus ensuite qu'à commuter simultanément les trois broches. C'est ce qui se passe dans la fonction Callback du CTR, appelée en cas de débordement du compteur. La fonction de configuration a la structure classique (**listage 5** en haut). Son milieu comporte des instructions importantes. L'une paramètre le prédiviseur à 1 et l'autre convertit le CTR 32 bits en compteur 16 bits.

On trouve, après cette fonction, l'enregistrement et l'activation de la Callback ainsi que l'appel des fonctions de configuration dans la fonction *Main*. La dernière instruction de la fonction *Main* avant la boucle vide s'intitule `rtc_count_set_period(&rtc_instance, 1000)`. 1000 est la valeur de débordement du CTR. Le clignotant change d'état toutes les secondes parce que le générateur GCLK 2 délivre au CTR un



signal d'horloge prédivisé à un kilohertz et que le débordement a lieu après 1 000 périodes d'horloge exactement.

Le second projet fait travailler le CTR en mode Calendrier. Ici, seule la LED verte est utilisée. Une alarme CTR en provoque le changement d'état toutes les secondes : elle clignote. Pour ce faire, il faut que la Callback de l'alarme CTR commute la LED et que l'alarme suivante soit paramétrée à un intervalle d'une seconde. La fonction de configuration du CTR prend la forme du **listage 5** (en bas).

Dans la fonction *Main*, l'Alarm0 qui déclenche la Callback requise, est paramétrée à 01.04.2015 00:00:01. Bien plus loin, après la fonction d'enregistrement et d'activation de la Callback Alarm0 et l'appel des nombreuses fonctions de configuration, l'heure de démarrage du CTR est paramétrée à 31.03.2015 23:59:59. De ce fait, la première alarme est déclenchée deux secondes après le début du programme, d'où le changement d'état de la LED verte. Ensuite, le déclenchement de l'alarme se fait au rythme des secondes : la LED change d'état toutes les secondes.

Il faut, pour les deux projets, que le générateur CLK 2 soit activé dans le fichier *conf_clocks.h*. On a besoin des bibliothèques CTR dans la variante Callback ; elles sont, comme d'habitude, ajoutées aux projets par l'Assistant ASF.

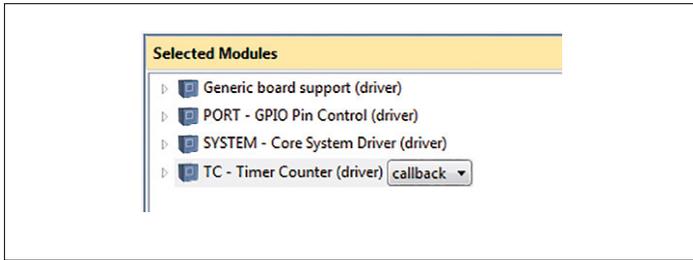


Figure 7. Toutes les bibliothèques pour le projet de temporisateur dans l'Assistant ASF.

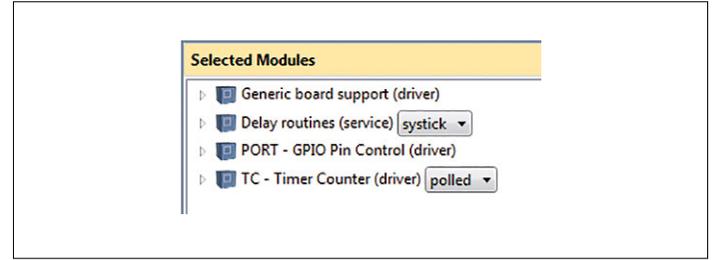


Figure 8. Les bibliothèques nécessaires pour le projet de MLI (PWM).

mètre, TC6 dérive son horloge du générateur principal 0, qui ne requiert pas d'être activé spécifiquement (on se souvient qu'il doit aussi cadencer le µC).

Compilez le programme et gardez les deux LED à l'oeuil [4].

Vous pouvez maintenant mettre en œuvre certains des éléments périphériques les plus complexes du µC. Plus tard, lorsque

nous en serons au système d'événement, nous découvrirons une nouvelle fonction du temporisateur. D'ici là, prenez du bon temps à essayer et modifier les fonctions décrites ici !

(150041 – version française : Guy Raedersdorf)

Listage 5. Le compteur en temps réel (RTC).

```
void configure_rtc_count(void)
{
    struct rtc_count_config config_rtc_count;
    rtc_count_get_config_defaults(&config_rtc_count);

    config_rtc_count.prescaler      = RTC_COUNT_PRESCALER_DIV_1;
    config_rtc_count.mode           = RTC_COUNT_MODE_16BIT;
    rtc_count_init(&rtc_instance, RTC, &config_rtc_count);
    rtc_count_enable(&rtc_instance);
}
...

void configure_rtc_calendar(void) //configure the RTC
{
    struct rtc_calendar_config config_rtc_calendar;
    rtc_calendar_get_config_defaults(&config_rtc_calendar);
    config_rtc_calendar.clock_24h = true;

    alarm.time.year      = 2015; //first alarm date
    alarm.time.month     = 4;
    alarm.time.day       = 1;
    alarm.time.hour      = 0;
    alarm.time.minute    = 0;
    alarm.time.second    = 1;

    config_rtc_calendar.alarm[0].time = alarm.time;
    config_rtc_calendar.alarm[0].mask = RTC_CALENDAR_ALARM_MASK_YEAR;

    rtc_calendar_init(&rtc_instance, RTC, &config_rtc_calendar);
    rtc_calendar_enable(&rtc_instance);
}
```

19^e (et dernier) jour :

les cartes multicouche

Neil Gruending (Canada)

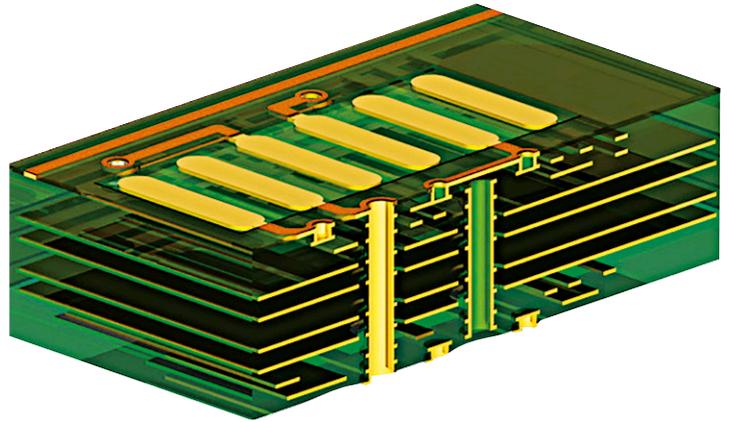
Jusqu'ici nos exemples utilisaient des cartes à deux couches, mais DesignSpark peut en gérer bien plus que ça !

Tous les exemples présentés dans cette série s'appuyaient sur des cartes double couche, mais DesignSpark supporte des cartes à quatre couches ou plus. Je l'ai déjà brièvement mentionné dans les volets précédents. Voyons comment créer des cartes multicouche dans DesignSpark.

Prévoir l'agencement des couches

La première étape de la conception d'une carte multicouche est de décider des couches nécessaires. Les couches sont typiquement utilisées pour les signaux, les alimentations, ou bien mixtes. On peut router ce que l'on veut sur une couche de signal mais par convention on n'y route que des signaux et pas d'alimentations. Sur une couche d'alimentation, on ne trouve qu'un *net* d'alimentation qui occupera toute la surface. Une couche d'alimentation partagée sera utilisée pour dédier de grandes surfaces de cuivre à plusieurs *nets* d'alimentation. Une couche mixte est une couche d'alimentation sur laquelle on aura routé quelques signaux ; on cherchera habituellement à en minimiser le nombre.

DS prend en charge tous ces types de couches avec ses couches électriques, c'est l'utilisation que vous en ferez qui en définira le type. Si vous ne routez que des signaux sur une couche, elle sera considérée comme une couche de signal. Mais rien ne vous empêche d'y ajouter des zones de cuivre d'alimentation pour créer une couche mixte. L'utilisation de plusieurs zones de cuivre pour des alimentations aboutira à une couche d'alimen-



tation partagée. Si vous voulez une vraie couche d'alimentation avec un seul *net*, vous pouvez configurer la couche pour cela.

Réglage des couches

Le moyen le plus facile de régler la configuration des couches est d'utiliser l'assistant *New PCB Wizard* (fig. 1). Vous choisirez le nombre de couches désirées sur votre carte ; la couche du dessus est numérotée *layer 1*. Ensuite vous configurerez vos couches d'alimentation si vous souhaitez plus de quatre couches électriques et que les options d'agencement vous conviennent. Par exemple, pour une carte à quatre couches, vous pourrez configurer les couches 2 et 3 en couches d'alimentation. Une fois terminé avec l'assistant, il est facile de modifier l'agencement des couches à l'aide du menu *Settings* → *Design Technology*. L'onglet *Layers* liste toutes les couches de la carte et permet de les éditer ou d'en ajouter de nouvelles. La figure 2 montre un exemple, créé avec l'assistant, de carte à 8 couches où les couches 2 et 3 sont des couches d'alimentation. Une couche d'alimentation dans DS est une couche électrique à laquelle il faudra associer un *net*.

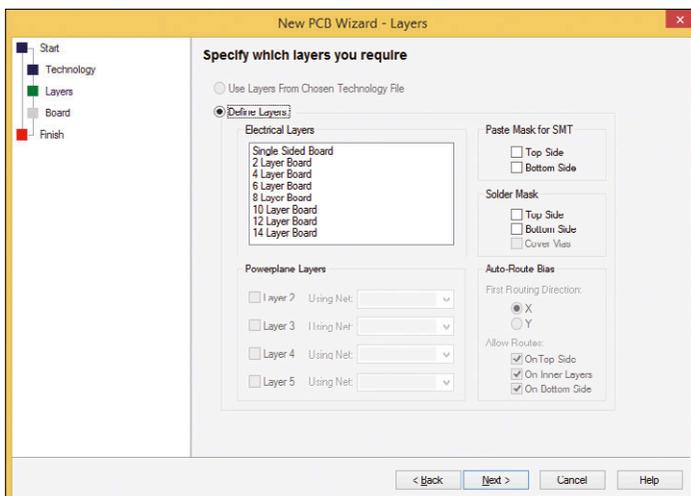


Figure 1. Assistant de création de carte.

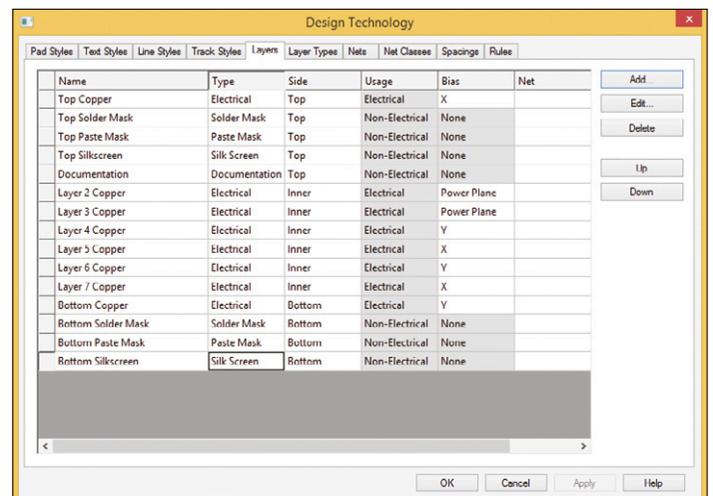


Figure 2. Fenêtre Design Technology Layers.

en coopération avec **DESIGNSPARK**

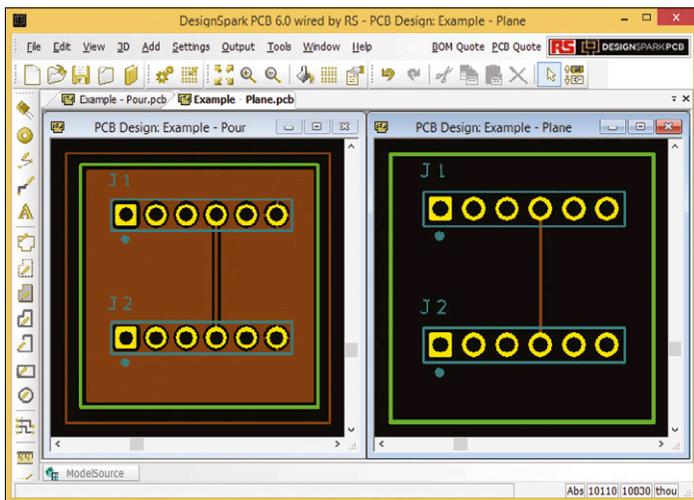


Figure 3. Zones de cuivre vs. plans.

Petit conseil : donnez un nom explicite comme *Top Signal* ou *Ground Plane* à chacune des couches afin de faciliter le choix de la couche lors du routage. Normalement on ne route pas de piste sur les couches formées d'un plan mais DS permet de le faire si nécessaire. Les pistes seront visibles comme n'importe quelle autre piste mais lors de la production des fichiers Gerber, DS s'assurera que les plans ne sont pas en contact avec les pistes.

Zones de cuivre et plans

Je préfère employer des zones de cuivre plutôt que des plans, même si celles-ci demandent un peu plus de travail. Voyez l'exemple de la **figure 3** : la carte de gauche utilise une zone de cuivre pour le plan de masse, celle de droite un plan. Électriquement c'est la même chose et les fichiers Gerber seront presque identiques. La seule différence est que l'on peut voir le résultat final dans DS avec une zone de cuivre alors qu'il faudrait créer et visualiser les fichiers Gerber avec une autre application pour admirer le résultat avec les plans d'alimentation. Une autre caractéristique sympa des zones de cuivre est leur flexibilité extrême pour créer des plans d'alimentation. Si vous aviez besoin d'une petite zone à 1,8 V au milieu d'une couche d'alimentation 3,3 V, vous utiliseriez une zone de cuivre 1,8 V sur le plan 3,3 V et DS vous avertirait avant de vous laisser le faire. Cependant, la meilleure solution est d'avoir recours à des zones de cuivre pour le 1,8 V et le 3,3 V, et de régler l'ordre de création correctement.

Production du Gerber

Par chance, la production de fichiers Gerber pour les cartes multicouche se passe exactement comme avec les cartes double couche. La **figure 4** montre la fenêtre *Output Manufacturing Plots* pour notre carte d'exemple à 8 couches ; l'onglet *Settings* est affiché, ce qui permet de voir les paramètres disponibles pour chaque couche. En temps normal, vous n'aurez pas besoin de les modifier. Ils servent à changer l'isolation entre les plans et à régler l'allègement thermique pour les plans des couches

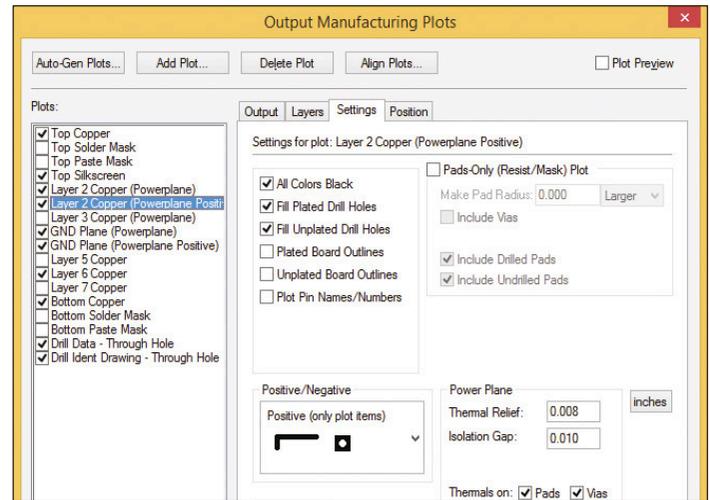


Figure 4. Option de production des fichiers Gerber.

d'alimentation. Notez également que toutes les couches de cuivre ne sont pas sélectionnées : DS n'a sélectionné que celles utilisées pour cet exemple simple. Vous remarquerez peut-être que les couches *Layer 2 Copper* et *GND Plane* sont suivies des options de traçage *Powerplane* et *Powerplane Positive*. Normalement, les plans d'alimentation sont tracés en négatif et c'est exactement ce que l'option de traçage *Powerplane* fera. Cependant, j'ai aussi utilisé des pistes sur ces deux couches qui ne pourront pas être représentées en négatif ; DS devra donc produire une image positive qui ne contiendra que la piste et les pastilles des composants. C'est en combinant les deux que l'on obtient l'image finale de la carte (**fig. 5**). La zone noire à l'intérieur de la carte représente le plan, sauf à l'endroit où se trouve la piste en violet. Le bleu foncé représente l'isolation entre le plan et la piste. L'isolation est également visible autour du contour de la carte.

Avant de produire tous les fichiers Gerber, je vous recommande de créer un répertoire dédié aux fichiers créés automatiquement : cliquez sur le bouton *Options* et renseignez le dossier pour l'option *Where Plot Files Are Written*. Tous les fichiers Gerber se retrouveront dans ce dossier plutôt que dans le dossier principal du projet.

Conclusion

Nous venons de voir comment créer une carte multicouche et régler leur agencement. Cet article conclut le tour d'horizon des capacités de DesignSpark PCB. J'espère que cette série vous incitera à l'essayer sur votre prochain projet ! ◀

(150023 – version française : Kévin Petit)

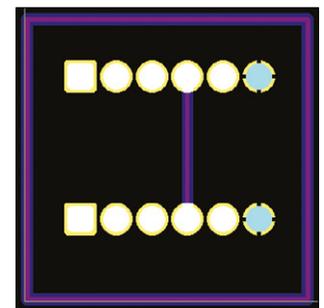


Figure 5. Tracé Gerber de la couche de masse.

C'est WunderBar ! l'Internet des Objets

Jaime González-Arintero (Elektor)

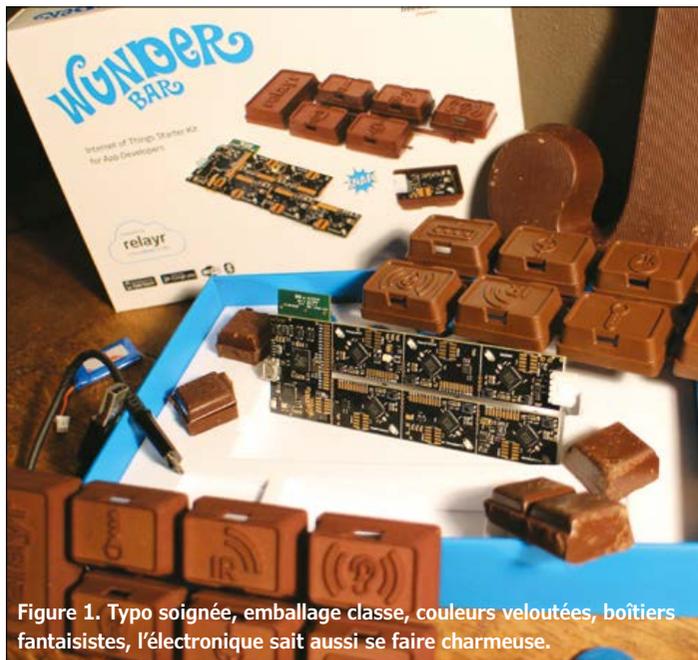


Figure 1. Typo soignée, emballage classe, couleurs veloutées, boîtiers fantaisistes, l'électronique sait aussi se faire charmeuse.

Du chocolat bien carré

J'avoue, je n'avais aucune idée de ce qu'était WunderBar la première fois que j'en ai entendu parler. WunderBar se présente comme un « kit de démarrage destiné aux développeurs qui écrivent des applications pour l'Internet des Objets ». Voilà qui est clair ! Ce qui l'était moins pour les visiteurs du dernier salon **electronica** de Munich, c'était la présence sur notre stand d'étranges carrés de chocolat. Intrigués vous aussi ? Voyons cela de plus près.

Le kit comprend un module maître et un jeu de six modules que je décrirai un peu plus loin. Considérez l'ensemble comme une architecture maître/esclave. Les données collectées par les modules capteurs sont envoyées au module maître par Bluetooth Low Energy (BLE). Le maître télécharge ensuite ces données par Wi-Fi vers une plateforme en nuage du site **relayr**. Cette sorte de barre e-chocolatée sert donc de lien entre un réseau de capteurs et le nuage **relayr**. Toutes les données collectées par les modules sont envoyées et stockées dans ce nuage. Elles peuvent ensuite être récupérées et utilisées par d'autres dispositifs avec les droits d'accès afférents, à moins qu'un capteur n'ait été rendu public (voir plus bas). Il me semble logique ici de supposer qu'un capteur n'est pas réellement « interrogé » au moment où sa valeur est demandée, mais que celle-ci est extraite des valeurs précédemment chargées et stockées dans le nuage par le module maître.

* en allemand, wunderbar signifie miraculeux, merveilleux

L'Internet des Objets (IdO) a jeté une pomme de discorde entre les électroniciens. Le battage médiatique autour de l'IdO n'y est sans doute pas étranger et n'aura fait (selon moi) qu'introduire un peu plus de confusion dans l'esprit du grand public. Cette intelligence ambiante annonce-t-elle vraiment une révolution, ou n'est-elle, comme certains le pensent, qu'une solution parmi d'autres pour relever les grands défis de demain ? Plutôt que de gloser sans fin sur le sujet, j'ai préféré étudier l'objet en question. Pour cela j'ai utilisé le kit WunderBar. Il ne provient pas d'un géant californien (*Intel*, pour n'en citer aucun), mais d'une jeune pousse berlinoise appelée **relayr** [1]. Autre particularité de WunderBar, il n'est pas axé sur le matériel, mais sur le logiciel. Cela étant dit, qu'a-t-il donc de si *wunderbar** ?

Une recette secrète ?

WunderBar se distingue bien sûr par l'originalité de son conditionnement, mais ce qui le rend vraiment unique ce sont ses codes d'exemple, ses bibliothèques et kits de développement logiciels (SDK), sans oublier une documentation très fournie. Et rien de secret dans cette recette, le code source et le matériel sont ouverts, autrement dit rendus publics pour qui souhaite les étudier ou les modifier. Pour cela rendez-vous sur le dépôt Github [2].

La boîte qui contient le WunderBar est si séduisante que l'ouvrir donne l'impression de déballer un paquet cadeau sous le sapin. La première chose que l'on découvre est la tablette de chocolat en plastique dont les carrés serviront de boîtiers aux modules. On trouve ensuite un câble USB, une batterie Li-ion de 3,7 V/130 mAh pour le module maître, cinq piles boutons CR2032 de 3 V pour les modules capteurs, six adhésifs pour fixer les boîtiers et, bien sûr, la carte WunderBar (**fig. 1**). Chaque module peut être détaché de la carte. Si, avant de les séparer, vous alimentez le module maître par USB (la batterie étant reliée à la carte !), ils seront eux aussi alimentés, même si vous n'avez pas encore inséré leurs piles boutons. Bien que la carte soit séparable, les modules capteurs sont en effet reliés à la même ligne d'alimentation par une sorte de crénelage (cf. *cartes crénelées*, Elektor mars 2015). Ces encoches serviront ultérieurement à empêcher les modules de sortir de

leurs boîtiers en plastique.

Le module maître repose sur un ARM Cortex-M4 MK24 de la série Kinetis de Freescale. Ce contrôleur est doté d'une mémoire flash de 1 Mo ainsi que de multiples périphériques et interfaces (dont SPI, I²C et UART). Il se présente dans un boîtier XFBGA extrêmement fin, que même dans nos rêves les plus fous nous serions incapables de *rebillor*. C'est un module Wi-Fi GainSpan GS1500 qui prend en charge la connexion Internet. La communication BLE avec les modules capteurs est quant à elle assurée par l'émetteur-récepteur intégré d'un SoC nRF51822 en boîtier QFN à 48 broches de Nordic Semiconductor.

Le kit est composé de six modules : luminosité/couleur/proximité, accéléromètre/gyroscope, température/humidité, son/niveau sonore, émetteur IR (pour commander la porte d'un garage, un système d'air conditionné...), et enfin un module *Bridge* équipé de 8 ports GPIO et d'un connecteur à 4 broches compatible avec les modules Grove (pour créer une interface avec d'autres outils de développement). Les deux derniers modules ne sont pas vraiment des capteurs, mais formeront certainement les briques des applications les plus intéressantes. Tous ces modules sont dotés d'un port de débogage et de huit broches que l'on peut utiliser comme GPIO, I²C, UART, MLI ou entrées analogiques.

Embarquement

La première étape consiste à suivre une procédure de reconnaissance et de configuration appelée *OnBoarding* [3]. Elle nécessite la création d'un compte **relayr** et l'installation sur un téléphone Android ou iOS de l'application *relayr Manager App* [4]. Le rôle de cette appli est d'enregistrer votre Wun-



Figure 2. Il est IMPOSSIBLE qu'on ne vous demande pas ce que sont ces *machins* !

derBar dans la plateforme **relayr** puis d'écrire divers identifiants sur le module maître et les six modules. Ces identifiants garantiront que seul le maître pourra recevoir les données des capteurs et les relayer vers le nuage. Avant de démarrer le processus, l'appli demande d'activer la connexion Bluetooth du téléphone. Comme en mode de fonctionnement normal la

▶ On vous demandera forcément : « c'est quoi ces *machins* ! » Effet badaud garanti !

WunderBar ne communique pas avec le téléphone de cette façon, je suppose que Bluetooth ne sert ici qu'à établir la connexion avec le nuage. Par la suite, il est possible d'accéder directement aux modules par BLE (cette fonction s'active depuis l'application). Si le Bluetooth ne vous sert pas après cette procédure, vous pouvez le désactiver sans que cela n'entrave le bon fonctionnement du kit. Une fois l'*OnBoarding* effectué, vous pouvez séparer les modules de la carte, les alimenter avec les piles boutons, les insérer dans leurs boîtiers chocolatés (**fig. 2**) puis les placer où bon vous semble afin de découvrir ce qu'ils renvoient. La **figure 3** montre le capteur de température et d'humidité posé sur le rebord d'une fenêtre du château Elektor. Aucun doute, d'après la capture d'écran de mon téléphone, j'étais bien aux Pays-Bas ! (Oui, un Espagnol comme moi ne peut s'empêcher de mettre un point d'exclamation après avoir parlé de la température d'ici !)

Chocolaterie logicielle

Comme je l'ai souligné plus haut, ce qui distingue vraiment WunderBar des autres kits est son environnement logiciel. Les cinq SDK actuellement disponibles sont tous *open source* : Android, iOS/OSX, JavaScript, Python et C# (.NET). Avant de pouvoir accéder aux services de **relayr**, et donc de commencer à coder, il faut obtenir une « clé API ». Cela ne prend

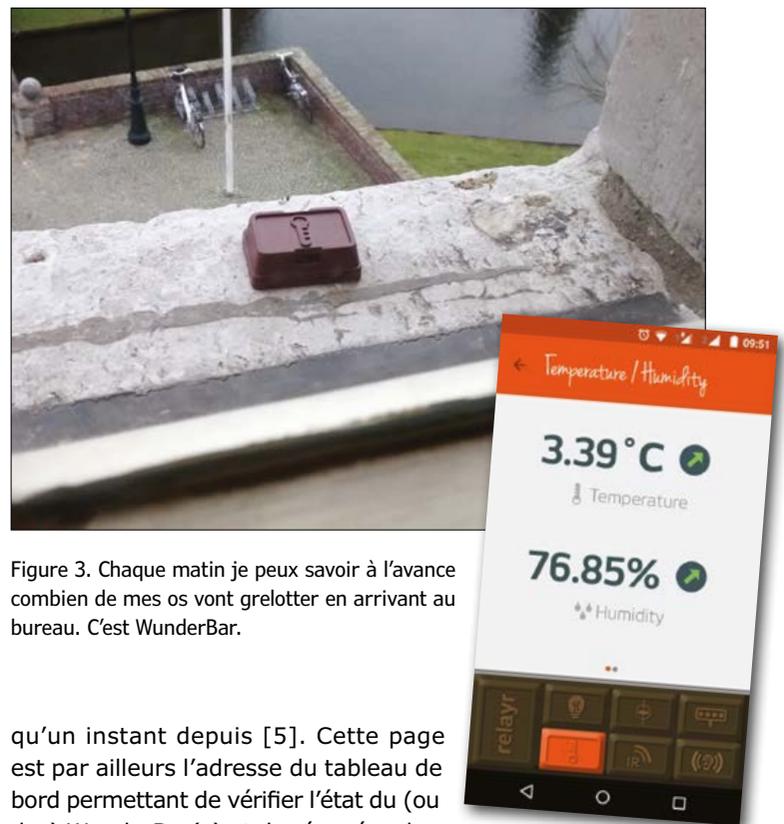


Figure 3. Chaque matin je peux savoir à l'avance combien de mes os vont grelotter en arrivant au bureau. C'est WunderBar.

qu'un instant depuis [5]. Cette page est par ailleurs l'adresse du tableau de bord permettant de vérifier l'état du (ou des) WunderBar(s) et de récupérer les valeurs lues par les capteurs. La **figure 4** montre le tableau de bord des capteurs que j'avais placés au château et chez moi

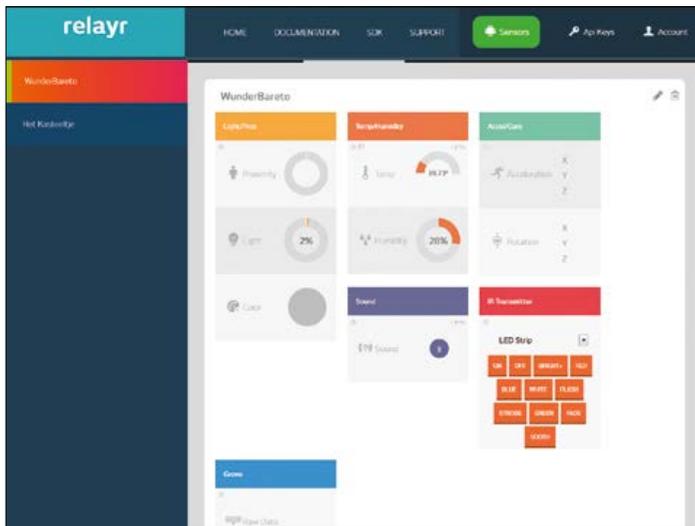


Figure 4. Dans le tableau de bord, sous *Sensors* -> *My Devices*, vous pouvez voir l'état des modules liés à votre compte.

pour faire un premier essai. Depuis ce tableau de bord, vous pouvez aussi rendre publiques les valeurs d'un capteur et les incorporer dans une application à l'aide d'un script. Dans le cas contraire, elles restent privées et visibles uniquement de leur propriétaire via le tableau de bord ou une application. On peut également définir le taux de rafraîchissement de chaque capteur. Relever la température d'une pièce toutes les secondes serait par exemple absurde.

Avant de taper frénétiquement du code avec l'un ou l'autre des SDK disponibles, vous pourriez vouloir chercher quelque source d'inspiration. Pour cela le site **relayr** propose quelques muses chocolâtées, dont l'application appelée *TellMeWhen*. Comme son nom le sug-



Figure 5. Sous surveillance : mes bonbons, mon guide de survie Espagnol/ Néerlandais, et mon système portable de stockage en nuage (aussi appelé disquette). Je pourrais fermer l'armoire à clé bien sûr, mais où serait le plaisir de surprendre le coupable ?

gère, elle avertit l'utilisateur qu'un seuil donné a été franchi, par ex. une certaine température ou une certaine distance. La notification est envoyée en fonction de règles que vous écrivez vous-même. J'ai testé avec succès *TellMeWhen* en plaçant le capteur de luminosité/proximité dans une armoire de mon bureau, celle dans laquelle mes collègues aiment à farfouiller lorsque je ne suis pas encore arrivé (je ne suis pas du matin). Les gars, ce sont MES bonbons. O-k ?

Mode ou révolution ?

J'avoue avoir d'abord été peu réceptif aux promesses de l'Internet des Objets. L'hyper médiatisation du sujet et la description d'applications qui me semblaient aussi futuristes qu'inutiles me rendaient plutôt sceptique. Seul l'avenir nous dira ce qu'il en est bien sûr, mais je suis maintenant sûr que l'Internet des Objets prendra son essor. Comme avec toute nouvelle technologie, l'envol sera lent et les adaptations progressives,

Maman, je passe à la télé !

J'ai présenté WunderBar pour Elektor.TV il y a quelques mois. C'est en [6] si vous voulez me voir déballer le kit et casser quelques carrés de chocolat. N'oubliez pas non plus notre lettre d'informations hebdomadaire !



mais l'IdO grandira car les applications envisageables sont innombrables. Pensez simplement aux possibilités offertes par un retour en temps réel de milliers de capteurs sans fil placés n'importe où dans le monde, ou à l'exploitation de leur historique, ou à... attendez... je crois que l'alarme de l'armoire de mon bureau vient encore de se déclencher... ◀

(140474 - version française : Hervé Moreau)

Liens

- [1] <http://relayr.io>
- [2] <http://po.st/wunderbarhw>
- [3] <https://relayr.io/wunderbar>
- [4] <http://po.st/wunderbarapp>
- [5] <https://developer.relayr.io>
- [6] [http://po.st/wunderbarunboxing`](http://po.st/wunderbarunboxing)

en coopération avec

DESIGNSPARK

gyrateur intégré TCA580

drôle de composant n°14

David Ashton (Australie)

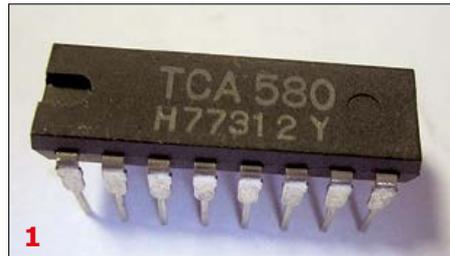
En désossant un vieil émetteur-récepteur radio, je suis tombé sur un circuit intégré inconnu : le TCA580, un gyrateur.

La première fois que j'ai entendu parler de gyrateur, c'était dans un simulateur de batterie (de musique) du n° 2 de l'édition anglaise d'Elektor (02/1975), article *How to Gyrate*. Un gyrateur simule une inductance. La théorie dit que les courants dans un condensateur et une inductance sont en opposition de phase. Un gyrateur produit à partir d'une source alternative appliquée aux bornes d'un condensateur un courant d'intensité plus élevée et en opposition de phase : voilà une inductance. L'article d'origine, à télécharger gratuitement sur [1], donne plus de détails et effleure l'aspect mathématique.

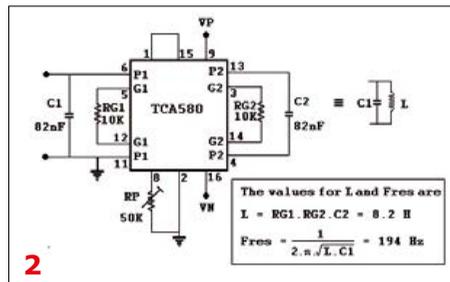
Le TCA580

Le TCA580 (**fig. 1**) est un circuit intégré gyrateur produit par Philips/Signetics au milieu des années 70 comme bloc de base pour des circuits de télécommunication. Ce circuit associé à un condensateur et deux résistances permet de créer une inductance virtuelle stable et élevée, jusqu'à 1 MHz (oui, Mega !), avec un très grand facteur de qualité ($Q > 1000$). Ajoutez un condensateur pour obtenir un circuit accordé à utiliser dans un filtre réjecteur ou comme détecteur de fréquence jusqu'à 10 kHz, sans souffrir de la taille, du prix et des mauvaises caractéristiques des inductances physiques. Aujourd'hui on aurait recours à des filtres actifs, des boucles à phase asservie ou au traitement numérique du signal mais dans les années 70, c'était une solution économique, réaliste et efficace.

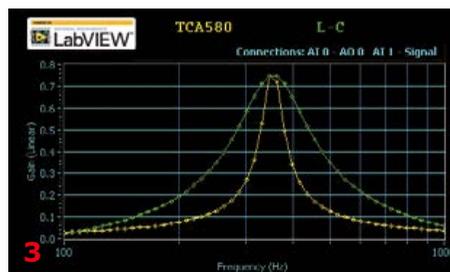
Les rares infos sur le TCA580 figurent dans une documentation de trois pages sur le net [2], avec peu de détails et de schémas d'application. Un circuit d'exemple montre une ajustable de 50 kΩ mais sans expliquer son utilité (il sert en fait à l'équilibre et à minimiser la dis-



1



2



3

torsion). J'ai découvert une référence à un volume (introuvable) de 1977 de la bibliothèque technique Philips intitulé *TCA580 Gyrateur IC. A Replacement for Wound Inductors in Low Frequency Circuits**. Néanmoins, la feuille de caractéristiques contient assez d'informations (**fig. 2**) pour câbler un circuit accordé sur une plaque d'essais. Après avoir fait fonctionner ce schéma avec des condensateurs de 47 nF pour C1 et C2, j'ai effectué une comparaison avec un circuit

LC discret (à l'aide d'une inductance à noyau réglable d'environ 14,4 H et d'un condensateur de 15 nF) qui résonnait autour de la même fréquence (350 Hz). Pour ne pas comparer des choux et des carottes, j'ai ensuite pris un condensateur de 160 nF pour C2 (celui du gyrateur) et le même condensateur de 15 nF que celui utilisé dans le circuit accordé pour C1 (le condensateur de résonance) : les fréquences étaient quasi identiques. La résistance entre la source et le circuit accordé était de 100 kΩ.

TCA580 vs. vraie inductance

Mes deux circuits résonnent à la même fréquence avec le même condensateur de 15 nF, je peux donc tracer deux courbes de réponse en fréquence (**fig. 3**, jaune pour le circuit à base de TCA580, vert pour le circuit discret). Le pic avec le TCA580 est très raide et étroit. Même en montant des composants de taille raisonnable pour les années 70, le circuit à base de TCA580 prendrait moins de place et aurait de meilleures performances que sa contrepartie discrète. Ces circuits étaient utilisés pour la production, le filtrage et la détection de tonalités par ex. (CTCSS). Les TCA580 sont toujours disponibles sur eBay pour environ \$10 pièce ; pas donné pour un circuit DIL. J'en ai quelques-uns, récupérés sur les vieilles radios et, même si je ne sais pas quoi en faire aujourd'hui, j'apprécie de m'amuser avec ces circuits qui étaient, dans les années 70, de petites merveilles de technologie. ◀

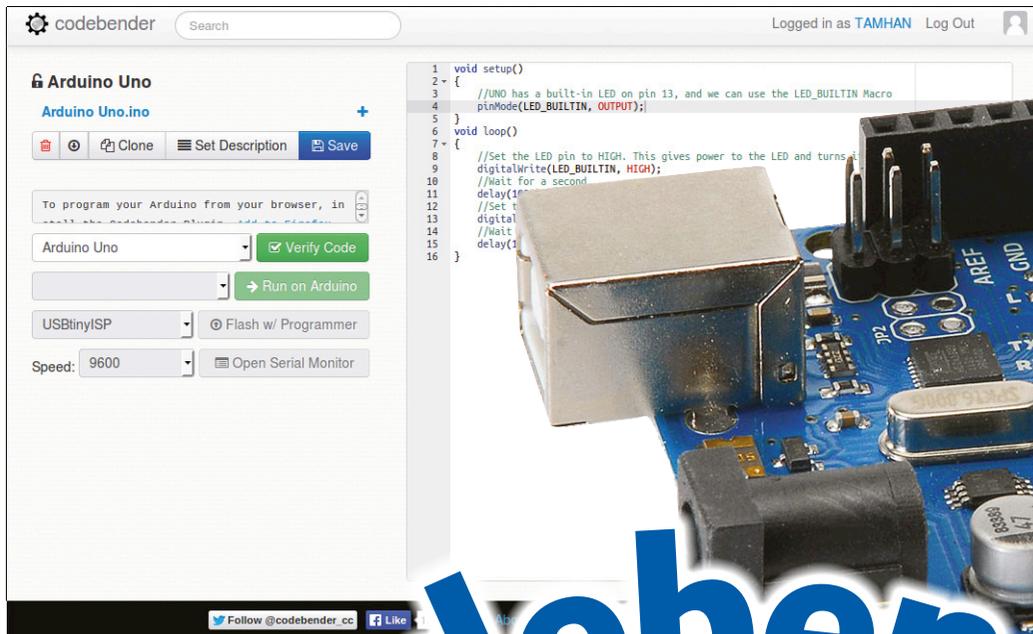
*NdT : Un substitut pour les inductances bobinées dans les circuits à basse fréquence

(150024 - version française : Kévin Petit)

Liens

[1] Article How to Gyrate : www.elektor-magazine.fr/150024

[2] Feuille de caractéristiques : www.datasheetarchive.com/dlmain/Databooks-4/Book618-6582.pdf



Codebender Arduino sur l'internet

Tam Hanna (Allemagne)

L'environnement de développement (IDE) de Massimo Banzi n'est pas du goût de tout le monde. En 2012, un groupe de développeurs grecs créa - surtout pour le plaisir - un IDE Arduino exécutable dans un navigateur. Au fil du temps il s'est révélé être une alternative sérieuse. En outre le stockage dans le Nuage évite le risque de perte de données. Grâce à des partenariats avec des créateurs de modules, un clic suffit pour faire migrer la plupart des bibliothèques *shield* vers ce projet.

La série PIC de Microchip a permis à nombre d'électroniciens de se familiariser avec la programmation des microcontrôleurs. Le succès de Massimo Banzi avec Arduino tient entre autres au fait qu'il permet aux non-électriciens de ne plus avoir à manipuler les puces ni à avoir à les « griller ». On connecte un Arduino à l'ordinateur via USB, le reste relève de la « Magie ».

Avec l'environnement de développement en ligne Codebender [cb], le transfert de progiciel est aussi devenu un jeu d'enfant. Il est possible d'ajouter à Chrome ou Firefox (Windows ou Linux) un greffon qui se charge de la communication entre la carte Arduino et l'ordinateur (Cf. **fig. 1**). Cependant, seules quelques-unes des cartes Arduino sont prises en charge, (Cf. encadré).

Pour utiliser Codebender, il faut s'enregistrer. Un assistant d'installation vous guide tout au long des étapes requises, installation du greffon comprise. On peut ensuite charger sur la carte Arduino le premier exemple de code (Blink.ino).

Une question d'IDE

La **figure 2** montre l'interface de Codebender : sur la partie droite de l'écran, l'éditeur de code qui repose sur l'éditeur ACE développé par Cloud9. Il se comporte - en gros - comme le font Eclipse et Cie, mais offre, en comparaison à l'IDE Arduino, des suggestions additionnelles plus intelligentes (*IntelliSense*).

Le processus de compilation rappelle - en partie - celui de l'IDE Arduino. Cliquez sur le bouton *Verify Code* pour un essai de compilation du code actuel de l'éditeur. En cas de succès, le programme affiche la taille du fichier compilé qui peut ensuite être envoyé vers la carte en cliquant sur *Run on Arduino*.

La barre d'état située au-dessus permet de commuter entre plusieurs fichiers *Sketch*. La flèche vers le bas télécharge le fichier .ino zippé et le fichier .hex écrit dans le contrôleur pour les stocker localement sur votre propre ordinateur. Pour l'instant, il n'est pas possible de télécharger le fichier .elf : l'ana-

lyse des programmes compilés en est plus difficile. Chrome affiche en outre, lors d'un clic sur l'option de téléchargement, un message d'erreur au sujet d'informations non stockées, mais sans importance pratique.

La flèche bleue présente sur la droite de l'écran sert à activer la fonction sociale de l'IDE. Elle permet le partage du code source via Twitter, Facebook et LinkedIn.

« Enfumer » les erreurs

Le célèbre AVR-GCC fournit des messages d'erreur dont l'interprétation requiert une compréhension plus profonde de C/C++. Le compilateur modulaire Clang (Cf. encadré), qu'utilise Codebender, est réputé lui pour ses conseils d'aide au débutant.

Le **listage 1** est un exemple de code avec trois erreurs. Après un clic sur le bouton *Verify Code*, Codebender balance un rapport d'erreur détaillé, qui souligne même par des tildes les passages concernés. L'IDE Arduino officiel est quant à lui bien moins bavard.

Des magazines scientifiques populaires ne cessent de demander que Codebender se charge aussi sous Clang de la compilation proprement dite. Cependant c'est hors sujet tant qu'il n'y aura pas de service fiable qui traduise le langage intermédiaire LLVM pour l'AVR (*backend*). Une entrée de blog publiée il y a un certain temps [TSIP] permet de créer le processus de la **figure 3**, qui décrit la compilation en détail.

Toutefois, les fichiers de compilation ne sont pas interchangeables. La version 1.5.7 de l'IDE Arduino a requis, pour le programme mentionné plus haut, 1 030 octets de mémoire de code alors que la compilation produite par Codebender en nécessitait 1 082. Dans le cas d'exemples plus complexes, la situation s'aggrave même : un produit créé par l'IDE Arduino de 3 052 octets en occupe 3 256 si compilé par Codebender. Des projets 100% « vides » occupent eux aussi, sous Codebender, quelques octets de plus – les amorces (*bootloader*) pourraient différer (marginale).

Coder à plusieurs

Ces dernières années, les processeurs en version Banzai se sont imposés dans les domaines des Maker et des semi-professionnels comme un standard *de facto*. L'ampleur du marché met à disposition une riche palette d'extensions qui facilitent la communication avec les composants les plus divers. Leur dénomination commune est *library* (bibliothèque) – Codebender en propose plusieurs centaines.

L'utilisation de bibliothèques prêtes à l'emploi commence par l'examen de la liste de la **figure 4** [Lib]. Pour utiliser les fichiers, il suffit de les lier au projet avec la directive *include* montrée.

Sus au Nuage

Codebender permet d'utiliser ses bibliothèques personnelles : en plus du recyclage (un classique), il devient possible de « modulariser » des *sketches* qui, normalement, ne comportent qu'un seul fichier.

Une question de processeur

Codebender travaille avec Arduino Uno et Leonardo, mais les systèmes complexes tels que Yun ou ceux basés sur la technologie ARM comme le Due ne lui disent vraiment rien.

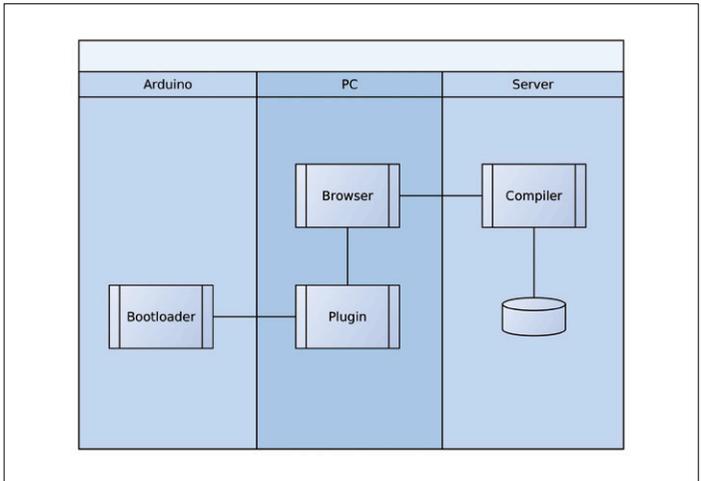


Figure 1. Codebender tourne dans un navigateur standard. Pour le transfert de progiciel vers une carte Arduino, on a besoin d'un greffon ; il en existe une version pour Firefox et Chrome.

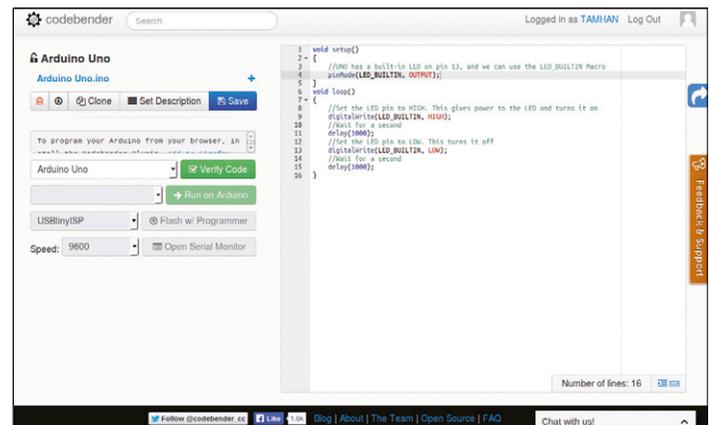


Figure 2. L'éditeur de code (Codeeditor) est l'élément central de l'application en ligne.

Listage 1. Petit exemple avec trois erreurs.

```
void setup()
{
  pinMode(LED_BUILTIN, OUTPUT);
}
void loop()
{
  digitalWrite(LED_BUILTIN, HEIH);
  delaI(1000);
  digitalWrite(LED_BUILTIN, LOW);
  delay(1000)
}
```

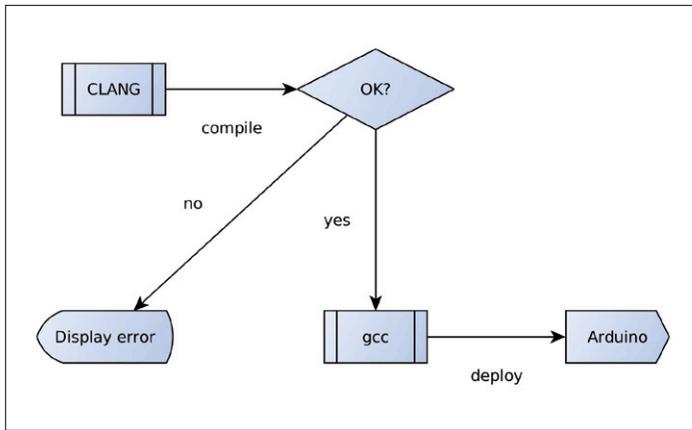


Figure 3. La détection des erreurs dans le code-source fait appel au compilateur frontal modulaire Clang. Le produit intermédiaire ainsi créé n'est cependant pas utilisé.

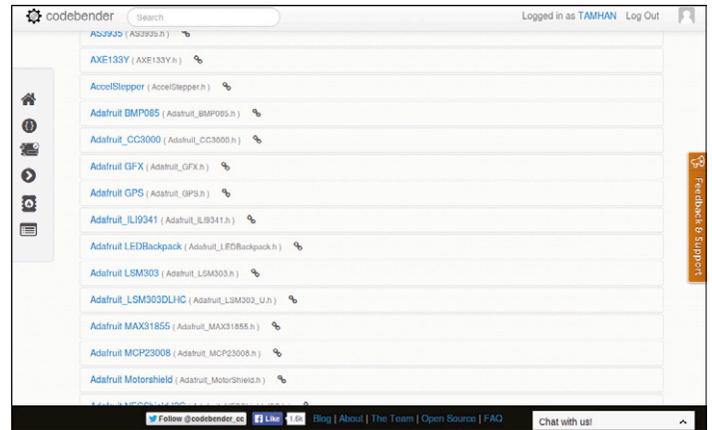


Figure 4. Liste des bibliothèques que l'on peut facilement intégrer dans ses programmes Arduino personnels.

Clang et LLVM

Clang est un projet de compilateur qui existe depuis des décennies, et constitue une alternative au fameux compilateur GCC. Il produit, pour une machine virtuelle appelée LLVM, du code ensuite converti, en statique ou dynamique, en code pour le système-cible concerné.

L'avantage de cette architecture est d'éviter au développeur du compilateur d'avoir à penser au système-cible. Son seul objectif est de convertir son langage dans le meilleur code LLVM possible (*frontend*). L'adaptation de LLVM en vrai code

machine se fait par les développeurs spécialisés qui n'ont pas besoin de connaître le langage de haut niveau (*backend*). Dans le cas de Codebender, le problème est qu'il n'existe pas – pour le moment – d'implémentation de type *backend* utilisable pour les microcontrôleurs AVR. De ce fait, Codebender n'utilise Clang que pour la compilation en code intermédiaire – quand ce processus s'exécute correctement, le programme est considéré comme exempt d'erreurs et peut être transmis au GCC.

Message d'erreur de Codebender :

```

Arduino Uno.ino:7:28: error: use of undeclared identifier 'HEIH'
    digitalWrite(LED_BUILTIN, HEIH);
                               ^
                               ^

Arduino Uno.ino:8:2: error: use of undeclared identifier 'delaI'; did you mean 'delay'?
    delaI(1000);
    ^~~~~
    delay

hardware/arduino/cores/arduino/Arduino.h:107:6: note: 'delay' declared here
void delay(unsigned long);
    ^

Arduino Uno.ino:10:13: error: expected ';' after expression
    delay(1000)
                ^
                ;
  
```

Message d'erreur de l'IDE Arduino officiel :

```

sketch_jan24a.ino: In function 'void loop()':
sketch_jan24a.ino:7:28: error: 'HEIH' was not declared in this scope
sketch_jan24a.ino:8:12: error: 'delaI' was not declared in this scope
sketch_jan24a.ino:11:1: error: expected ';' before '}' token
  
```

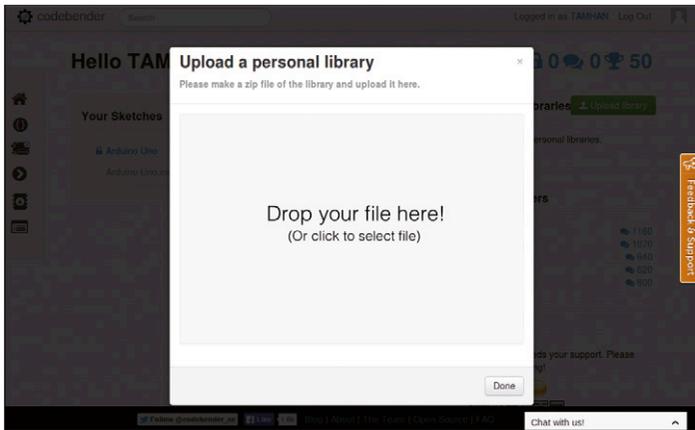


Figure 5. Codebender permet de télécharger ses propres bibliothèques vers le « Nuage ».

Pour cela, connectez-vous à CodeBender.cc pour afficher la page d'accueil de la **figure 5**. La fenêtre de dialogue *Personal Libraries* en haut à droite permet de télécharger (dans le sens montant) des archives .zip, composées de fichiers .c et .h. Les bibliothèques ainsi téléchargées peuvent ensuite être utilisées comme les bibliothèques normales. Dans le cas d'une collision de nom de code, Codebender donne la préférence à la version que vous avez téléchargée plutôt qu'à la version contenue dans Framework.

Ajout de code

Pour finir, il convient de noter ici que le code source de Codebender est ouvert. Les développeurs utilisent GitHub pour le partage de code – ouvrez [github] dans votre navigateur Web pour télécharger les différents modules vers votre ordinateur. Les utilisateurs peuvent soumettre aux développeurs des demandes de modification via d'Issue Tracker [IT] et ensuite voter pour leur mise en œuvre. Il est également possible, par la fenêtre de chat au bas de la page Web, de prendre contact directement avec le développeur « de service ».

Il n'y a pas pour le moment de contrats fixes qui permettent la gestion de projets non publics. On pourra, par le biais de support@codebender.cc, faire part de ses intérêts ; il se pourrait fort bien que des bêta-testeurs puissent, dès maintenant, essayer l'une ou l'autre fonction initialement prévue pour des clients payants.

En conclusion

Codebender sous-entend le stockage du code dans le Nuage : il n'existe pas pour le moment de moyen d'identifier un projet comme étant privé. De plus, les fichiers compilés sont en général plus volumineux de 5 à 15%.

Les développeurs qui travaillent sur des produits *open source* devraient penser à une migration : la fonction IntelliSense améliorée, le compilateur plus intelligent et l'hébergement automatique du code valent leur pesant d'or.

Les fournisseurs de bibliothèques et *shields* devraient, pour des raisons purement égoïstes, envisager un partenariat : les Grecs sont, à cause du système de bibliothèques (*library*), les heureux propriétaires d'une quantité monstrueuse de code d'application. On va au-devant de problème en cas de modification de l'API ; Codebender peut déterminer pour les clients payants quel pourcentage du code ne sera alors plus compilable. ◀

(140519 – version française : Guy Raedersdorf)

Liens

- [cb] www.codebender.cc
- [TSIP] <http://blog.tzikis.com/?p=454>
- [Lib] <https://codebender.cc/libraries>
- [IT] <http://feedback.codebender.cc/forums/165703-general>
- [github] <https://github.com/codebendercc>

Publicité

Une qualité professionnelle à prix discount!

reichelt.fr
elektronik

Les mini-PC

C'est le moment pour votre projet!

Plus puissant
grâce à quatre
CŒURS



Plus de 150 articles
Vous trouverez tout ce qu'il faut en matière
de cartes, composants et accessoires
sur www.reichelt.fr

Prix en € T.V.A. incluse, frais de port en sus | reichelt elektronik, Elektronikring 1, 26452 Sande (D)



RASPBERRY PI 2 B

38,50



Plus de 45 ans d'expérience

Envoi en 24 heures

Plus de 50 000 produits



+49 (0)4422 955-360

détecteur de cliquetis

J. T. van Es (Pays-Bas)

Un de mes amis construit des moteurs pour voitures de course et les personnalise. Il cherchait une meilleure méthode d'écoute des bruits du moteur, de préférence depuis sa position de travail au banc d'essai. Je lui propose alors d'utiliser des capteurs de cliquetis tels que ceux de son système électronique de gestion de moteur (*ECU, electronic control unit*). Sur le ton du scepticisme, il demande : « tu crois que ça marcherait ? » Je lui avoue que je n'en sais rien, mais que je vais essayer.

► une meilleure méthode d'écoute des bruits du moteur, de préférence depuis la position de travail au banc d'essai



Figure 1. Un détecteur de cliquetis se compose d'un disque de matériau piézoélectrique inséré entre quelques disques métalliques.

Je rentre à la maison, muni de quelques exemplaires de ces capteurs, dans l'espoir de découvrir ce qui se cache sous leur capot noir. L'examen de l'intérieur (**fig. 1**) ne m'encourage guère : une tranche de matériau piézoélectrique solidement prise en sandwich par quelques disques métalliques. Un trou au centre permet de boulonner le tout avec une vis M8 contre le bloc-moteur. La capacité de l'anneau doit faire un millier de picofarads.

Aujourd'hui, on en trouve facilement les caractéristiques [1], mais à l'époque, en 2007, elles n'étaient pas disponibles.

Aussi, ai-je collé un de ces bidules sur le cône d'un haut-parleur pour me faire une idée du signal qu'il pouvait produire. Une affaire vite réglée : il se comporte comme un microphone sur une large bande de fréquence. Un bon début, mais je ne sais

toujours rien de sa sensibilité ni de la grandeur du signal à mesurer.

La **figure 2** montre le schéma de l'amplificateur conçu pour ce capteur capacitif. J'ai donc choisi un amplificateur de charge à deux étages constitué d'amplificateurs opérationnels à entrée FET, apportant chacun un gain de l'ordre de dix. La bande passante à -3 dB va de 70 Hz à 36 kHz.

Dans le schéma, C1 et V1 représentent le capteur capacitif. R16 n'est là que pour la simulation, sans elle, il est impossible de déterminer le potentiel entre C1 et C7, lequel a pour but d'isoler de l'ampli op le courant de fuite du capteur. C1 et C2 déterminent le gain du premier étage, tandis que R14 et R15 fixent celui du second. Le réseau C2/R1 fixe la bande passante côté bas. Normalement, R1 devrait avoir une valeur de $33\text{ M}\Omega$, mais comme je n'en avais pas, j'ai ajouté à la sortie de l'ampli op le diviseur de tension R17/R18 pour atténuer de $1/3$ le signal vers R1. La fréquence de coupure supérieure est déterminée par R7 et C1.

Les mesures sur l'amplificateur confirment les valeurs de la simulation. Avec le capteur à l'entrée, il était manifestement trop sensible, bien au-delà de ce dont mon ami avait besoin : aucune amplification n'était nécessaire ! J'ai donc abandonné le petit boîtier de la **figure 3** au fond d'un tiroir.

Détecteur de cliquetis

Sur les automobiles et les motos, on utilise un détecteur de cliquetis (ou de saccade) pour éviter que ce phénomène, dû à la mise en combustion anticipée et incontrôlée du carburant, ou autoallumage, ne survienne dans le moteur à explosion.

Lors du processus normal d'allumage, on observe des vibrations normales du moteur, mais en cas de combustion par à-coups, les vibrations sont plus intenses dans un certain domaine de fréquence. Le signal correspondant du détecteur de cliquetis est filtré et traité dans le système de gestion moteur (*ECU, electronic control unit*), après quoi, le processeur peut déterminer s'il y a du cliquetis. Dans ce cas, il retarde le moment de l'allumage, souvent de 3 degrés. Quand le moteur ne cogne plus, il avance progressivement l'allumage par pas de 0,5 degré jusqu'au retour à la normale

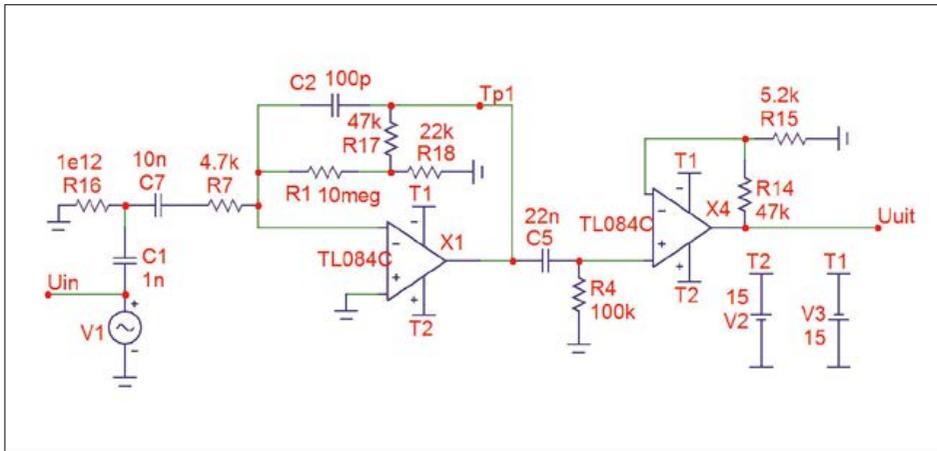


Figure 2. Le circuit d'amplification compte deux étages qui fournissent en tout un gain en tension de 100.

Début 2014, je l'en ai extrait pour le soumettre à de nouvelles mesures. Le bruit à la sortie s'élève 240 μ V, on en déduit qu'à l'entrée, il ne fait que 2,4 μ V. Bien emballé dans une feuille d'aluminium, le capteur est maintenant coincé dans mon étau d'établi. Mon oscilloscope est réglé sur 10 mV/div. Rien que de toucher la table donne de vigoureuses réactions. Y laisser tomber une allumette d'une hauteur de 2 cm produit une déviation de 40 mV. Même la parole est visible.

En voilà une sensibilité pour un capteur que l'on ne pourrait casser qu'avec brutalité d'un coup de hache, par exemple !

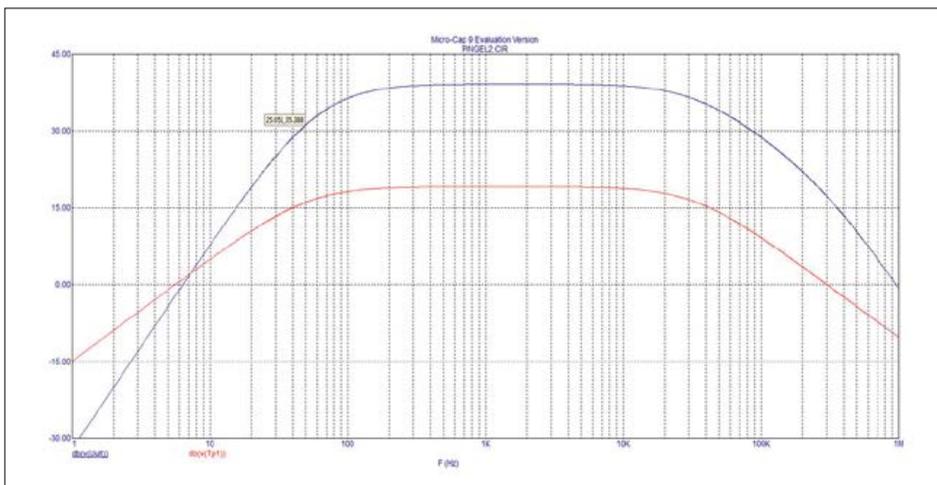


Figure 3. Comme résultat de la simulation du circuit, les caractéristiques de fréquence, en rouge à la sortie du 1^{er} amplificateur opérationnel, en bleu à la sortie du second.

Dans chaque automobile moderne, il y en a au moins un. Que d'expériences amusantes on peut réaliser avec ce genre de composant, comme cellule de lecture, comme stéthoscope ou plein d'autres choses. Je le garde précieusement. ◀

(140324 - version française : Robert Grignard)

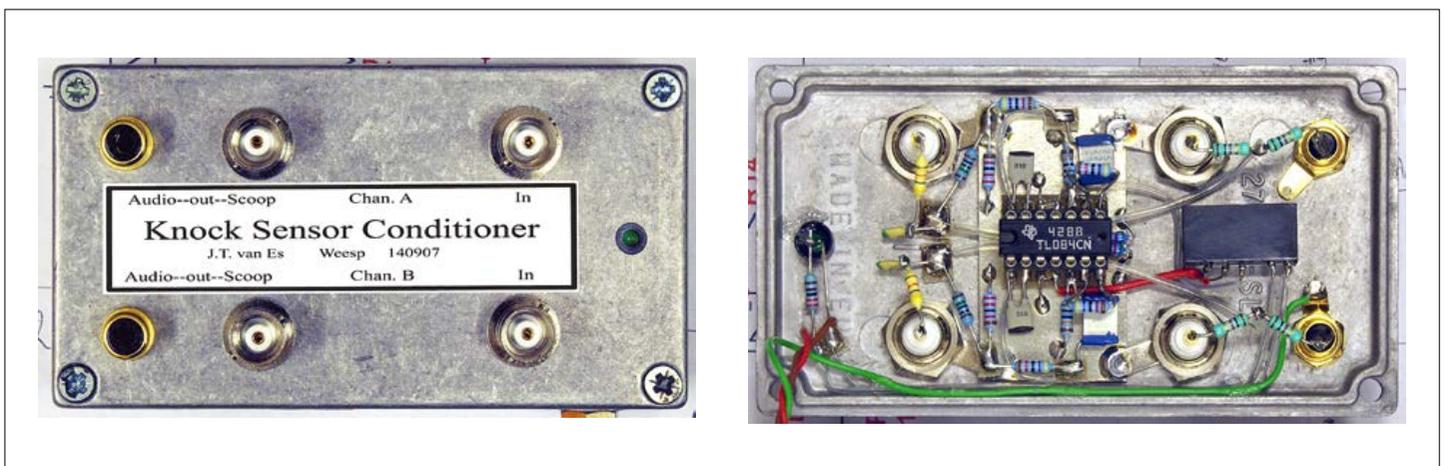


Figure 4. Le circuit que j'ai construit, je l'ai installé dans un boîtier en aluminium pour lui assurer un blindage optimal.

Lien

www.bosch-motorsport.de/media/catalog_resources/Knock_Sensor_KS-R_Datasheet_51_en_2779074187pdf.pdf

à la mémoire d'Yvon Doffagne,
montreur d'images qui a su rendre tangibles
les phénomènes physiques les plus abstraits

RESI & TRANSI[®] ne feront plus de bulles

Leur génial géniteur Yvon Doffagne ne décapuchonne plus son feutre depuis fin janvier. R&T ne chipoteront plus ni sur le sens des électrons ni sur celui de la vie. Leurs compagnons de tirage ont perdu le fil, la fière LEDY est débranchée et ce ballot de Condo complètement déchargé.

Nombreux sont nos lecteurs qui se souviennent d'Yvon comme le créateur, le dessinateur et le co-auteur* de la bande dessinée d'initiation à l'électronique dont Rési & Transi étaient les personnages principaux.

Passionné entre autres de théâtre et d'avion, Yvon excellait à expliquer par l'image des phénomènes difficiles à représenter. Il avait trouvé dans l'électronique un sujet de prédilection pour donner libre cours à sa fibre d'artiste pédagogue. Ses analogies notamment hydrauliques resteront inégalées. Des carrières d'électroniciens passionnés y ont pris leur source. J'ai eu le privilège et le plaisir de le côtoyer et de travailler avec lui dans les années 1980. D'abord sur ses deux albums *Échec aux mystères de l'électronique* et *Touche pas ma bécane* édités



par Elektor (en français, mais aussi en anglais, en allemand et en néerlandais). Puis sur la cinquantaine d'épisodes de deux pages de bande dessinée Rési & Transi en français parus dans le mensuel Elex, de 1989 à 1992... Je viens de les parcourir et je sors ébloui par l'allant et la fraîcheur intacte de ces 200 pages de BD ! Aux côtés d'un créateur de la trempe d'Yvon, nous n'avons peur de rien, il n'y a pas de sujet** aussi opaque soit-il qu'il n'ait osé aborder. Une fois, comme la rédaction d'Elex était en mal d'inspiration pour l'épisode du mois, Yvon a eu, faute de scénario électronique, l'idée généreuse de consacrer ses deux pages de BD à apprendre aux lecteurs à dessiner eux-mêmes les personnages Rési&Transi : « *Comme ça ils feront leur propre BD !* »

Mentionnons encore les magnifiques dessins qu'il avait faits pour la couverture d'Elex de janvier à mai 1991, avec pour thème les formes d'onde. En les revoyant aujourd'hui, ma satisfaction et mon admiration sont intactes, mais je ne peux plus le lui dire. Alors je m'adresse à Cookie, la délicate compagne d'Yvon et sa subtile coloriste, pour lui dire : **Merci l'Artiste !**

Denis Meyer

PS : Lecteurs de Rési&Transi, vous qui avez apprécié l'œuvre d'Yvon Doffagne, adressez-nous vos témoignages sous la forme qui vous conviendra et nous en ferons un recueil.

* avec Yves Causin

** Parmi les plus réussis à mon avis : les ondes (n°7, janv. 89), le potentiel (n°14, sept. 89), les harmoniques (n°28, déc. 90), les cascades (n°36, sept. 91), l'infra-rouge (n°37, oct. 91)... même *les bandes de valence* y sont passées. Le préféré d'Yvon était l'épisode sur les afficheurs à 7 segments... au gaz, en hommage au monde du théâtre (n°13, juil. 89). Et quand on n'a plus su quoi inventer, on s'est même risqué à faire un épisode *aphone* (n°27, nov. 90)... pas de texte dans les bulles, que du dessin. Et quel dessin !



L'homme Yvon Doffagne est évoqué ici dans les souvenirs de l'un de ses proches amis, Bob Safie, ancien directeur d'Elektor France

On ne peut pas ne pas aimer Yvon

Un artiste séduit par sa créativité. On l'admire. Tout le monde aimerait en connaître un.

J'ai beaucoup de chance d'avoir Yvon comme ami.

Ces dernières années, nous papotons souvent au sujet d'un de ses dessins quotidiens du *Candide** et refaisons gaiement le monde au téléphone ou par mél. Ou autour d'un verre, sur une terrasse au soleil, à l'occasion de vacances en famille dans son joli coin des Cévennes. Là nos échanges étaient interrompus toutes les deux minutes par les passants, leurs saluts cordiaux, des nouvelles truculentes, telle demande d'avis ou tel conseil amical... Yvon connaissait tout le monde.

Tout le monde appréciait Yvon.

Dans un petit étui à la ceinture, il portait toujours un Laguiole qu'il sortait au repas avec son fusil pour le rituel de l'affûtage, avant de le poser à droite de son assiette. Un jour, à la terrasse du *Cévenol* à Anduze, Yvon s'absente pour entrer chez le marchand d'à côté. À son retour, il me réclame un franc – oui, ça date ! – que je lui tends sans comprendre. Il me remet en échange un magnifique Laguiole à manche en corne noire : « Pour bien apprécier tes repas, il te faut ça, mais je ne peux pas te l'offrir, ça couperait notre amitié. Donc, tu me l'as acheté pour un franc. »

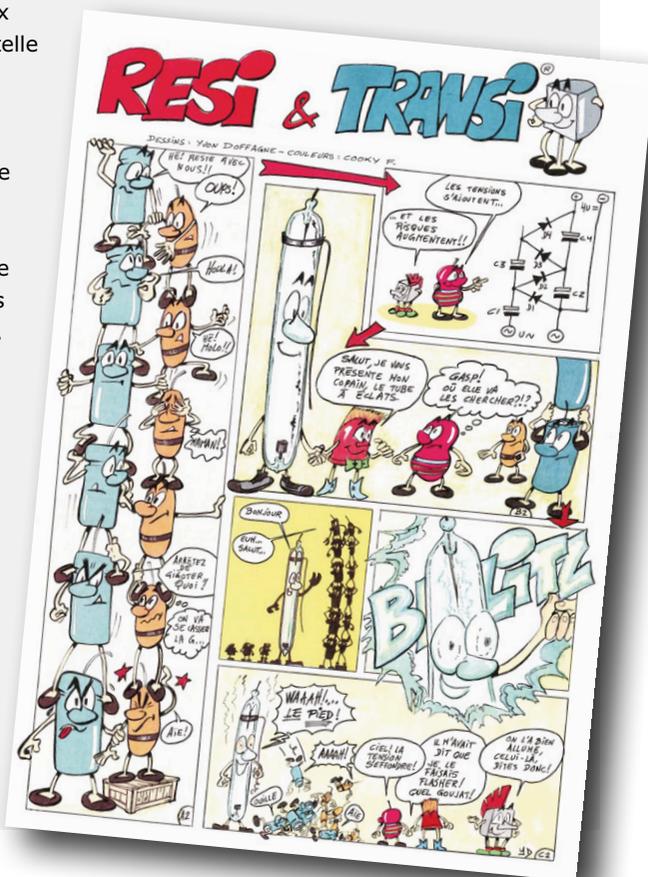
Je me souviens de nos deux participations au salon de la BD d'Angoulême. Pour moi c'était une première, pas pour Yvon. Il me guidait dans les cafés, les restos et tous les lieux envahis par ses confrères. Un soir, dans un bar bondé de dessinateurs, tous semblent le connaître et le saluent. Il se fraye un chemin et m'entraîne au comptoir, puis lance haut et fort en me désignant : « **C'est mon éditeur !** » C'est fou le nombre de gens qui ont voulu faire ma connaissance ce soir-là !

Yvon et Cookie ont officialisé leur union le jour où la comète Hale-Bopp devait passer près de la Terre. J'étais invité. C'était à un moment de ma vie où j'avais grand besoin d'affection... J'ai oublié la comète, jamais je n'oublierai ce week-end de fête à Anduze.

Bye Bye, Yvon

Bob Safie

* www.candide-graffiti.com



pilote de LED bon marché à haut rendement 0,5 A/33 V

Valentin Kulikov (FuturoLighting)

Voici le portrait en patte d'un pilote de sources à courant constant doté d'une entrée à MLI rapide. Simple, idéal pour piloter des LED de haute et moyenne puissance, ce module admet des tensions d'entrée de 8 à 33 V et son courant de sortie est configurable par pas de 0,1 à 0,5 A.

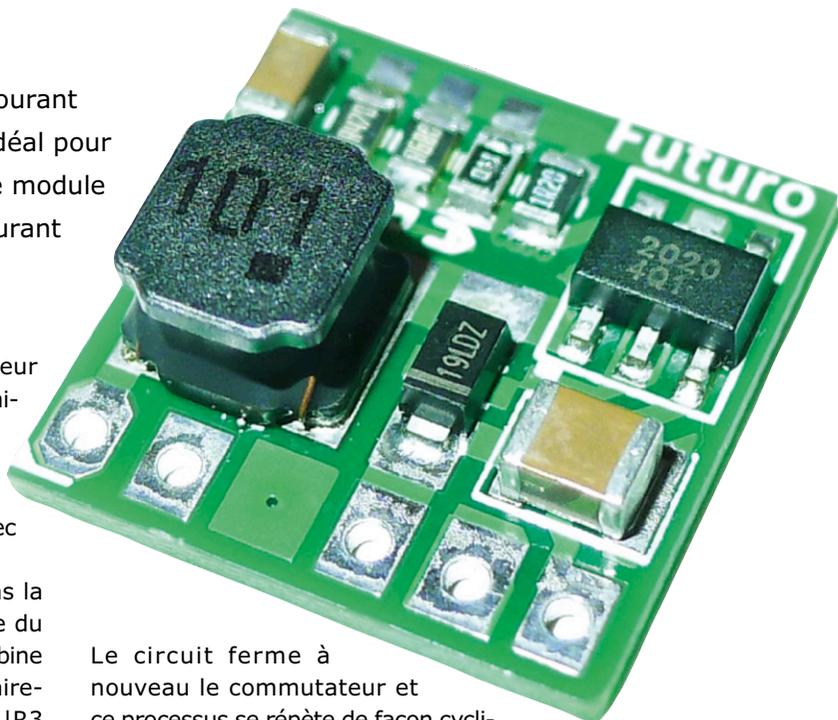
Le module (**fig. 1**) est construit autour du pilote abaisseur (*buck driver*) TS19376 (en boîtier SOT89-5) de Taiwan Semiconductor. La régulation du courant par hystérésis permet à la puce d'atteindre des rendements de plus de 90 % sans utiliser de compensations. Le courant de sortie est déterminé par les résistances en parallèle R1, R2 et R3 avec une pente de 0,13 Ω/A.

Le principe de la régulation par hystérésis est décrit dans la fiche technique [1]. En résumé : le commutateur interne du pilote TS19376 relie la tension d'entrée à la charge via la bobine L1. Le courant qui traverse cette bobine augmente linéairement et la différence de potentiel aux bornes de R1||R2||R3 est surveillée. Si l'hystérésis est de 15 % (19,5 mV), le circuit ouvre le commutateur interne lorsque le courant a provoqué une différence de potentiel de :

$$130 + V_{csn_hys} = 149,5 \text{ [mV]}$$

Le courant qui traverse la bobine et D1 décroît alors linéairement jusqu'à la valeur :

$$130 - V_{csn_hys} = 110,5 \text{ [mV]}$$



Le circuit ferme à nouveau le commutateur et ce processus se répète de façon cyclique (**fig. 2**).

La fréquence de commutation est déterminée par les valeurs du courant de sortie (I_{LED}), de la tension d'alimentation (V_{cc}), de la tension de sortie et de L1.

Gradation par MLI

Le courant moyen de la LED peut être commandé par MLI (*PWM*). Cette technique courante est facile à mettre en œuvre avec un microcontrôleur ou encore un temporisateur 555 (**fig. 3**). Le signal relié à l'entrée MLI du module doit être

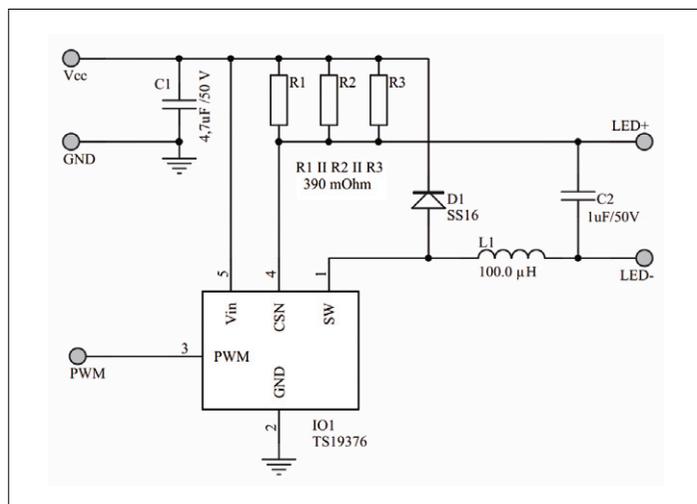


Figure 1. Schéma du pilote de LED.

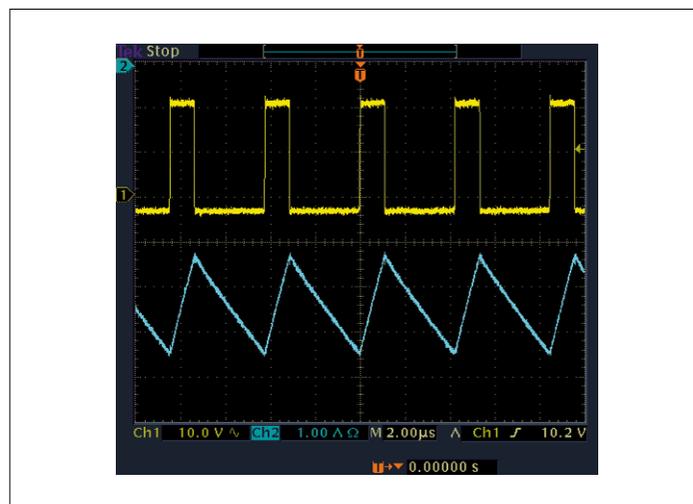


Figure 2. Courant et tension au nœud de commutation (GND de l'oscilloscope reliée à V_{cc}).

< 0,3 V pour le niveau bas et > 2 V pour le niveau haut (CMOS). Le TS19376 accepte des fréquences relativement élevées, donc une gradation par MLI rapide avec une résolution supérieure à 8 bits est possible.

Puisque l'entrée MLI possède une résistance de rappel vers le haut, sans MLI appliquée I_{LED} vaut l'intensité maximale. Pour éviter les papillotements, il est recommandé d'utiliser une fréquence MLI > 100 Hz.

Réalisation

Le TS19376 nécessite côté cuivre une surface de refroidissement thermiquement reliée à la face composants par des vias. Un condensateur d'entrée à faible ESR est requis pour supprimer les pointes de courant durant la commutation du pilote. Les valeurs recommandées pour C4 sont de 4,7 à 100 μ F, avec comme qualité diélectrique au moins X7R ou X5R. C1 doit être monté aussi près que possible des pastilles d'alimentation d'IO1. Les valeurs optimales pour L1 vont de 47 à 120 μ H. Une inductance faible convient mieux pour les courants forts, une inductance élevée convient mieux aux courants faibles, pour lesquels il n'y a pas de retard à la commutation. Les autres composants doivent être placés de façon à obtenir la boucle de commutation la plus « courte » possible, et donc le moins d'interférences électromagnétiques. Le départ de l'enroulement de la bobine doit aussi être relié au nœud de commutation (pastille SW d'IO1).

La valeur de D1 garantit de faibles courants de saturation à la température max. de fonctionnement et un faible t_{tr} . La tension directe de D1 affecte le rendement ; une valeur de V_d plus faible donne un meilleur rendement et une dissipation de chaleur moindre.

Une marge de 30 % est recommandée pour le courant direct max. de la diode par rapport à I_{LED} . Notre choix : SS16 (1 A/60 V) de Taiwan Semiconductor.

C2 réduit d'autant plus les ondulations du courant de sortie et la fréquence MLI que sa capacité est grande.

Le TS19376 se désactive lorsque sa température atteint 150 °C, se réactive lorsqu'elle repasse sous 115 °C. Cette protection évite de brûler le circuit imprimé. Un dissipateur peut être fixé au module avec du ruban thermoconducteur double face (p. ex. du Bond-Ply de Bergquist). Selon l'application visée, on peut ajouter au module un filtre EMI et une protection contre les inversions de polarité (PMOS de commutation p. ex.). Le module est implanté sur une carte double face FR4 de 16x16 mm et 5,5 mm d'épaisseur.

Caractéristiques :

- Topologie : Buck
- Régulation : hystérésis
- Tension d'entrée : 8 à 33 VCC
- Courant de sortie : 100 à 500 mA
- Fréquence de commutation : 1 MHz max.
- Pente du courant : 0,13 Ω /A
- Dimensions : 16 x 16 x 5,5 mm
- Poids : 1,6 g
- Protection thermique
- Protection contre les surintensités
- Gradation par MLI jusqu'à 20 kHz

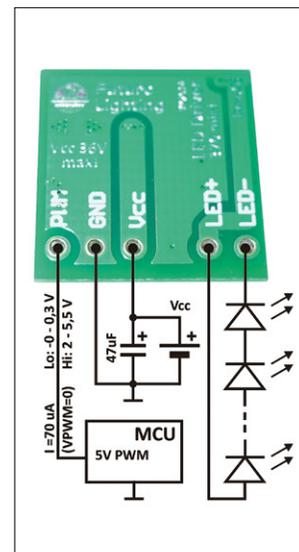


Figure 3. Diagramme de connexion du pilote de LED.

Conclusion

Les applications possibles de ce pilote de LED vont de la commande de LED à moyenne et haute puissance au chargement de batterie, en passant par tout ce qui exige la présence d'une source de courant constant. Le nombre maximal de LED chaînées est déterminé par la plus petite tension d'alimentation (V_{cc}) permise. Garder la tension totale V_{LED} proche de V_{cc} donne les meilleurs rendements (fig. 4). Pour $V_{cc} = 12$ V p. ex., trois LED en série est un bon choix ($V_{LEDD} \sim 3$ V). Toutes les mesures ont été automatisées et réalisées à température ambiante.

Le pilote de LED avec sélection du courant de sortie de 0,1 à 0,5 A peut être commandé chez FuturoLighting [2]. Le TS19376 et la diode SS16 sont disponibles (en quantité minimale) auprès de Microdis Electronics [3], distributeur agréé de Taiwan Semiconductor.

Pour leur soutien, je remercie M. Bilik (Würth Elektronik) et M. Reguli (Microdis Electronics). ◀

(150048 - version française : Hervé Moreau)

Liens

- [1] www.taiwansemi.com/products/datasheet/TS19376_A12.pdf
- [2] www.fulit.eu/store
- [3] www.microdis.net

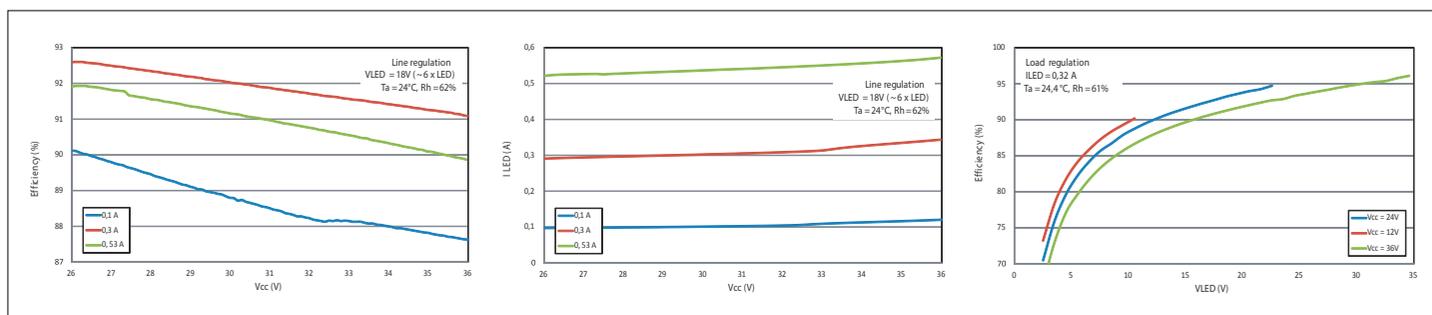
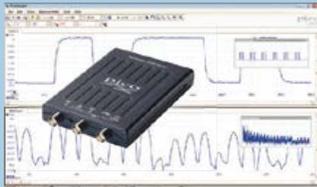


Figure 4. Caractéristiques du pilote sous différentes conditions.

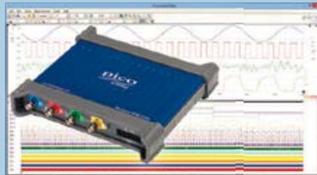
OSCILLOSCOPES PC

Petit Prix



- Bande Passante 10 MHz à 200 MHz
- Echantillonnage de 100Mé à 1Gé/s
- Résolution 8 bits (améliorée à 12 bits)
- 8 to 48 kS buffer memory
- Mémoire tampon de 8 à 48 kS
- Dès 96 €

Signaux Mixtes



- 2 ou 4 voies analogiques + 16 logiques
- Bande passante de 50 à 200 MHz
- Résolution 8 bits (améliorée à 12 bits)
- Mémoire tampon de 64 à 512 Mé.
- Alimentation par USB ou bloc secteur
- Dès 604 €

8 voies



- Bande Passante 20 MHz
- Echantillonnage 80 Mé/s
- Résolution 12 bits (améliorée à 16 bits)
- Mémoire tampon 256 Mé.
- Alimentation par USB
- Pour 1688 € seulement

Résolution Flexible



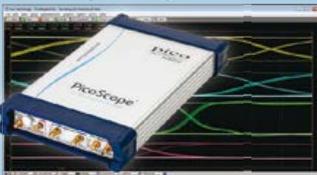
- 8, 12, 14, 15, et 16 bits dans un même appareil
- Bande passante de 60 à 200 MHz
- Echantillonnage à 250 Mé/s to 1 Gé/s
- Mémoire tampon de 8 à 512 Mé
- Alimentation par USB ou bloc secteur
- Dès 846 €

Mémoire 2Giga-échantillons



- Bande passante de 250 MHz à 1GHz
- Echantillonnage à 5 Gé/s
- Résolution 8 bits (améliorée à 12 bits)
- Mémoire tampon de 256 Me à 2Gé
- Alimentation par bloc secteur
- Dès 2414 €

Sampling 20 GHz



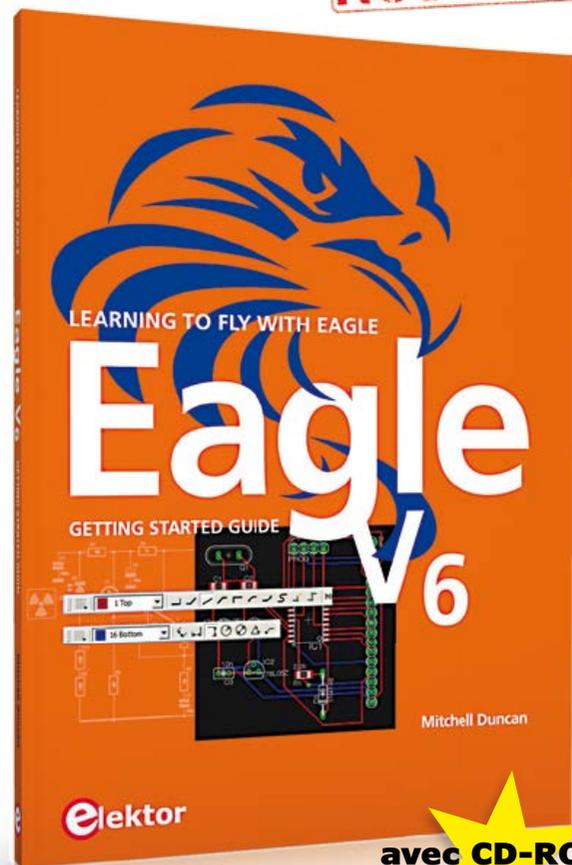
- Bande passante du continu à 20 GHz
- Temps de montée 17.5 pS
- 16 bits, plage dynamique de 60 dB
- Alimentation par bloc secteur
- Générateur de signaux, reconstruction d'horloge, TDR/TDT diff.
- Dès 10,996 €

Logiciel complet inclus en standard avec décodage et analyse de bus série (CAN, LIN, RS232, I2C, I2S, SPI, FlexRay), mémoire segmentée, test de masque, analyse de spectre, kit de développement logiciel (SDK) tout en standard avec mises à jour gratuites.
Cinq ans de garantie pour les oscilloscopes temps-réel.
Deux ans de garantie pour les oscilloscopes à sampling.

www.picotech.com/PS419

Guide de démarrage pour Eagle V6

NOUVEAU



avec CD-ROM
livre entièrement en couleur

LEARNING TO FLY WITH Eagle V6 GETTING STARTED GUIDE

Que vous soyez électronicien amateur ou professionnel, ce guide de prise en main vous initiera rapidement au logiciel de conception de circuits imprimés EAGLE de CadSoft.

Ce livre (en anglais) a été écrit pour tous ceux qui souhaitent découvrir les possibilités d'EAGLE. L'auteur s'adresse aussi bien au novice en conception de circuits imprimés qu'au professionnel déjà familier d'un autre logiciel de CAO, mais qui envisage de passer à EAGLE.

Ce livre est accompagné d'un CD-ROM gracieusement fourni par CadSoft Computer GmbH (www.cadsoft.de). Il contient une version d'Eagle pour Windows, Linux et Mac.

ISBN 978-1-907920-20-2

206 pages - 34,50 €

Clemens Valens (Elektor.Labs)



Le paradoxe du concepteur

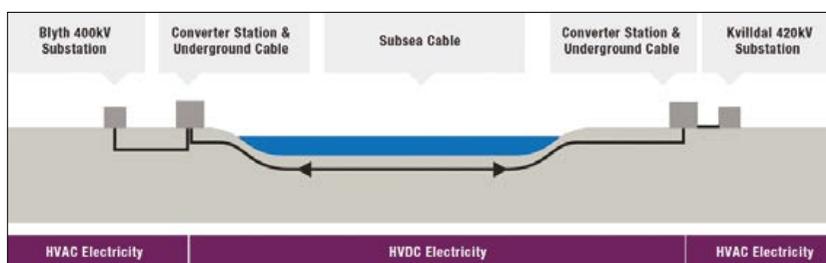
Une des difficultés du concepteur est d'identifier les composants qui lui permettront de réaliser son circuit. Si en leur temps Tesla et Edison n'avaient à leur disposition que des résistances, des condensateurs, des bobines et, plus tard, des diodes, les composants se comptent aujourd'hui par dizaines de milliers. Il en existe pour toutes les fonctions imaginables et, histoire d'en faciliter le choix, ils sont souvent produits par plusieurs constructeurs et sous différents boîtiers. Si par chance vous dénicher le composant idéal proposé en version unique par un seul fabricant, il est probable que votre supérieur vous répondra qu'il est trop risqué d'utiliser un composant fourni par une seule source. Une autre fois votre chef refusera d'ajouter un nouveau produit à ses stocks. Alors vous vous rabattez sur ce qu'il y a déjà et pas sur ce qu'il pourrait y avoir (de mieux). Une fois le projet (presque) terminé, le circuit imprimé commandé, vous découvrez qu'il aurait été plus facile, plus économique, plus rapide, d'utiliser le AW50M3 d'Acme. Que faire ? Soit il

sera trop tard pour modifier votre schéma, soit vous serez autorisé à la changer. Comment sortir de ce sac de noeuds ? Vous ne pouvez pas.

Concevoir c'est comme arpenter l'escalier de Penrose à la recherche de la dernière marche.

Une sacrée rallonge

Le *NSN Interconnector* devrait être inauguré en 2020. Ce câble d'alimentation sous-marin de 515 kV CC sera alors le plus long du monde et reliera Kvitlidal, en Norvège, à Blyth, au Royaume-Uni. Il délivrera la colossale puissance de 1,4 GW, de quoi alimenter 700 000 foyers. Le câble sera enfoui dans la croûte océanique entre 1 et 3 m de profondeur. Coût estimé du projet : entre 1,5 et 2 milliards d'euros, soit environ 1,5 €/W (en supposant que le budget ne sera pas respecté). Les interconnecteurs servent à relier les réseaux électriques de pays différents et il s'en construit tout autour du globe. Une interconnexion d'un millier de kilomètres entre le Royaume-Uni et l'Islande est d'ailleurs à l'étude. Elle permettrait au Royaume-Uni (et à l'Europe) d'accéder aux abondantes ressources géothermiques, hydrauliques et éoliennes de l'Islande.



Bienvenue dans Elektor Labs

C'est dans Elektor Labs que les projets grands et petits, analogiques et numériques, vieux jeu et *méga cool* prennent forme pêle-mêle pour vous permettre de les réaliser à votre goût.

Notre histoire

Elektor Labs existe depuis les années 1970. En ce temps-là, c'est la même personne qui soudait les circuits et écrivait les articles. Le labo n'a pas seulement vu arriver et passer le transistor, le circuit intégré, le microprocesseur et les composants montés en surface, mais il s'est toujours porté à l'avant-garde des nouvelles vagues technologiques pour diffuser la bonne parole et les rendre accessibles au plus grand nombre.

Notre offre : La célébrité



La plupart des électroniciens créatifs sont modestes. Trop discrets. Ce n'est pas parce qu'elle est griffonnée sur un rond de bière qu'une trouvaille épatante ne mériterait pas l'attention. Elektor Labs vous aide à affiner le fruit de vos études jusqu'à la perfection. Notre rédaction et nos illustrateurs donneront à votre création l'ampleur éditoriale qu'elle mérite, mais c'est toujours votre nom qui figurera en tête de l'article, car c'est *vous* l'auteur du projet. En plus, vous serez rémunéré honorablement, même si le plus grand honneur restera celui d'être publié en plusieurs langues et lu dans le monde entier. Auteurs de livres, de blogs ou de vidéos, vous ne serez pas moins bien traités. Étudiant(e)s et jeunes électronicien(ne)s, une publication dans une revue comme Elektor ne déparerait pas votre C.V. !

Notre équipement

Il y a l'eau courante, le gaz, l'électricité et des machines à café. Trois des plus belles pièces du château d'Elektor sont réservées au labo, mais nous avons bien du mal à ne pas envahir le reste.

Nos paillasses débordent de cartes, de composants en vrac, d'accessoires mécaniques, de prototypes et d'un fatras ingérable. L'appareillage lourd est heureusement dans un local séparé.

Notre site elektor-labs.com

C'est un portail bi-directionnel pour mettre en vedette votre projet et pour suivre ceux des autres. Venez vous informer de ce qui se passe chez les autres électroniciens comme vous, placez vos contributions, et éclatez-vous dans l'interaction. Les projets les plus suivis sont mûris par Elektor Labs et certains deviennent des articles dans le magazine Elektor. Toujours sous votre nom et pour votre bénéfice.

354

propositions

42

études en cours

154

projets achevés

550

projets en tout

Nos produits

Nos produits sont dans le magazine et sur nos sites. Nous produisons texte et illustrations utilisés par la rédaction pour assembler les articles, mais surtout les circuits imprimés, certains assemblés prêts à l'emploi, ainsi que le logiciel et les composants programmés, des kits, des modules, des outils, des accessoires, des vidéos et un service d'info technique.

Nos webinaires

Autrefois nos ingénieurs les plus bavards testaient leurs prototypes en discutant. À la vue d'un micro, ils se taisaient. Pour les faire parler sur *elektor.tv*, il a fallu un talent de reporter. *Elektor Labs* présente aussi des webinaires où l'on parle bien et beaucoup. Ils sont annoncés dans notre lettre d'information *Elektor.POST!*

Nos experts et nos concepteurs

L'équipe est formée d'électroniciens expérimentés de tout plumage qui, avec ou sans prestigieux diplômes, cumulent quelque 200 années d'expérience en électronique. En plus, *Labs* est au cœur d'un réseau d'experts consultés ponctuellement, notamment quand ça coince.

Nos principes

Les réalisations et les produits sortis des tuyaux de *Labs* répondent tous à des exigences sévères. Les produits des articles du magazine estampillés *LABS* doivent fonctionner avec l'appareillage étalonné disponible au labo. La correspondance entre schéma et liste de composants doit être parfaite. Les kits sont soumis à des tests par échantillonnage périodique. Nous suivons la directive *ROHS* et les autres normes de sécurité applicables à notre situation. Les erreurs constatées font l'objet d'une publication.

analyseur UART/RS232

pour tenir à l'œil le flux sériel

Thomas Schlott, Francisco Ramirez & Thomas Scherer

Pour l'échange de données entre appareils, on privilégie l'UART. Cette interface est d'usage facile et quantité de puces de convertisseurs peuvent s'y associer. Mais si, en cours de développement ou de test, vous vous apercevez que quelque chose cloche dans la communication, vous pouvez vous attendre à un dépannage épineux. Sauf si vous avez notre analyseur sériel UART qui vous simplifie la vie en enregistrant les octets dans les deux sens à la milliseconde près. Des raccordements variés et une séparation galvanique lui permettent de s'insinuer sans risque dans toute liaison sérielle à niveaux TTL ou RS232.

Quand un système électronique ne fonctionne pas comme il faut, on peut s'arracher les cheveux en vain si la cause

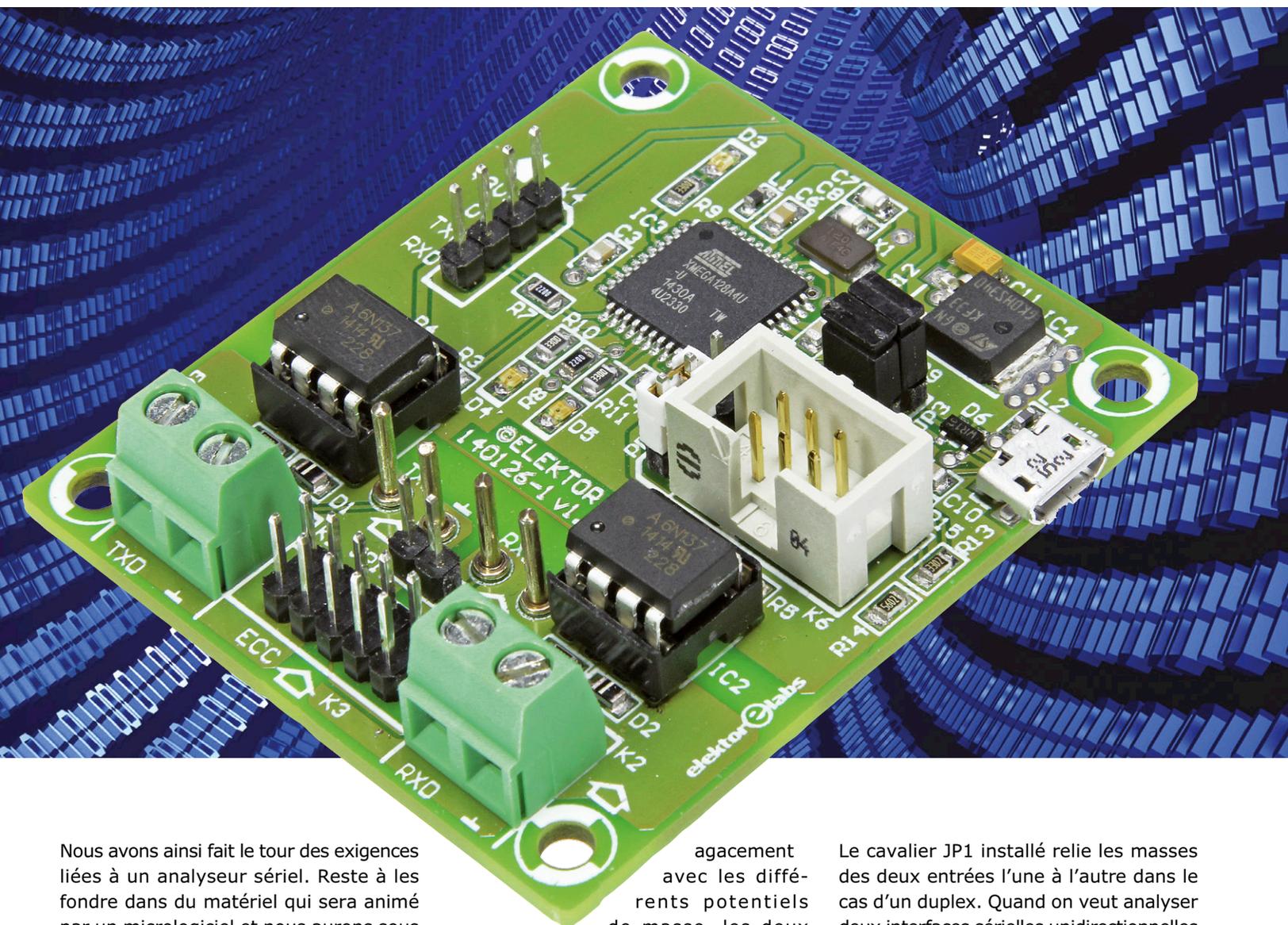
est intermittente ou non reproductible. Lorsque vous intervenez, les données « fautives » sont déjà parties sur le

câble et se sont évanouies dans le Nirvana numérique. Pour comprendre ce qui s'est passé, il faut noter les données disponibles et de préférence avec des repères temporels.

Mais ce n'est pas tout, deux appareils reliés l'un à l'autre par interface sérielle peuvent envoyer chacun des octets, la liaison est alors bidirectionnelle, en duplex. L'analyseur sériel doit donc disposer de deux canaux de données pour arriver à surveiller complètement le trafic dans les deux sens. Et les enregistrer ne suffit pas, il doit aussi analyser les données. Il lui faut pour cela utiliser un PC, de préférence par USB, soyons modernes.

Caractéristiques techniques

- Deux entrées sérielles isolées par photocoupleur pour une liaison bidirectionnelle
- Masse des entrées séparable
- Transmission des données au PC par USB
- Alimentation directe par USB
- Interface utilisateur par commandes sur terminal
- Débits binaires jusqu'à 230 kBd
- Enregistre jusqu'à 900 horodatages
- Durée d'enregistrement jusqu'à 2h20 (h:mm)
- Résolution 1 ms



Nous avons ainsi fait le tour des exigences liées à un analyseur sériel. Reste à les fondre dans du matériel qui sera animé par un micrologiciel et nous aurons sous la main un ustensile pas trop cher, mais extrêmement utile.

Matériel et ressources

Aucun doute, les fonctions énumérées réclament un microcontrôleur qui récolte les données, leur ajoute un repère temporel, les stocke dans un tampon et les transmet à un PC. Les exigences en matériel sont ainsi modiques : il nous faut deux U(S)ART, l'USB, une horloge précise et quelques E/S à usage général (*GPIO*). Un contrôleur AVR actuel doté d'une interface USB embarquée est taillé sur mesure pour cette fonction, tout particulièrement s'il dispose d'une mémoire interne suffisante. Le choix s'est porté sans hésitation sur l'ATxmega128a4u [1] en boîtier CMS dont on peut encore souder à la main les 44 broches.

En regard de la capacité du circuit complet de la **fig. 1**, l'analyseur sériel est assez sobre. De manière à éviter tout

agacement avec les différents potentiels de masse, les deux entrées sérielles sont isolées séparément de l'électronique de l'analyseur en même temps que du PC au moyen de photocoupleurs numériques particulièrement rapides du type 6N137. Lors des tests, ces photocoupleurs se sont montrés stables jusqu'à un débit binaire de 230 kbd. Le micrologiciel n'a pas pu suivre les débits supérieurs du câble FTDI de 460, voire 920 kbd. Les deux FET T1 et T2, configurés en source de courant, transforment des niveaux d'entrée jusqu'à 30 V et un courant de 3 mA environ. Les signaux d'entrée RXD et TXD peuvent être conduits depuis les lignes ou l'interface à analyser par des fils volants jusqu'aux bornes à vis K1 et K2. Quant à K3, il sert à analyser une liaison UART qui passe par un *Embedded Communication Connector* (ECC) [2]. Donc par exemple entre un *shield* d'extension Elektor et un convertisseur RS485, un module radio 433 MHz ou le convertisseur USB proposé par Elektor. Plus sur ces sujets dans les encadrés.

Le cavalier JP1 installé relie les masses des deux entrées l'une à l'autre dans le cas d'un duplex. Quand on veut analyser deux interfaces sérielles unidirectionnelles différentes au lieu d'une bidirectionnelle, on enlève JP1. On peut prélever sur K4 les données sérielles après isolation galvanique et mise en forme, sur une sonde d'oscilloscope par exemple.

Le contrôleur IC3 prend connaissance par ses broches 12 et 16 des signaux sériels. Il dispose de suffisamment de broches pour répercuter directement sur trois témoins, les LED D3 à D5, l'état actuel des lignes. Pour la fonction d'enregistrement des données sérielles, le quartz X1 est indispensable.

À droite de la figure 1, K6 est le connecteur de programmation à six contacts que l'on peut utiliser, en fonction de JP4, en mode ISP ou PDI. Les signaux sériels de la prise USB K5 sont directement reliés aux broches 26 et 27 de IC3. La matrice de diodes D6 protège l'électronique des surtensions comme des tensions négatives. Lors de la programmation, il faut

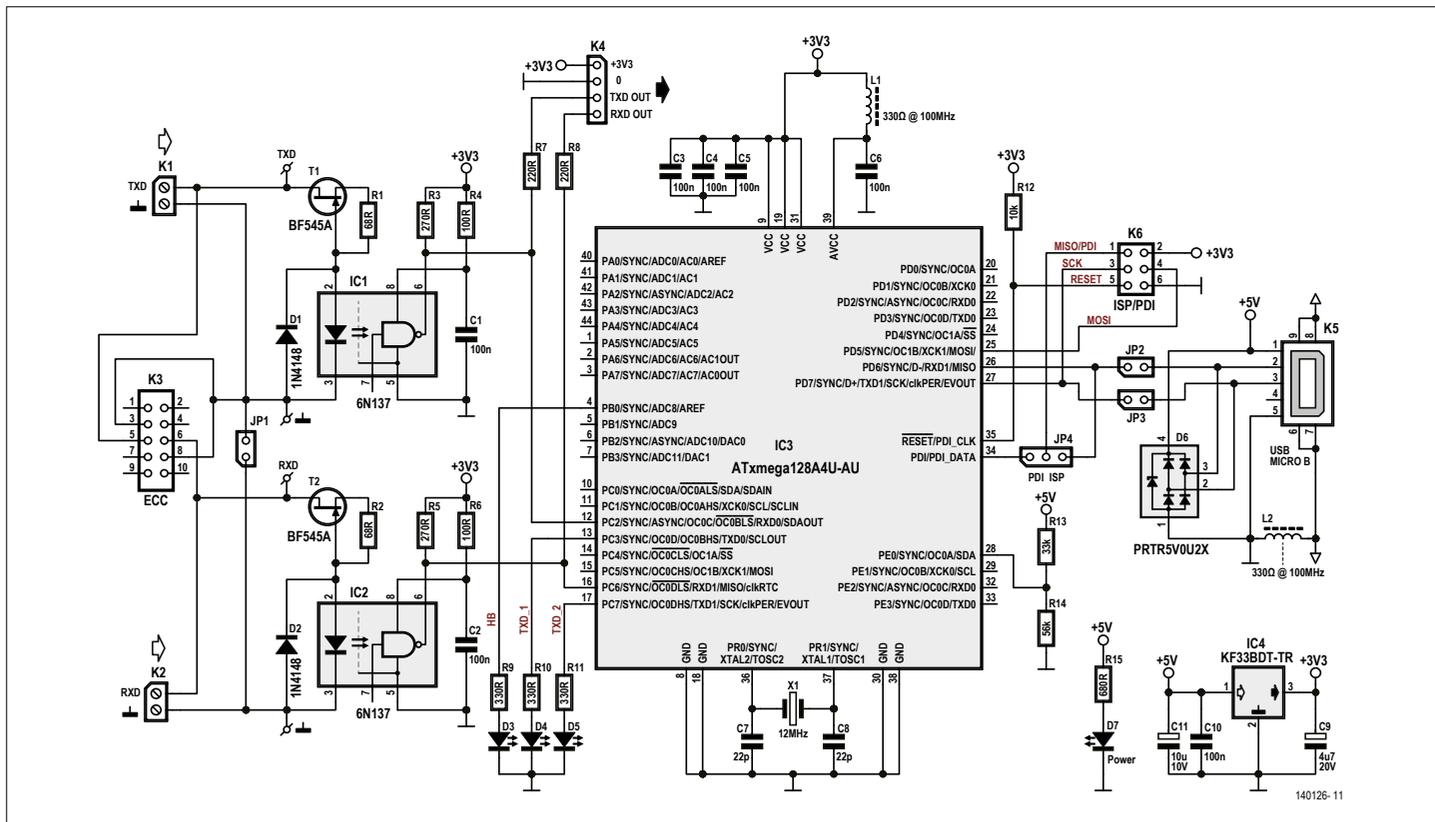


Figure 1. Schéma de l'analyseur sériel pour UART/RS232.

impérativement retirer les cavaliers JP2 et JP3 et ne pas oublier de les réinstaller après !

Les contrôleurs de la famille Xmega d'Atmel détiennent des caractéristiques supérieures à leurs prédécesseurs de la série

ATmega : des performances accrues et une vitesse plus grande, mais ils ne supportent pas la tension de 5 V. Aussi, pour pouvoir l'alimenter sur la prise USB, il lui faut un convertisseur de 5 à 3,3 V, c'est IC4. La LED D7 s'allume quand l'analyseur est ainsi raccordé à un PC en service. La tension d'alimentation de 5 V, préalablement soumise au diviseur R13/R14, est appliquée à l'entrée analogique de IC3, broche 28. De cette façon, l'analyseur peut vérifier si le PC est opérationnel ou si des perturbations sont à prévoir. Cette fonction n'est pas encore implémentée dans la version actuelle du micrologiciel.

L'horodatage, le repère temporel

Le repère temporel occupe en interne 23 bits. Il permet une résolution en millisecondes sur un intervalle de temps de 2 h et 20 mn. Le format de l'horodatage est **a:bb:cc:ddd**, avec :

- a : heures, 0 à 2
- bb : minutes, 00 à 59
- cc : secondes, 00 à 59
- ddd : millisecondes, 000 à 999

Combiné à l'octet de date, un horodatage complet nécessite 4 octets. Dans les 8 Ko de SRAM du contrôleur, à côté des variables et autres données du programme, il y a de la place pour 900 horodatages complets.

Micrologiciel et fonctions

Un circuit à microcontrôleur ne fait rien d'utile sans un logiciel pour exécuter les fonctions requises. Il a été développé avec le Studio 6.2 d'Atmel, l'original selon les idées de notre lecteur Thomas Schlott. Notre pigiste Francisco Ramirez l'a ensuite optimisé, notamment avec une liaison USB CDC (*Communication Device Class*, cf. ci-dessous). Le système d'exploitation du PC considère une communication par USB CDC de la même manière qu'avec une interface sérielle classique, un port COM virtuel, qui permet d'utiliser un pro-

gramme de terminal pour échanger des caractères par USB.

Comme d'habitude, ce logiciel, codes source et hexadécimal, est disponible gratuitement sur le site à la page de ce projet [3]. Les fonctions y sont commentées et transformées en fichier de données (*.c) et leurs définitions en fichiers d'en-tête (*.h) correspondants. Les tâches concrètes du micrologiciel sont les suivantes.

- Écriture en tampon des données sérielles entrantes, déclenchée par interruption.
- Traitement des commandes USB déclenché par interruption.
- La fonction principale du micrologiciel se répartit en deux tâches :
 - Initialisation et préparation de la mémoire pour les variables et les modules GPIO, temporisateur, USART, horloge, USB, EPROM, mémoire Flash et RAM. La RAM prend les variables, la Flash contient le code plus les chaînes prédéfinies et l'EEPROM, les derniers réglages.
 - Lancement de la boucle principale.

- La LED D3, dite cœur battant, clignote pour signaler que l'électronique est active.
- Les LED D4 (canal 1) et D5 (canal 2) indiquent quand des données transitent sur chacun des canaux.
- En vue de l'affichage des données en console, les sommes des infos de chaque canal, accompagnées de repères temporels, sont lues en mémoire et transmises par USB.
- Stockage des réglages, notamment le format de sortie des données. On peut aussi changer de mode au moyen de commandes (cf. encadré Commandes).

Subtilités du logiciel

Pour compléter l'interface USB, Francisco Ramirez a dû se procurer auprès d'Atmel le logiciel Framework ASF [4] qui contient une pile USB pour la communication USB CDC. Mieux valait ne pas essayer de décortiquer l'ASF pour en extraire les parties nécessaires au projet. Il a pris l'ASF tel quel comme base du projet. On commence avec un nouveau projet par *User-Board-Template* pour laisser l'assistant ASF du Studio d'Atmel assembler les modules requis. Pour ce projet, il fallait ceci :

- Generic Board Support
- GPIO – Entrées/sorties d'usage général
- IOPORT – Contrôleur de port d'entrée/sortie
- Implémentation du System Clock Control – XMEGA-A1U/A3U/A4U/B/C
- TC – Timer Counter
- USB-Device CDC

Dès l'abord, l'ASF est difficile à comprendre, il y a bien une grande quantité de code préparé et beaucoup de documentation, mais tout cela est réparti sur de nombreux fichiers. Un simple exemple, la mise en œuvre de l'horloge système s'opère en ASF de manière inhabituelle. On s'aperçoit qu'il faut définir une horloge indépendamment pour chaque module périphérique du contrôleur. La tentative d'utiliser l'oscillateur interne à 32 kHz pour l'USB et comme horloge système a échoué. En revanche, on a pu sans difficulté exploiter le quartz externe à 12 MHz comme horloge mère. Francisco Ramirez raconte son expérience avec ASF sur notre site elektor-labs.com [5].



configurer avec des commandes simples

Construction et mise en service

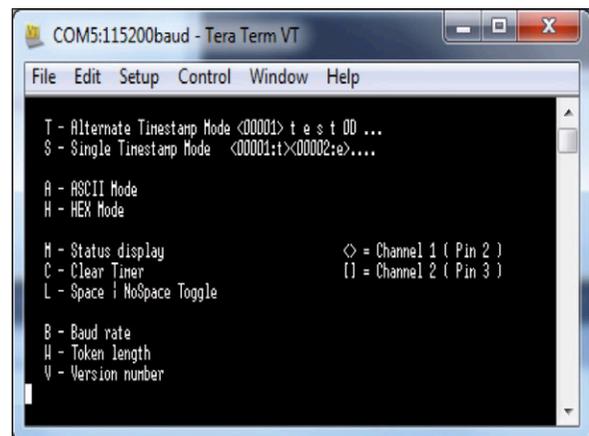
Elektor a réalisé un petit circuit imprimé (à la **fig. 2**) dont on voit le prototype construit par le laboratoire à la **fig. 3**.

Comme tous les composants jusqu'aux embases à picots, le photocoupleur, les prises et les bornes à vis sont des CMS, il faut beaucoup de soin lors du soudage,

Commandes

On configure les paramètres nécessaires à partir d'un programme de terminal ordinaire comme HTerm, TeraTerm, etc. au moyen de commandes connues de l'analyseur sériel.

Voici les commandes prévues dans l'interface utilisateur.



- S : Single Time Stamp Mode – imprime individuellement chaque caractère sur une nouvelle ligne avec son horodatage (*Time Stamp*).
- T : Alternate Time Stamp Mode – imprime l'horodatage suivi des caractères qui arrivent. Un nouvel horodatage, sur une nouvelle ligne, n'intervient que lors d'une réception sur l'autre canal.
- A : ASCII Mode – imprime les données reçues sous forme ASCII.
- H : HEX Mode – imprime les données reçues au format hexadécimal.
- E : Execute Line Feed – le caractère <LF> est passé tel quel au programme de terminal, il provoque un saut de ligne si le réglage le prescrit.
- M : Show Status – montre l'état actuel des divers réglages.
- C : Clear Timer – réinitialise l'horodatage.
- L : Space / No Space Toggle – détermine s'il faut ou non un espace entre les caractères.
- B : Baud Rate – choix dans une liste du débit binaire pour l'interface sérielle.
- W : Token length – choix de la longueur du jeton RS232.
- I : Inversion – à chaque commande, inverse le niveau du canal désigné ; dans une liaison RS232, les deux canaux doivent être *not inverted*, en TTL-UART sur *inverted*.
- V : Version number – donne le numéro de version du logiciel.

Pour une commande qui réclame un paramètre, saisir d'abord le caractère de la commande, puis choisir le paramètre dans la liste qui apparaît.

Il faut aussi savoir que le signe typographique qui entoure les données diffère d'un canal à l'autre, pour les distinguer :

- <> : canal 1 – les symboles de différence enclosent horodatages et données reçus du canal 1.
- [] : canal 2 – les crochets enclosent horodatages et données reçus du canal 2.

en commençant par IC3 et IC4, puis les résistances et les condensateurs. Comme ces derniers sont au format 0805, on peut encore les souder convenablement sans loupe. Quant aux diodes, LED et condensateurs au tantale, il y a lieu de respecter leur polarité.

Comme l'étage d'entrée est équipé de photocoupleurs, il est aisé de vérifier à l'oscilloscope sur K4 la présence effective de signaux sériels sur les deux entrées RXD et TXD. Si vous ne disposez pas d'un programmeur adapté, vous pouvez vous procurer à l'e-choppe [3] un microcontrôleur déjà programmé. Et si vous ne tenez pas à souder vous-même, il y a le module d'analyseur sériel assemblé, prêt à l'emploi, à la même adresse.

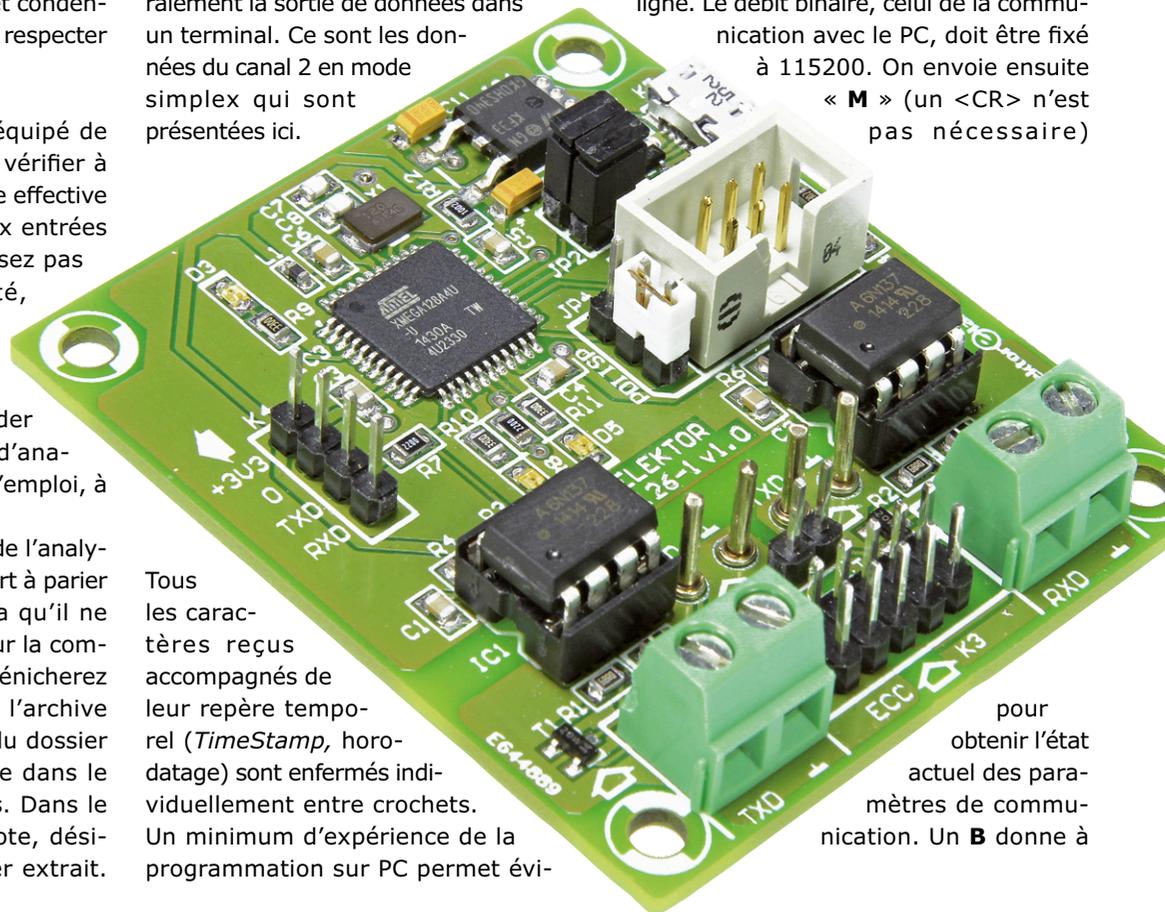
Lors du premier branchement de l'analyseur sériel sur votre PC, il y a fort à parier que Windows vous annoncera qu'il ne peut pas trouver de pilote (pour la communication USB CDC). Vous dénicherez le fichier nécessaire .inf dans l'archive .zip en [3]. Après extraction du dossier de l'archive, cherchez la carte dans le gestionnaire de périphériques. Dans le dialogue d'installation du pilote, désignez comme source le dossier extrait.

Après l'installation, l'analyseur sériel s'affiche avec comme type « *Communication Device Class ASF Example* ».

La **fig. 4** montre à quoi ressemble généralement la sortie de données dans un terminal. Ce sont les données du canal 2 en mode simplex qui sont présentées ici.

Tous les caractères reçus accompagnés de leur repère temporel (*TimeStamp*, horodatage) sont enfermés individuellement entre crochets. Un minimum d'expérience de la programmation sur PC permet évi-

demment d'écrire un logiciel qui arrange les données reçues pour un tableau. On peut aussi régler le terminal pour que le caractère <LF> produise un saut de ligne. Le débit binaire, celui de la communication avec le PC, doit être fixé à 115200. On envoie ensuite « **M** » (un <CR> n'est pas nécessaire)



pour obtenir l'état actuel des paramètres de communication. Un **B** donne à

Liste des composants

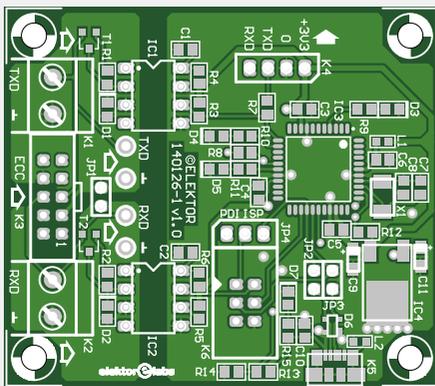


Figure 2. Disposition des composants sur le circuit imprimé.

Résistances : (CMS 0805 à 5 %, 125 mW sauf indication contraire)

R1, R2 = 68 Ω, 1 %
R3, R5 = 270 Ω
R4, R6 = 100 Ω
R7, R8 = 220 Ω
R9 à R11 = 330 Ω
R12 = 10 kΩ
R13 = 33 kΩ
R14 = 56 kΩ

R15 = 680 Ω

Condensateurs :

(10 %, 50 V, CMS 0805 sauf indication contraire)
C1 à C6, C10 = 100 nF, X7R
C7, C8 = 22 pF, C0G/NP0 5 %
C9 = 4,7 μF, 20 V, tantale, CMS A
C11 = 10 μF, 10 V, tantale, CMS A

Inductances, filtre :

L1, L2 = 330 Ω @ 100 MHz, 0,08 Ω, 1,7 A, CMS 0603

Semi-conducteurs :

D1, D2 = TS4148 RY
D3 à D5 = LED, jaune, CMS 0805
D6 = PRTR5V0U2X
D7 = LED verte, CMS 0805
T1, T2 = BF545A, CMS SOT23
IC1, IC2 = 6N137, DIP8
IC3 = ATmega128A4U-AU
IC4 = KF33BDT-TR

Divers :

K1, K2 = bornes à vis à 2 pôles, encartable, au pas de 5,28 mm
K3 = embase à 2x5 picots, pas de 2,54 mm
K4 = embase à 1x4 picots, pas de 2,54 mm
K5 = prise micro USB type B, CMS

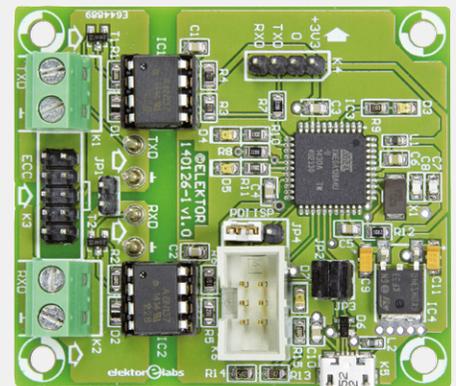


Figure 3. La carte de l'analyseur sériel UART/RS232 terminée (prototype du labo).

K6 = embase à col de 2x3 picots, pas de 2,54 mm
JP1 à JP3 = embase à 2x1 picot, pas de 2,54 mm
JP4 = embase à 1x3 picots, pas de 2,54 mm
JP1 à JP4 = cavalier à 2 contacts, pas de 2,54 mm
PC1 à PC4 = broche encartable diam. 1,3 mm
X1 = quartz 12 MHz, 18 pF, CMS 5x3,2 mm (Abrakon ABM3-12.000MHZ-D2Y-T) circuit imprimé réf. 140126-1 v1.0 [3] microcontrôleur programmé réf. 140126-41 [3] module prêt à l'emploi réf. 140126-91 [3]

l'écran la liste des débits binaires possibles pour la liaison avec l'analyseur. Choisir un débit par les caractères **O** à **F**. Avec **I** suivi de **1** ou **2**, on précise que le niveau du canal (1 ou 2) doit ou non subir une inversion. Pour une liaison RS232, les canaux Ch1 et Ch2 doivent rester *not inverted*, pour TTL-UART, mettre sur *inverted*. Si la liaison se déroule avec les habituels 8 bits de données (8N1), on devrait obtenir au moins les caractères enregistrés dans le programme de terminal.

Les autres commandes concernent principalement le format de sortie, vous en trouverez un sommaire dans l'encadré Commandes. ◀

(140126 - version française : Robert Grignard)

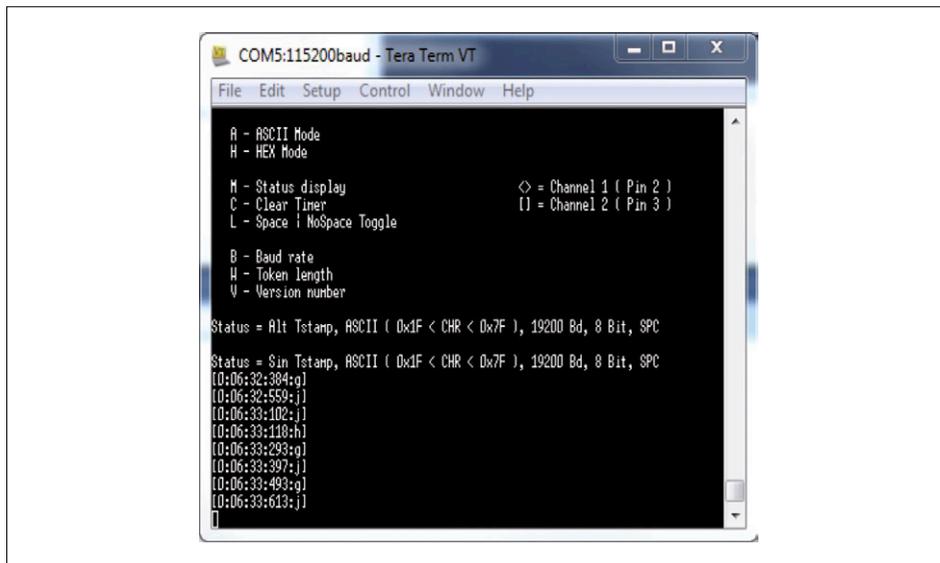
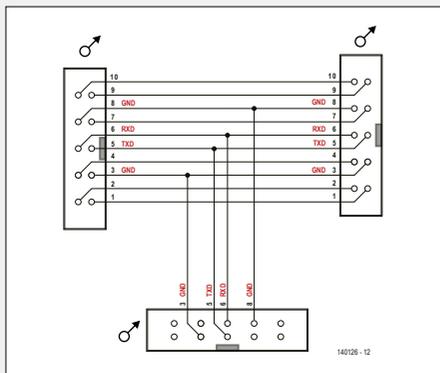


Figure 4. Sortie des données sur le canal 2 dans Tera Term.

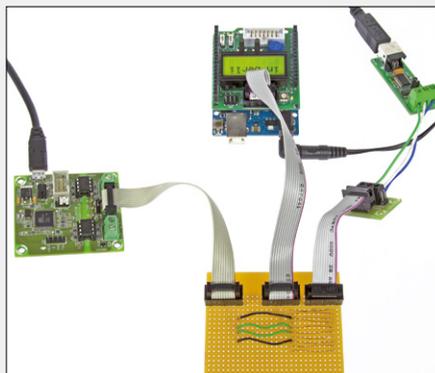
Liens

- [1] www.atmel.com/devices/ATXMEGA128A4U.aspx
- [2] www.elektor-magazine.fr/130155
- [3] www.elektor-magazine.fr/140126
- [4] www.atmel.com/tools/avrsoftwareframework.aspx?tab=overview
- [5] www.elektor-labs.com/contribution/from-the-lab-new-software.14381.html
- [6] www.elektor-magazine.fr/140009
- [7] www.elektor-magazine.fr/130023

Enregistrer des données avec ECC



Elektor a spécifié l'ECC, *Embedded Communication Connector* qui délivre des signaux TTL-UART sur deux broches d'une embase à picots de 2x5 contacts. Le *shield* d'extension pour Arduino Uno [6] est équipé d'origine de ce modèle de connecteur ; un adaptateur pour la carte SAM D20 du cours ARM suivra. Au moyen d'un câble plat, on peut brancher sur ce connecteur mâle



différents modules de communication dont on peut déjà citer un convertisseur RS485 [2], un module radio 433 MHz [7] et un convertisseur UART/USB (dans cette édition). Un accès NFC et d'autres modules suivront.

Si l'on veut partager l'enregistrement, pour que l'Arduino Uno, par exemple, puisse l'afficher, on peut brancher

directement le *shield* d'extension par câble plat sur l'analyseur sériel. Dans le cas où l'on désirerait enregistrer le flux bidirectionnel, on aura recours à un répartiteur de signaux en Y, doté alors de trois connecteurs mâles ECC. Nous avons bien entendu construit au labo une telle carte.

Pour les premiers essais, les collègues ont utilisé une plaque perforée pour réaliser un prototype. La photo montre le partage des données à enregistrer, échangées entre l'Arduino Uno et un PC (voyez à ce sujet la page d'initiation « Découvrir » dans cette édition) Le répartiteur en Y ainsi que l'analyseur y sont insérés entre le *shield* et le convertisseur RS485.

La masse ECC n'est prolongée que sur un canal de l'analyseur quand le cavalier JP1 n'est pas installé (cf. fig. 1). Il faut donc le mettre en place.



bienvenue dans votre e-choppe

Elektor recommande

La rédaction craque pour Raspberry Pi 2

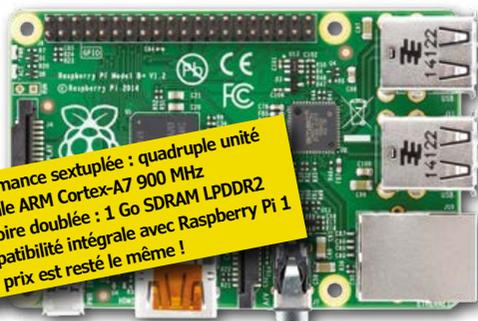
On savait qu'il arriverait, l'ordinateur à tout faire, avec accès à l'internet, graphisme HD et qui tient dans une main. Mais pour un prix qui, en pièces d'1 €, tient lui aussi dans la main, un truc à la framboise, conçu comme matériel d'apprentissage, l'eusses-tu cru ? Non, depuis 3 ans, nous allons de *surprise Pi* en *surprise Pi*.

Des certificats de naissance de produits magnifiques, j'en reçois des brouettes. Celui du **Raspberry Pi 2**, début février, a été une décharge d'adrénaline. Je vais réaliser un vieux rêve : créer pour mon cher synthétiseur analogique des années 80, avec ses dizaines de modules toujours en état de marche, l'interface



musicien-instrument qui combine la supériorité de la synthèse analogique à une commande numérique moderne, avec interface graphique et mémorisation instantanée des configurations sonores même les plus complexes.

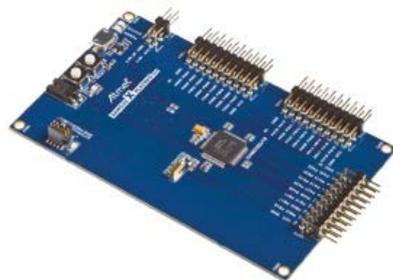
Denis Meyer,
Elektor France



- Performance sextuplée : quadruple unité centrale ARM Cortex-A7 900 MHz
- Mémoire doublée : 1 Go SDRAM LPDDR2
- Compatibilité intégrale avec Raspberry Pi 1
- Et le prix est resté le même !

Vos favoris :

1. Atmel SAM D20 Xplained Pro
www.elektor.fr/top1



2. Raspberry Pi Mod. B+
www.elektor.fr/top2

3. Raspberry Pi 2 Mod. B
www.elektor.fr/top3

4. DVD 2014
www.elektor.fr/top4

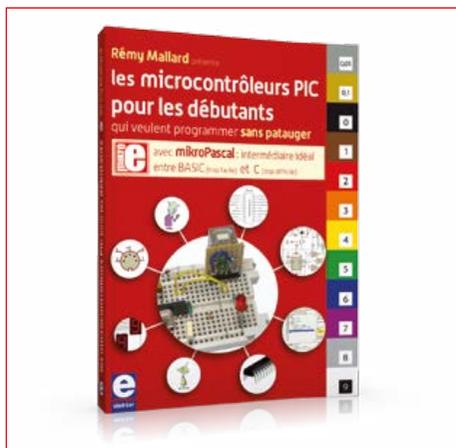
5. Arduino Uno
www.elektor.fr/top5

6. Maîtrisez les microcontrôleurs à l'aide d'Arduino
www.elektor.fr/top6

7. BL600 e-BoB
www.elektor.fr/top7

8. L'électronique pour les débutants
www.elektor.fr/top8

Les microcontrôleurs PIC pour les débutants

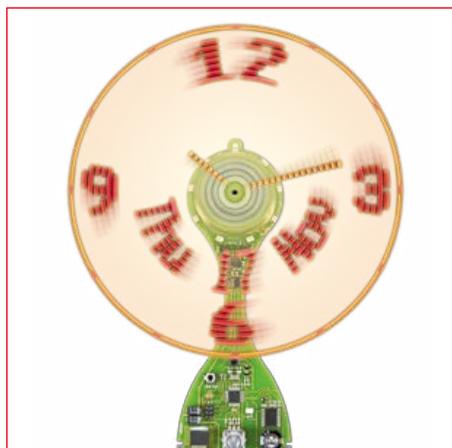


Rémy Mallard initie les débutants à la programmation des PIC au moyen d'exemples pratiques. Il commence par les principes essentiels de programmation, puis regroupe par chapitre les informations nécessaires à la réalisation de chaque exemple. Il fait la part belle aux « petits » contrôleurs à 8 bits, qui disposent entre autres de comparateurs, d'un oscillateur interne, de convertisseurs A/N, de communication à deux fils ou série, et bien davantage.

Prix (membres) : 34,16 €

www.elektor.fr/debutpic

Clic-Clock : Cadran lumineux interactif circulaire



L'électronique n'est jamais aussi belle que lorsqu'elle marie habilement la magie à la physique, la mécanique au logiciel, l'imagination à la rigueur et à la précision, le goût du beau à la passion du travail bien fait. Ici, le maître mot est la persistance rétinienne; cette horloge affiche heure, date et température. Le transformateur doit encore être construit avant l'utilisation ; une description de la construction est disponible.

Prix (membres) : 125,96 €

www.elektor.fr/clic-clock

USBProg 5.0 programmeur open source + interface web



Le programmeur USBprog est apprécié des lecteurs d'Elektor qui s'en servent comme outil multifonctionnel pour divers contrôleurs. Voici une nouvelle version, dotée d'un serveur internet. Son interface HTML lui permet de fonctionner avec tous les systèmes d'exploitation et toutes les tablettes. Aucune installation d'un logiciel propre ! L'USBprog 5.0 est livré monté et testé avec boîtier, alimentation et câble.

Prix (membres) : 38,25 €

www.elektor.fr/usbprog5



Module Intel Edison : concentré de puissance



Edison est le premier ordinateur sur carte d'une nouvelle série à faible coût proposée par Intel. Sa vocation est de faciliter l'accès à l'Internet des Objets (IOT) aussi bien aux très grands fabricants d'appareils électroniques qu'à de petits bureaux d'études ou à des individus isolés. En dépit de sa petite taille, cette plate-forme de développement offre une impressionnante palette de fonctions, d'excellentes performances, une durabilité et un large éventail d'E/S et le support logiciel.

Module Intel Edison

concentré de puissance : à vous d'en faire quelque chose d'exceptionnel

Egg-bot DeLuxe

même une poule sous LSD ne pondra jamais des oeufs aussi bien décorés

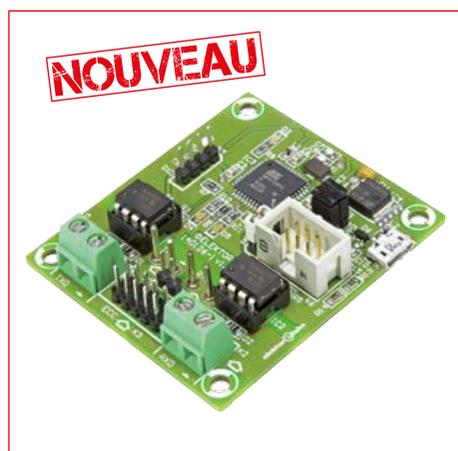
Arduino Baba et les 37 capteurs

37 capteurs variés, d'usage universel, compatibles avec l'univers Arduino



Prix (membres) : 23,95 €
www.elektor.fr/intel-edison

analyseur UART/RS-232

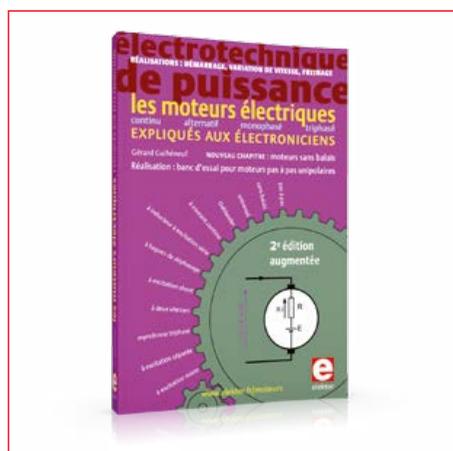


Si, en cours de développement ou de test, quelque chose cloche dans la communication entre deux appareils, vous pouvez vous attendre à un dépannage épineux. Sauf si vous avez notre analyseur sériel UART qui vous simplifie la vie en enregistrant les octets dans les deux sens à la milliseconde près. Des raccordements variés et une séparation galvanique lui permettent de s'insinuer sans risque dans toute liaison sérielle TTL ou RS-232.

Prix (membres) : 31,46 €

www.elektor.fr/analyseur UART-RS232

Les moteurs électroniques expliqués aux électroniciens

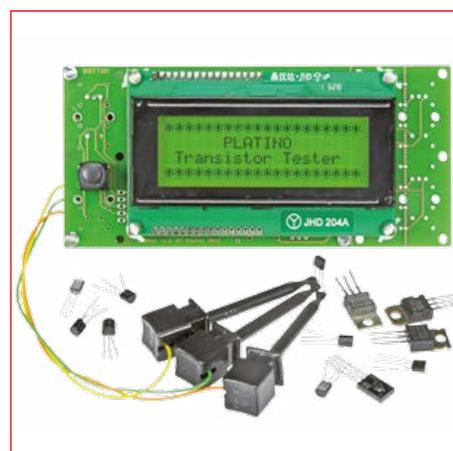


Les électroniciens découvriront ici les moteurs électriques, tandis que les électrotechniciens admettront qu'électronique n'est pas synonyme de complexité. L'information est digeste : constitution, fonctionnement, caractéristiques, domaines d'utilisation, réalisations simples et concrètes. Le lecteur mesurera ses connaissances grâce à des questionnaires d'évaluation (avec corrigés).

Prix (membres) : 39,50 €

www.elektor.com/moteurs

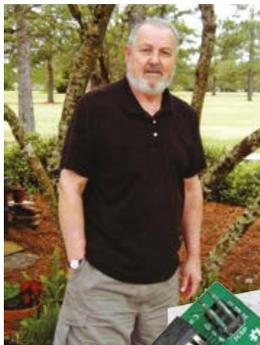
Testeur de transistors avec Platino



Ce testeur permet de trier comme bons ou mauvais les transistors sans référence ou douteux, de déterminer leur type (bipolaires PNP et NPN, MOSFET à canal N ou P) et de mesurer leur gain. L'instrument de mesure repose sur la carte Platino d'Elektor. Sont fournis : circuit imprimé, microcontrôleur programmé, circuit imprimé polyvalent pour AVR-Platino, afficheur et boîtier.

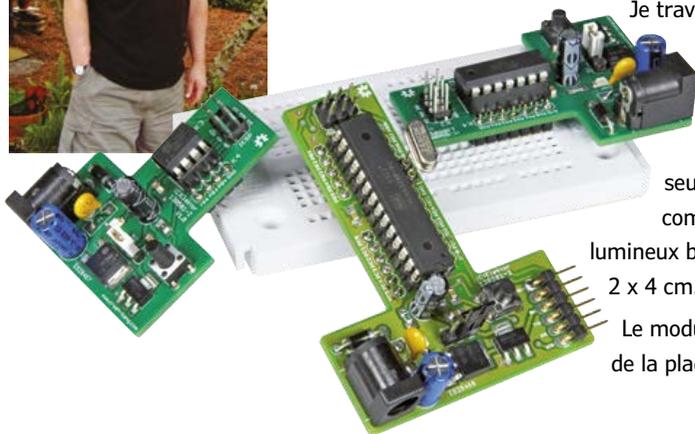
Prix (membres) : 66,95 €

www.elektor.fr/platino-transistortester



par Gene Sellier

J'ai acheté ces modules pour passer d'une carte Arduino complète vers une application pour laquelle un simple ATtiny suffirait. C'est donc surtout le T-Board 8 qui m'a fait de l'usage, car mon but était d'employer un ATtiny à 8 broches et de réduire ainsi le nombre de composants mis en œuvre. J'ai trouvé que pour les essais sur plaque d'expérimentation, ces modules en T sont parfaits.



Je travaille avec l'IDE WinAVR et un programmeur USBtinyISP d'Adafruit. Pour une commande de ventilateur sous 12 V, il me suffisait d'un ATtiny 45, d'une thermistance, d'un résistance et d'un MOSFET. Le microcontrôleur mesure la température dans un coffret et commute le ventilateur pour la maintenir en-dessous d'un certain seuil. Deux broches et quelques lignes de code suffisent. Du coup, comme il restait des broches inutilisées, j'ai rajouté un indicateur lumineux bicolore en façade. Tout ça tient sur un morceau de carte perforée de 2 x 4 cm.

Le module en T facilite considérablement l'étude au stade de la plaque d'expérimentation.

L'original de ce compte-rendu se trouve sur www.elektor.fr/t-boards

Votre témoignage sur un article vendu dans l'e-shoppe d'Elektor peut vous rapporter 100 € sous la forme d'un bon d'achat de produits à valoir sur vos achats dans notre boutique en ligne.

Informations complémentaires et conditions sur www.elektor.fr/rev



ARDUINO DAY 2015
WEEK! MARCH 24th - 30th

GREAT ARDUINO DEALS @ ELEKTOR.COM



Module Intel Edison

concentré de puissance : à vous d'en faire quelque chose d'exceptionnel

Egg-bot DeLuxe

même une poule sous LSD ne pondra jamais des oeufs aussi bien décorés

Arduino Baba et les 37 capteurs

37 capteurs variés, d'usage universel, compatibles avec l'univers Arduino

Egg-Bot peut imprimer sur toutes sortes d'objets sphériques dont le diamètre est approximativement compris entre 3 et 10 cm. Vous pouvez l'utiliser pour créer des œufs que même une poule sous LSD ne pondra jamais, personnaliser des décorations de Noël, des balles de ping-pong, ou encore des ampoules. Dingue et cool, mais Egg-Bot n'est pas qu'un gadget. Son électronique et son programme ont été conçus pour pouvoir être adaptés à un nouvel usage. Egg-Bot est donc aussi un moyen ludique et pratique pour s'initier à la robotique faite maison et à la commande numérique par ordinateur (CNC).

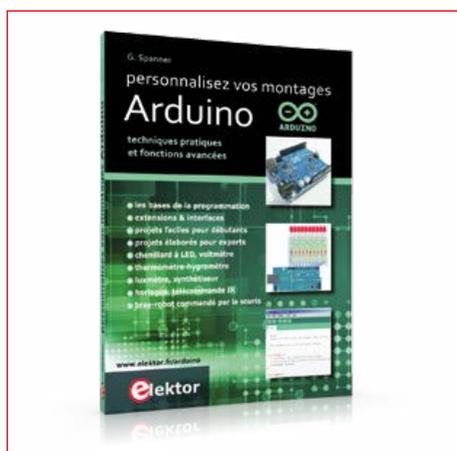


Prix (membres) : 179,95 €
www.elektor.fr/eggbot

Personnalisez vos montages Arduino

offre mystère

Carte Atmel SAM D20 Xplained pro



L'objectif de ce livre est de vous emmener à pas guidés vers la maîtrise d'Arduino. Les projets sont regroupés par thème, avec des bases théoriques. Vous apprendrez à exploiter des techniques essentielles (conversion analogique-numérique, modulation de largeur d'impulsion, pilotage de différents types d'afficheurs, interface I2C, interruptions).

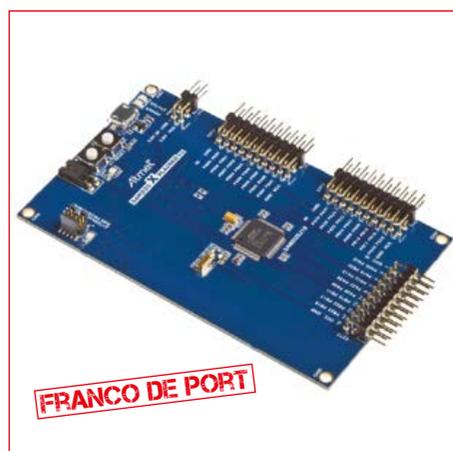
Prix (membres) : 31,05 €



oserez-vous ?



www.elektor.fr/mystery



La carte Atmel SAM D20 Xplained Pro est conçue pour le prototypage avec les microcontrôleurs Cortex-SAM D20-M0+. Elle bénéficie d'un riche environnement d'outils pratiques et gratuits. Elle est utilisée notamment dans notre nouveau cours d'initiation aux µC à 32 bits dont le premier épisode a été publié dans le n° de janvier-février 2015, p. 112, et se poursuivra dans chaque numéro tout au long de l'année. Ce module est livré monté et testé. Lancez-vous !

Prix (membres) : 26,96 €

www.elektor.fr/samd20-board



- ✓
analyseur UART/RS232:
 - 140126-1 circuit imprimé
 - 140126-41 ATXmega128A4U-AU
 - 140126-91 module assemblé

- ✓
robot aérateur de cave
 - 100892-1 Platine Platino
 - 100892-41 microcontrôleur programmé pour Platino
 - 140154-91 e-BoB CC2

- ✓
e-BoB
 - module assemblé 140270-91

- ✓
convertisseur USB ↔ série
 - 130542-1 circuit imprimé Platino
 - 130542-91 assemblé

à commander en ligne sur
www.elektor.fr/biznis

elektor•post : votre lettre d'information électronique !



Chaque semaine, plus de 100.000 lecteurs trouvent elektor.post dans leur boîte de réception le vendredi matin, avec des informations passionnantes sur le monde de l'électronique.

Vous ne recevez pas elektor.post ? Vous ne savez pas ce que vous manquez ! :

- toutes les deux semaines un projet électronique Elektor inédit et gratuit sous forme de PDF (d'une valeur de 2,50 €) est joint à cette lettre
- des offres de réduction spéciales dans l'e-choppe d'Elektor
- 5 € de remise sur votre prochain achat dans l'e-choppe d'Elektor

Inscrivez-vous, c'est gratuit !
www.elektor.fr/elektorpost

DVD Elektor 2014



Ce DVD-ROM réunit tous les articles d'ELEKTOR parus au cours de l'année 2014, en français (mais aussi en anglais, allemand et néerlandais). Il contient le texte des articles ainsi que les schémas, et tous les dessins des circuits imprimés, sous forme de fichiers à haute résolution. Une fonction de recherche dans la table des matières vous permet de trouver immédiatement l'article souhaité.

Prix (membres) : 24,75 €

www.elektor.fr/dvd-2014

Maîtrisez les microcontrôleurs à l'aide d'Arduino

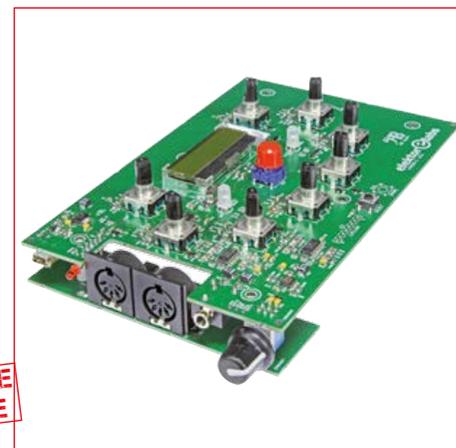


Des montages Arduino inédits, étudiés spécialement pour se débarrasser définitivement de ses amis et de sa famille, et pour se retrouver enfin seul et libre de passer tout son temps à apprendre la programmation des microcontrôleurs ! Le seul livre sérieux de micro-électronique et de micro-informatique dans lequel il soit question d'Arduino mais aussi de Blanche-Neige, de la Cucaracha (en stéréo), de Saint Augustin, de Scarlatti et de Pindare (pas Pandore).

Prix (membres) : 36,85 €

www.elektor.fr/arduino

Synthé J²B : plateforme ouverte pour la synthèse musicale numérique



Inspiré par l'Atmegatron, ce nouveau synthé numérique d'Elektor invite (et incite) à l'expérimentation. Modifiez à votre guise le code source ouvert (en C). Huit codeurs rotatifs à bouton poussoir commandent, entre autres, la forme d'onde, le type de filtre, l'enveloppe ADSR de filtrage et de modulation d'amplitude, la fréquence du LFO, la vitesse des arpèges, la distorsion, le déphasage, la modulation de largeur d'impulsion, le portamento et l'excellent *broyeur de bits*.

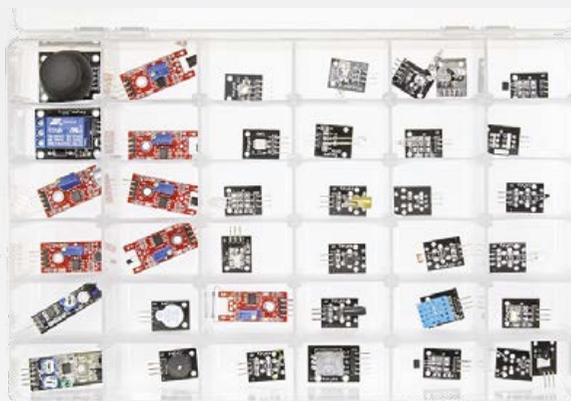
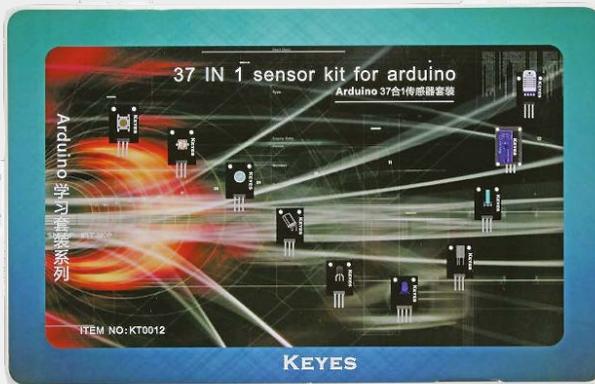
Prix (membres) : 112,45 €

www.elektor.fr/j2b-synthesizer



« Vous avez aimé Ali Baba et les 40 voleurs, vous adorerez Arduino Baba et les 37 capteurs ! »

Une riche collection de capteurs variés et universel, utilisables directement avec les cartes Arduino (pas de soudage requis, compatibilité broche à broche directe). Avec cet ensemble bien conçu, vous mesurez tous que vous voulez à l'aide d'Arduino. Parmi les 37 composants, livrés dans une boîte plastique pratique, en plus des capteurs habituels vous trouverez notamment un module à LED clignotant automatique, plusieurs modules à LED bicolores ou tricolores, un détecteur de battement cardiaque, un détecteur de chocs, un interrupteur à clé, un capteur laser, un manche de commande X/Y et bien davantage.



Module Intel Edison

concentré de puissance : à vous d'en faire quelque chose d'exceptionnel

Egg-bot DeLuxe

même une poule sous LSD ne pondra jamais des oeufs aussi bien décorés

Arduino Baba et les 37 capteurs

37 capteurs variés, d'usage universel, compatibles avec l'univers Arduino



Prix (membres) : 49,46 €

www.elektor.fr/arduino-sensor-kit

You've been generous

THANK YOU.

NOW IT'S OUR TURN:

Member appreciation Month

All kinds of extra's

MEMBERS ONLY!



robot aérateur de cave

Platino régule l'humidité

Luc Lemmens (Elektor-lab) d'après une idée de Danny Winkler

Il n'est pas rare qu'une cave soit humide. Une température basse associée à l'absence de chauffage en est souvent la cause. Les relents de moisissure, la corrosion et la pourriture du bois et des objets entreposés s'ensuivent et c'est loin d'être enviable. Une bonne ventilation peut enrayer cette contrariété ou au moins la soulager, mais qu'est-ce qu'une bonne ventilation ? Ouvrir une porte ou une fenêtre ne suffit pas. Le système proposé ici prend les choses par le bon bout, il constitue un moyen efficace pour espérer transformer une cave en espace de vie.

Traditionnellement, les caves comme les greniers restent dans une maison des locaux peu propices à un séjour agréable. Dès l'origine, ils étaient destinés à être des espaces de rangement où l'on ne se rend que quand il est nécessaire d'entreposer des objets ou de les récupérer, sans s'y attarder.

Normalement, on ne les chauffe pas et ne leur octroie un grand nettoyage et un rangement qu'une fois par an tout au plus.

Ce n'est pas exactement un endroit pour y passer du temps. La cave à vin constitue sans doute l'exception qui confirme la règle, mais ceci est une autre histoire. Le grenier profite quelque peu du chauffage des étages inférieurs, mais la cave est froide par définition, destinée au garde-manger où il était possible d'augmenter la durée de conservation des aliments. De nos jours, réfrigérateurs et congélateurs ont pris la relève avec succès. Quant à l'humidité, nous n'en vou-

lons certainement plus. Mais l'atmosphère froide de la cave favorise la condensation : l'air humide et chaud qui y entre se refroidit et l'eau qu'il contient se dépose sur les parois froides, murs et sols. C'est pourquoi vous constatez parfois, au moins en été, des gouttes d'eau suspendues aux canalisations dans la cave, et elles ne proviennent pas d'une fuite dans un tuyau ou un raccord.

Il va de soi que l'équilibre hygrométrique des locaux souterrains d'une habitation

est d'une grande complexité parce que de nombreux facteurs peuvent intervenir, comme des inondations périodiques, des revêtements inadéquats ou des infiltrations d'eau souterraine par manque d'étanchéité des murs ou du sol. Ce sont des aspects de la question à soumettre à un architecte ou à des spécialistes. Nous partons ici du principe que les caves sont saines et que l'humidité provient essentiellement de la condensation.

C'est généralement le cas dans les anciennes caves, puisque les prescriptions pour les constructions récentes devraient empêcher le phénomène de se produire. En effet, les moisissures causées par l'humidité peuvent nuire gravement à la santé. Tout cela sans parler de l'oxydation des métaux, de la pourriture du bois et d'autres objets que nous souhaitons préserver. De toute façon, on n'apprécie pas l'odeur de moisi dans les caves. Les ventiler peut résoudre beaucoup de choses, mais comment s'y prendre de manière efficace ?

L'électronique à l'aide

Sur notre site elektor-labs.com [1], un lecteur allemand a présenté un système à microcontrôleur pour rendre acceptable le degré d'humidité dans sa cave et le maintenir à la valeur voulue. Deux capteurs, l'un à l'intérieur, l'autre à l'extérieur, mesurent l'hygrométrie et la température. Selon les valeurs enregistrées, le système actionne une fenêtre et fait tourner ou arrête automatiquement un ventilateur. Il a ainsi réussi à résoudre le problème d'humidité.

Le projet nous a paru suffisamment digne d'intérêt pour le reprendre, le traiter et le publier dans notre magazine. Il ne comportait pas de projet de circuit imprimé et au lieu de recommencer le développement d'une carte à microcontrôleur avec LCD, boutons de commande et alimentation, nous avons décidé d'utiliser à la place notre carte Platino comme base de ce projet. Le logiciel est encore une fois écrit dans l'environnement de développement Arduino, si bien que la plupart de nos lecteurs pourront aisément, si c'est nécessaire ou souhaité, adapter le processus de réglage et les autres fonctions du système. L'électronique supplémentaire pour la commande du ventilateur et la manœuvre de la fenêtre ainsi que le raccordement correct des capteurs suivront dans une prochaine publication.

Spécifications

- Deux capteurs ChipCap2 pour mesure de l'humidité relative (précision 2 %) et température ($\pm 0,6$ °C)
- LCD à 4x20 caractères
- Tension d'alimentation : 8 à 12 V_{CC}
- Logiciel pour microcontrôleur ATmega328 (Arduino)
- Commande par codeur rotatif à bouton-poussoir
- Manœuvre électrique d'ouverture de fenêtre
- Commande de ventilateur ou déshumidificateur

Le projet d'origine mettait en œuvre deux capteurs d'humidité et de température relativement chers, nous les avons remplacés par des capteurs ChipCap2 meilleur marché, ceux que nous avons présentés dans le numéro double de janvier-février, installés sur un eBoB, un médiateur pour en faciliter le montage, vu que ces capteurs en CMS sont peu maniables (cf. **fig. 1**).

Le système de mesure...

Difficile de faire plus simple : un capteur mesure la température et l'humidité dans la cave, l'autre se charge de la même chose dehors. La mesure de l'humidité relative ne nous dit pas grand-chose, parce que la température joue un rôle essentiel. En effet, l'air chaud peut contenir plus de vapeur d'eau que l'air froid. Aussi, ventiler pendant une chaude journée d'été aura l'effet contraire, l'air chaud et humide s'engouffre dans la cave et l'humidité se condense dans cet espace froid.

C'est avec la température que nous pourrons, à partir de l'humidité relative mesurée, déterminer l'humidité absolue et donc la concentration d'humidité exprimée en grammes d'eau par mètre cube d'air, précisément ce dont nous avons besoin pour décider de l'attitude à prendre (cf. encadré). Comme dans d'autres flux, l'humidité se déplace des hautes concentrations vers les basses. Connaissant l'humidité absolue dehors et dedans, nous saurons quand il est opportun d'aérer la cave.

... et le principe de réglage

La méthode elle non plus n'est pas compliquée. Dès que la concentration en humidité est plus élevée dedans que dehors, il faut aérer, ou pas. Pourtant, ce n'est pas aussi simple que ça. Supposons qu'il fasse un froid glacial dehors, il ne serait pas conseillé d'ouvrir et de laisser la

fenêtre ouverte des heures durant. Indépendamment de la perte d'énergie dans la maison, le risque existe que des canalisations d'eau gèlent dans la cave. Sans parler d'une panne de capteur, auquel cas nous ne pouvons pas permettre au système d'agir sans contrôle. Et même si ce n'est pas pour tout de suite, il ne faut pas non plus que l'humidité relative tombe à un niveau trop bas dans la cave. Aussi, définissons les critères d'ouverture et de fermeture de la fenêtre de la cave, par conséquent de mise en marche et d'arrêt de l'aération.

- L'un des capteurs est en panne : fermer la fenêtre et stopper le ventilateur
- Température intérieure < zéro : fermer la fenêtre et stopper le ventilateur
- Humidité relative dedans trop basse : ouvrir la fenêtre et stopper le ventilateur
- Humidité absolue dedans > dehors : ouvrir la fenêtre, faire tourner le ventilateur
- Humidité absolue dedans < dehors : fermer la fenêtre et stopper le ventilateur

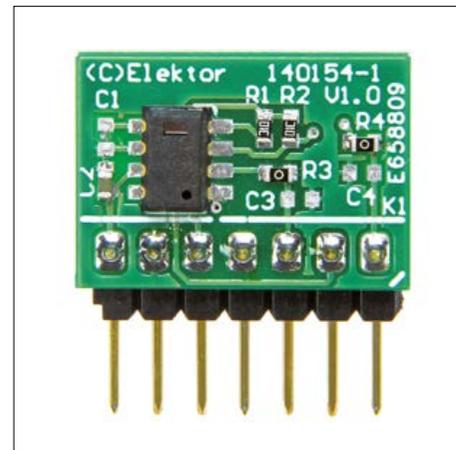


Figure 1. Le petit médiateur imprimé facilite grandement l'installation des capteurs ChipCap2.

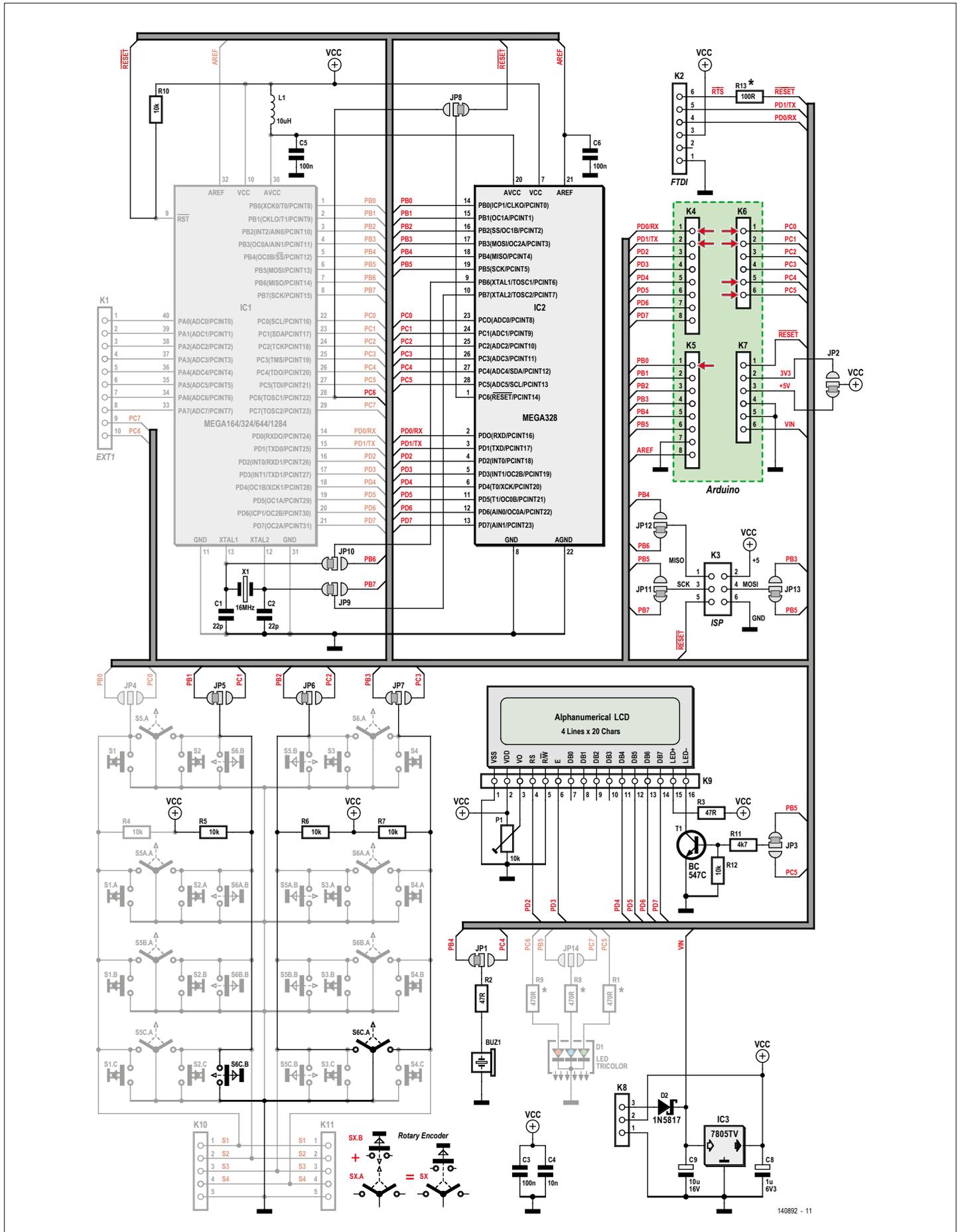


Figure2. Le schéma de Platino. Les composants et cavaliers nécessaires à ce projet sont mis en évidence.

L'ordre des divers critères indique aussi leur priorité, la 1^{re} étant la plus haute. Les états de la fenêtre et du ventilateur sont évalués toutes les dix minutes, donc le système n'a pas besoin d'hystérésis, on ne risque pas de commutations intempestives accompagnées de bruit et de dépense d'énergie.

Le matériel nécessaire

Notre platine polyvalente Platino (100892) d'octobre 2011 constitue le cœur du dispositif. On y trouve un ATmega328 avec chargeur d'amorçage Arduino, on l'utilise normalement comme une carte Arduino Uno sous environnement de développement Arduino. La **fig. 2** reproduit le schéma complet de cette carte, les composants nécessaires à ce projet-ci y figurent en gras. Vous y verrez aussi les ponts de soudure qui doivent être ouverts ou fermés. La liste des composants de cet article indique également les composants indispensables à ce projet.

Ne pas oublier les capteurs de température et d'humidité, bien entendu : il s'agit de deux unités de notre CC2-eBoB (réf. 140154-91), carte assemblée prête à l'emploi. Les capteurs ChipCap2 communiquent par bus I²C et disposent d'origine de l'adresse 0x28. L'un des deux devra donc changer d'adresse pour qu'ils puissent vivre ensemble sur le même bus. Il y faudra un bout de logiciel, mais nous y penserons quand Platino sera construit.

Construction

Joindre au schéma la liste des composants devrait vous faciliter la vie lors de l'assemblage de la carte Platino. Mieux vaut commencer par les ponts de soudure et cavaliers JP1 à JP14. Toujours par souci de clarté, le **tableau 1** énumère les positions requises. Vous n'aurez besoin des cavaliers JP11, 12 et 13 que si vous comptez utiliser l'adaptateur ISP AVR pour programmer l'ATmega328. Pour exploiter comme nous le chargeur d'amorçage Arduino, il faut brancher sur K2 de Platino un convertisseur sériel USB vers TTL. Un câble FTDI de 5 V convient parfaitement et permet lors des essais ou d'un développement ultérieur du micrologiciel de fournir l'alimentation de l'ensemble du circuit imprimé de commande. La bobine L1 sert uniquement à découpler l'alimentation analogique de l'ATmega328. Comme notre application ne fait pas usage de la conversion A/N, vous

Tableau 1. Ponts de soudure / cavaliers sur Platino

cavaliers :		cavaliers :	
JP1	PB4 (vibreux)	JP8	remise à zéro
JP2	5V	JP9	XTAL2
JP3	(éclairage arrière)	JP10	XTAL1
JP4	-	JP11	PB7
JP5	PB1	JP12	PB6
JP6	PB2	JP13	PB5
JP7	PB3	JP14	-

pouvez franchement remplacer L1 par un pont de câblage.

Parlons encore des deux cartes de capteurs, l'une d'elles doit donc changer d'adresse I²C, nous allons le faire avec un sketch écrit spécialement pour l'occasion. Commençons par raccorder l'un des CC2-eBoB à Platino. Les liaisons I²C aboutissent sur K6, broche 5 pour SDA, broche 6 pour SCL. *Connectez momentanément la broche VDD sur la sortie numérique PB0 (c'est la broche 1 de K5) de Platino !* La modification d'adresse I²C n'est réellement possible que pendant une courte période après l'enclenchement de l'alimentation du capteur ChipCap2, aussi notre sketch peut-il lui-même appliquer et retirer la tension. La broche de masse

GND est reliée à la broche 4 de K7. Lancez l'environnement de développement Arduino sur votre ordinateur et chargez le sketch *CC2A_set_I2C_address.ino*, il est disponible gratuitement sur [2]. Deux constantes sont définies dans le sketch :

1. CURRENT_I2C_ADDRESS = 0x28
(valeur standard, dans notre application, le capteur intérieur)
2. NEW_I2C_ADDRESS = 0x22
(le capteur extérieur)

Il est aussi possible d'utiliser d'autres adresses dans le domaine de 0x00 à 0x7f, mais il faut alors y adapter le code source du système d'aération, où elles s'appellent *Inside_sensor* et *Outside_sensor*. Si ce n'est déjà fait, branchez la carte Pla-

Humidité relative et absolue

Ce système utilise l'humidité absolue comme variable de réglage. Mais la plupart des capteurs donnent la valeur de l'humidité relative et la température, ils ne sont pas capables de délivrer directement cette grandeur. Bien que la détermination de l'humidité absolue fasse intervenir d'autres grandeurs physiques, on obtient une bonne approximation par les formules :

$$\text{Pression de vapeur} = F/100 \times 6,1078 \times 10^{((a \times T)/(b+T))}$$

$$\text{Humidité absolue [g/m}^3] = 216,7 \times \text{pression de vapeur}/(273,15+T)$$

Dans lesquelles :

a = 7,5 (constante)

b = 237,3 (constante)

T = température mesurée en degrés Celsius

F = humidité relative

Source : <http://www.wettermail.de/wetter/feuchte.html#f1>

Ces formules, le logiciel les utilise et leur calcul s'effectue dans la fonction *Calc_Abs_Hum* du sketch Arduino.

Il existe encore d'autres méthodes d'approximation, mais en pratique, il y a peu de divergences entre elles. Dites-vous bien que dans ce projet, même l'absolu peut être relatif : il n'est pas important de déterminer des valeurs exactes, tout ce qui compte, c'est de savoir à coup sûr si la plus haute concentration en humidité est dehors ou dedans.

Tableau 2. Connexions et dénomination des broches sur la carte de commande

Arduino	ATmega328	fonction
ouverture de fenêtre :		
Digital 0	PD0	HAUT pendant ouverture
Digital 1	PD1	HAUT pendant fermeture
Analog 0	PC0	fin de course BAS à l'extrémité (fenêtre ouverte)
Analog1	PC1	fin de course BAS à l'extrémité (fenêtre fermée)
LCD :		
Digital 2	PD2	RS
Digital 3	PD3	Enable
Digital 4	PD4	Data D4
Digital 5	PD5	Data D5
Digital 6	PD6	Data D6
Digital 7	PD7	Data D7
Analog 5	PB5	marche/arrêt éclairage arrière
codeur rotatif :		
Digital 9	PB1	bouton-poussoir
Digital 10	PB2	phase A
Digital 11	PB3	phase B
ventilateur :		
Digital 8	PB0	marche/arrêt
vibreur :		
Analog 4	PB4	marche/arrêt

tino au PC par un câble FTDI, compilez et chargez *CC2A_set_I2C_address.ino* pour effectuer le changement d'adresse I²C. Dès lors, nous pouvons raccorder les deux cartes de capteurs sur le même bus I²C. Provisoirement, reliez leurs connecteurs d'alimentation aux broches 3 (+5 V) et 4 (GND) de K7 et les lignes I²C SDA sur la broche 5 et SCL sur la broche 6 de K6. La carte d'extension pour ce projet, à paraître ultérieurement, portera un connecteur à 4 voies approprié à ces liaisons.

Pour terminer ce projet, il manque donc le ventilateur et l'ouverture de fenêtre dont la commande et la construction feront l'objet d'un autre article. Pour calmer votre impatience, nous vous livrons déjà la **figure 3** et le **tableau 2** qui exposent les correspondances et les fonctions des différentes broches de Platino. Vous y verrez qu'une entrée de Platino est réservée à la commande d'allumage et d'extinction du ventilateur. L'ouverture de fenêtre réclame deux lignes de commande pour le moteur

(ouverture et fermeture) et deux entrées pour les interrupteurs de fin de course, en option, destinés à s'assurer du bon déroulement de l'opération.

Le micrologiciel

Nous voilà prêts pour le premier test du robot aérateur de cave. Téléchargez le sketch *_140432_Control_Unit.ino* du site Elektor-magazine [3], chargez-le dans le logiciel Arduino et transférez le sketch dans Platino.

Après la mise sous tension, la fenêtre se ferme et le ventilateur s'arrête. Dès que les premières mesures arrivent, Platino adapte leurs états aux circonstances.

Le codeur rotatif ne joue bien sûr pas un rôle de figuration dans le dispositif. Il rend le système plus universel et permet de répondre aux souhaits de chacun, de le modifier en fonction de situations particulières au moyen de différents réglages, lesquels sont enregistrés dans l'EEPROM interne de l'ATmega328 pour être reproduits après une coupure de l'alimentation. Lors du tout premier démarrage, la mémoire n'est pas programmée et ce sont les paramètres standard qui sont automatiquement pris en compte. Ils concernent :

- La température à laquelle la protection contre l'humidité entre en action. D'origine à 3 °C, réglable entre 3 et 10°C.
- La limite inférieure de l'humidité relative dans la cave. Si cette limite est dépassée, on ne fera que ventiler pour rendre la cave moins sèche. D'origine 30%, réglable entre 20% et 80%.

On enregistre ces paramètres pendant le service normal en appuyant sur le codeur rotatif. Le menu des paramètres s'affiche alors, la fenêtre se ferme et le ventilateur s'arrête ainsi que le réglage automatique, de sorte que quand on débranche le système, la fenêtre est fermée. C'est pratique quand on veut laver la fenêtre, par exemple. Dès que les paramètres sont adaptés, le robot aérateur reprend son activité.

Des réglages existent aussi pour l'ouverture électrique de la fenêtre. Il y en a de différents types et certains lecteurs désièreront sûrement réaliser eux-mêmes la partie électromécanique. Nous réservons cette section pour un projet complémentaire. De nouveaux réglages de l'ouver-

Liste des composants Platino

Résistances (5 %/250 mW) :

R2, R3 = 47 Ω
R5, R6, R7, R10, R12 = 10 kΩ
R11 = 4,7 kΩ
R13 = 100 Ω
P1 = 10 kΩ potentiomètre ajust.

Condensateurs :

C1, C2 = 22 pF céramique
C3, C5, C6 = 100 nF
C4 = 10 nF
C8 = 1 µF/50 V radial, au pas de 2,5 mm
C9 = 10 µF/50 V radial, au pas de 2,5 mm

Inductance :

L1 = 10 µH ou pont de câblage

Semi-conducteurs :

D2 = diode Schottky 1N5817
T1 = BC547
IC2 = ATmega328 (programmé. avec chargeur d'amorce Arduino réf. 100892-41)
IC3 = 7805

Divers :

LCD1 = LCD 4 x 20 caractères alphanumériques avec éclairage (p.ex. Midas MC4200B6W-SPR)
X1 = quartz 16 MHz
S6 = codeur rotatif à bouton-poussoir Alps EC12E2424407
BUZ1 = vibreur alternatif 12 mm ABT-410-RC carte Platino réf. 100892-1
2 x CC2-eBoB, réf. 140154-91

ture de fenêtre ne pourront s'effectuer alors qu'en poussant sur le bouton du codeur rotatif au moment de la mise sous tension du robot.

Les réglages une fois exécutés, le système fonctionne automatiquement. Les humidités relatives et températures internes et externes sont mesurées en continu et transcrites sur l'affichage en même temps que les concentrations en humidité calculées.

Dans le cas exceptionnel où l'un des deux capteurs ne fonctionnerait pas, c'est-à-dire qu'il ne serait pas possible de l'interroger par le bus I²C, le robot réagirait immédiatement et indiquerait sur le LCD quel capteur est coupable. En outre, le vibreur acoustique attirerait l'attention sur la situation de panne.

Quand les capteurs se comportent comme il se doit, on adapte si nécessaire toutes les dix minutes la position de la fenêtre et la marche du ventilateur. L'afficheur montre à quel critère le robot satisfait et l'état actuel de la fenêtre et du ventilateur. La **fig. 4** vous présente quelques configurations de l'afficheur LCD.

Conclusion

Nous n'avons traité que du système régulateur dans cet article, un autre suivra pour s'intéresser à la commande du ventilateur et à l'ouverture et la fermeture automatique de la fenêtre. C'est certainement la construction de cet automate qui mérite notre attention. Un nouveau circuit imprimé s'y ajoutera pour relier le tout à Platino.

Même si le robot aérateur semble fonctionner convenablement, nous ne prétendons pas avoir trouvé la panacée qui transforme toute cave humide et envahie de moisissures en un endroit paradisiaque. La description originale sur notre site .LABS [1] montre bien que le concepteur est encore au travail pour perfectionner et étendre son projet. Considérez-le comme une aide pour améliorer le taux d'humidité dans un local construit et entretenu convenablement. En cas de

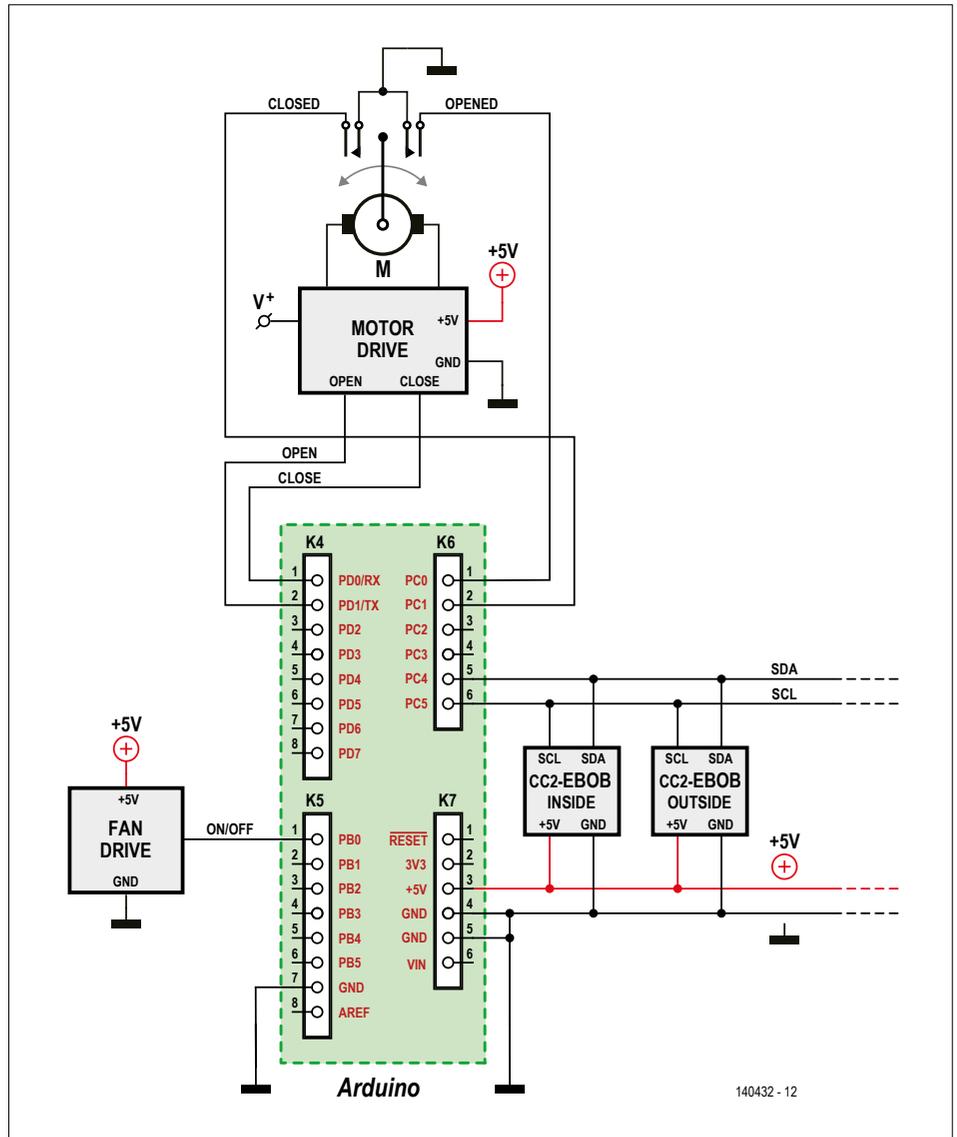


Figure 3. Le câblage complet du robot aérateur.

fuite ou d'humidité intense, faites toujours appel à des spécialistes, non seulement pour votre santé et la conservation

de vos biens, mais aussi pour préserver l'état de votre maison ! ◀

(140432 - version française : Robert Grignard)

Liens

- [1] Projet original : www.elektor-labs.com/project/feuchtegesteuerte-kehlert-ftung-humidity-basement-ventilation-140154.13770.html
- [2] Page du projet CC2-eBoB: www.elektor-magazine.fr/140154
- [3] Page du projet de cet article : www.elektor-magazine.fr/140432



Figure 4. Le robot aérateur vous livre ici en alphanumérique quelques-uns de ses paramètres.

galvanomètre à aiguille commandé par microcontrôleur

un vieux tachymètre reprend du service avec un ATtiny2313



Michael Möge (Allemagne)

Longtemps relégués au rang d'antiquités, les bons vieux galvanomètres à bobine mobile passent aujourd'hui pour de beaux objets, plus agréables à regarder que les afficheurs numériques modernes. Je propose de marier les fonctions actuelles à l'esthétique d'antan.

Mon projet de départ : remplacer l'indicateur de vitesse mécanique – le Sumlog – d'un bateau de plaisance d'un âge certain en restant dans le style d'origine en associant haute technologie et *look* rétro. Le nouveau codeur monté sous la coque ne commande plus l'afficheur au moyen d'un arbre de transmission mécanique, mais y est relié par un câble qui véhicule un signal carré, produit par un ILS (inter-

rupteur à lame souple), de fréquence proportionnelle à la vitesse du bateau (de 10 à 60 Hz).

Il me fallait une électronique capable de mesurer la fréquence du signal carré et de mettre le résultat en forme pour l'afficher sur un instrument à aiguille, avec un débattement de 270° minimum. Le boîtier Sumlog d'origine, aiguille en métal et cadran gradué compris, sera conservé et la distance parcourue sera affichée sans possibilité de réinitialisation.

Du 555 au µC

Mes premiers essais électriques analogiques à base de NE555 et de simple intégrateur RC furent concluants, mais pas pour l'affichage sur 270° par instrument à aiguille classique. Il n'est pas évident de mouvoir l'aiguille métallique avec un tel débattement ? J'en suis donc venu à utiliser un microcontrôleur pour analyser le signal du codeur. J'ai obtenu des résultats intéressants avec un servo de modélisme piloté par le µC, et faisant office d'organe de mesure, mais où trouver des servos qui couvrent 270° ? Un servo classique atteint tout au plus 180°. Pas question d'utiliser un démultipliateur. Ma solution prend la forme d'un moteur pas-à-pas spécial, comme ceux utilisés dans les tableaux de bord automobiles pour les tachymètres. On trouve ces moteurs pas à pas remarquablement stables, pour pas cher, comme pièce de rechange auto sur eBay [1].

Celui que j'ai trouvé ne tourne pas sur

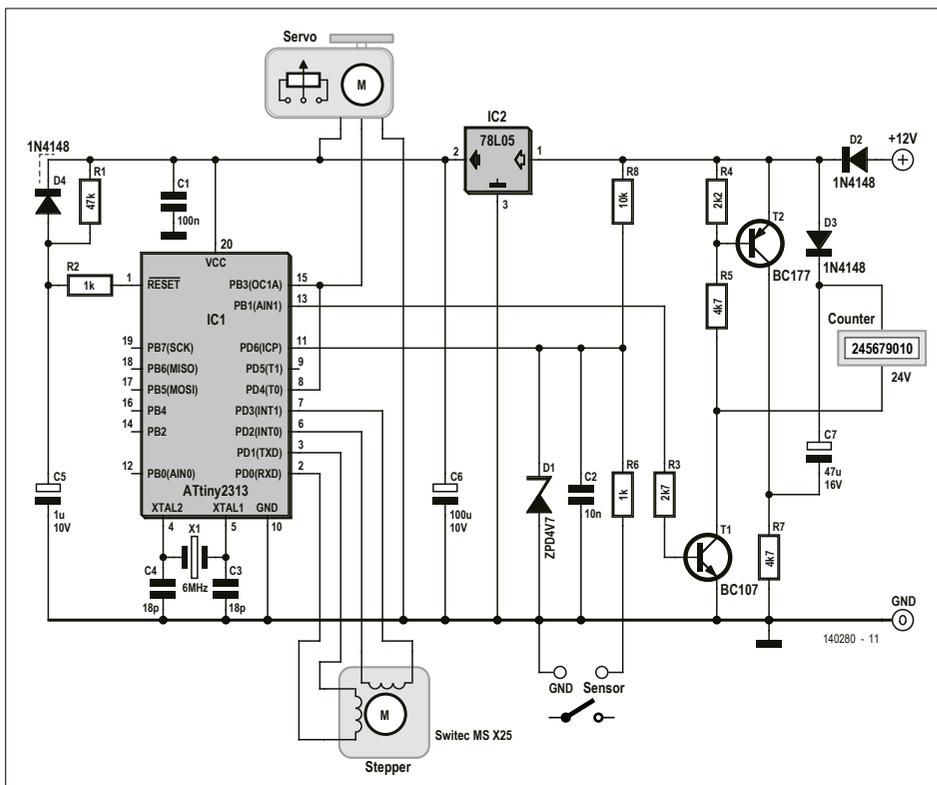


Figure 1. Le µC commande directement un servomoteur ou un moteur pas à pas.

360°, mais sur 315°, en 1 260 pas. Doté de butées solides, ses couples de positionnement et de maintien sont élevés mais il ne donne pas d'information de retour de la position du curseur. Au démarrage, on fait faire au moteur pas à pas un mouvement de 1 260 pas vers la gauche, jusqu'à ce qu'il soit en butée, indépendamment de la position réelle de l'aiguille. En une demi-seconde, il se retrouve donc à 0. La commande du moteur pas à pas se fait en 8 phases par le biais de deux bobines qu'un ATtiny2313 d'Atmel attaquera directement.

Le matériel simple de la **figure 1** permet d'utiliser soit un servo (avec débattement de 180°), soit un moteur pas-à-pas à débattement de 315°. On retrouve, autour de l'ATtiny2313, le réseau d'initialisation (*Reset*), le filtre d'entrée RC (R6/C2) pour l'entrée du capteur, et le quartz.

En fonction de l'application, on connectera les bobines du moteur pas à pas ou du servo directement au µC. Au lieu du moteur pas-à-pas spécial, on pourrait utiliser un petit moteur pas-à-pas bipolaire normal à condition de bloquer le point 0 par une butée mécanique. Un moteur pas à pas sans réducteur, à pas de 1,8°, bat l'angle d'affichage de 315° en 175 pas. En règle générale, le couple développé est si puissant qu'il est possible de réduire sensiblement l'intensité du courant de bobine. La commande directe des bobines par le µC reste toutefois inhabituelle.

Outre l'affichage de la vitesse, il restait à afficher la distance parcourue. On utilise pour cela un compteur électromagnétique qui affiche en permanence la valeur en sortie sans possibilité de remise à zéro. Le processeur additionne les impulsions fournies par le codeur. Lorsqu'une valeur de consigne est atteinte, il émet une impulsion de 10 ms sur PB1. Il n'existe (pratiquement) pas de compteurs électromagnétiques alimentés par 5 V_{CC}, la plupart [2] fonctionnent entre 10 et 30 V. C'est pourquoi les impulsions du contrôleur sont amplifiées par le circuit autour de T1 et T2. Le compteur utilisé est un exemplaire 24 V ; d'autres compteurs pourraient éventuellement être commandés directement par le µC ou par le seul T1.

J'ai monté ce circuit sur une plaque à trous casée sans problème dans le boîtier d'origine (**fig 2**). Le moteur pas-à-pas ou le servo n'ont aucune difficulté à mouvoir l'équipage mobile du vieux tachymètre. À

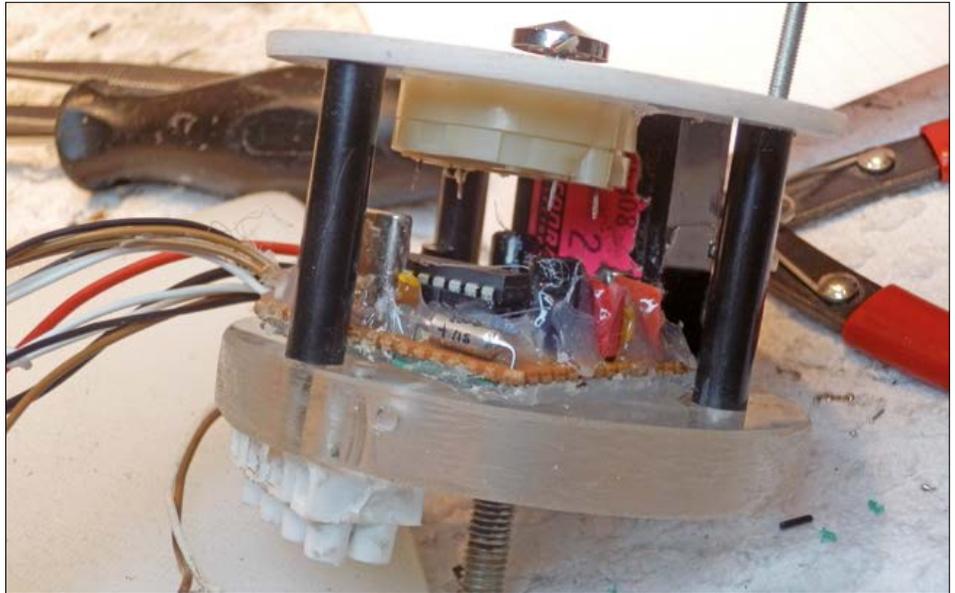


Figure 2. L'électronique casée dans le boîtier d'origine.

l'usage, le bouton d'initialisation ajouté à titre de sécurité s'est révélé inutile. Le moteur pas à pas maintient la position de l'aiguille même en cas de chocs violents et de fortes vibrations.

Le logiciel

Le logiciel (richement commenté) [3] est plus intéressant que le matériel. Comme

il vous faudra sûrement l'adapter aux composants que vous utiliserez, je vous recommande de l'examiner en détail. La **figure 3** en donne l'ordinogramme. La fréquence du quartz de 6 MHz est ramenée à 93 750 Hz par division par 64 (registre TCCR1B) et utilisée comme signal d'horloge par Timer 1. Ce dernier travaille en mode MLI (PWM) à 10 bits

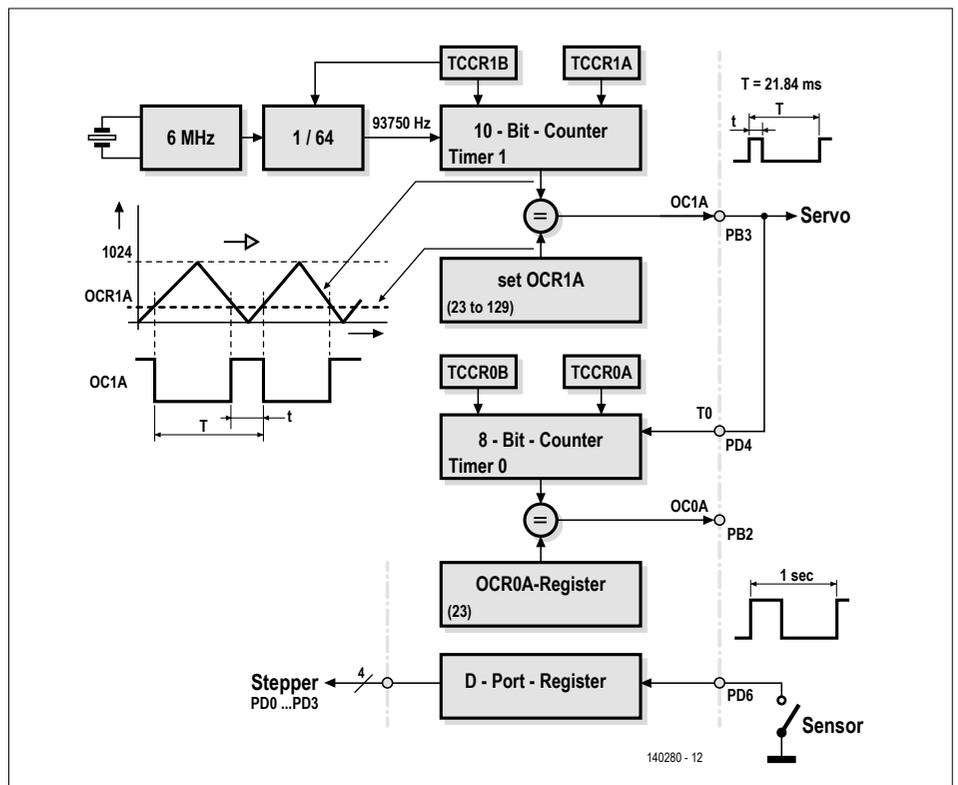


Figure 3. Ordinogramme du logiciel.

(reg. TCCR1A). Dans le registre TCNT1, on procède à un comptage des impulsions de 0 à 1023, suivi d'un décomptage jusqu'à 0. À ladite fréquence de 93 750 Hz et avec 2 x 1024 pas de comptage, cela nous donne un signal triangulaire d'une période de 21,85 ms ($2 \times 1024 / 93750$). Les servos sont normalement conçus pour des longueurs d'impulsion de 20 ms, mais ils peuvent traiter des valeurs comprises entre 15 et 35 ms. Pour avoir 20 ms exactement, il faudrait utiliser un quartz de 6,5536 MHz.

Lorsque la valeur décroissante du registre TCNT1 atteint la valeur dans le registre OCR1A, la sortie OC1A (PB3) est mise au niveau haut, et, en cas de valeur croissante dans le registre TCNT1, cette ligne est remise au niveau bas. La ligne OC1A = PB3 présente un signal rectangulaire de période de 21,85 ms et dont la largeur d'impulsion dépend de la valeur que contient OCR1A. Pour 23, la largeur d'impulsion est de 0,5 ms environ, elle atteint 2,7 ms pour un contenu de 129. Le signal disponible sur PB3 est capable d'attaquer directement un servo de modélisme, il suffit d'adapter le contenu du registre OCR1A pour modifier le réglage du servo. L'angle de rotation classique de 180° est parcouru en 106 pas (129-23). Avec le moteur pas à pas, le registre OCR1A est paramétré à la valeur fixe de 25 pour avoir des impulsions proches de 50 Hz. Le signal 21,85 ms d'OC1A (PB3) arrive (par cavalier) à l'entrée de T0 (DP4) du Timer 0 qui fonctionne en mode CTC (registres TCCR0A et TCCR0B). Il additionne alors les impulsions d'entrée à T0 jusqu'à ce que soit atteinte la valeur de référence du registre OCR0A, commute la sortie OC0A (PB2), se remet à 0 et redémarre. Sur PB2, il émet un signal carré symétrique, d'une période de 1 s (1,00464 s plus exactement) lorsque la valeur de référence de OCR0A est de 23. Le choix de cette valeur de 23 (au lieu de 25) permet d'égaliser la commande par une période de 21,85 ms au lieu de 20. La routine des lignes 89 à 96 interroge l'ILS, signal d'entrée sur PD6 (interrogation classique ou *polling*). Deux boucles courtes permettent de gommer les rebonds du contact. Si l'impulsion d'entrée est fiable, les variables de comptage `speed` (vitesse) et `way` (distance) sont incrémentées.

La routine d'analyse (lignes 61 à 85), au début de la phase de niveau bas sur PB2, 1'/sec donc, enregistre dans la variable

`speed_new` la valeur mesurée contenue à cet instant dans la variable `speed`. La valeur de mesure précédente est transférée dans `speed_middle`, l'antépénultième étant dans `speed_old`. On dispose donc de trois mesures à 1 s d'intervalle que l'on intègre lors de l'analyse. Ces valeurs de mesure sont mises à jour par roulement 1'/sec.

Après conversion mathématique, cette valeur moyenne sert de valeur d'affichage (ligne 70). La formule de conversion nécessaire dépend de l'application ; voici un exemple pour le codeur spécifié : à 10 Hz = 2,5 nœuds, l'aiguille doit être proche de la butée gauche et à 55 Hz = 10 nœuds, proche de la fin d'échelle à droite. Pour votre propre montage, vous utiliserez sûrement une autre formule mathématique. La valeur d'affichage après conversion doit se retrouver dans la variable `speed` (pour le servo 23 à 129, dans le cas du moteur pas à pas 0 à 1260).

La variable de comptage `way` compte directement les impulsions du codeur. Lorsque le seuil (nombre) prédéterminé est atteint (ici 1820 pour 0,1 mile nautique), une impulsion de 10 ms est émise sur PB1 à destination du compteur électromagnétique et `way` est remise à 0.

Moteur pas-à-pas ou servo ?

Avec un servo, on transfère simplement la valeur de la variable `speed` dans le registre OCR1A, alors que dans le cas d'un moteur pas à pas `speed` fait office de variable de transfert lors de l'appel de la fonction `MOTOR()`. Ensuite, `speed` est remise à 0, et le comptage redémarre. La variable auxiliaire `flag` limite le rythme de l'analyse à une par seconde (1 cycle de mesure/sec). Le temps requis par le traitement de la valeur mesurée allonge l'intervalle de mesure. Pour compenser un peu cela, on ne calcule qu'avec 1 s, alors qu'en fait il faut 1,00464 s.

Les sorties PD0 à PD3 commandent le moteur pas à pas ; la fonction `MOTOR()` sert à piloter le moteur pas à pas. On enregistre, dans les variables `actual_position` et `coilstep`, la position réelle

(0-1260) et la dernière position de phase de la commande à 8 phases du moteur (0-7). La position de consigne est transmise par le biais de la variable locale `new_position`. La matrice `coil[n]` comporte les 8 valeurs de sortie de la commande en phase du port D. Le fait que PD4...6 soient des entrées évite qu'une attribution d'un 0 soit source de problèmes ; ce qui évite d'avoir à masquer le port D. La variable `direction` calcule le sens de rotation. Ensuite, on fait avancer le moteur pas à pas jusqu'à ce que les variables `actual_position` et `new_position` correspondent. Les lignes 114 à 117 servent au paramétrage de la rotation de phase du moteur. La vitesse de rotation maximale (moins de 5 ms d'un pas au suivant) oblige à insérer une boucle de temporisation (ligne 120). La position réelle est enregistrée dans des variables globales `actual_position` et `coilstep`, de sorte qu'elle est à nouveau disponible lors de l'appel suivant de la fonction.

Autres tâches

Le logiciel, en C++ sous AVR Studio 4.12 est facile à adapter à d'autres applications. J'en ai une version différente, pour remplacer le tachymètre mécanique d'une moto ancienne : le couple aimant permanent sur la jante + ILS sur la fourche avant délivre des impulsions rectangulaires, dont la fréquence est proportionnelle à la vitesse de la moto.

J'ai également remplacé le compte-tours mécanique d'un vieil utilitaire *Unimog*. Les impulsions du codeur sont dérivées de la borne W de l'alternateur, converties ensuite par diode Zener et résistance série en signal carré de 5 V. Il reste à adapter la temporisation et la durée de la mesure.

Ni l'électronique ni le programme ne posent le plus de problèmes, le défi est surtout mécanique : il faut du doigté pour qu'un vieil instrument associé à un circuit moderne ne perde pas son cachet.

(140280 - version française : Guy Raedersdorf)

Liens

- [1] Moteur pas-à-pas SWITEC MS X25 sur eBay (cherchez stepper).
- [2] www.gemmer-zaehlerbau.de
- [3] Logiciel en C www.elektor.magazine.de/140280

étiquettes antivol

contrôle d'accès sous Bascom

Burkhard Kainka (Allemagne)

La carte Arduino Uno, le *shield* d'Elektor et le langage Bascom forment une équipe solide pour mener à bien les applications les plus diverses. Cette fois ce sera un contrôle d'accès sans contact au moyen d'étiquettes antivol, dont un programme sophistiqué évalue la résonance.



Quand vous effectuez un achat dans un grand magasin, surtout quand il s'agit de vêtements, vous voyez la caissière soumettre ces étranges étiquettes antivol à un traitement spécial pour éviter le déclenchement de l'alarme quand vous sortez avec la marchandise que vous venez de payer. J'ai récupéré une petite collec-



Figure 1. Une étiquette AM est faite de deux résonateurs et d'un ruban magnétique (plus court).

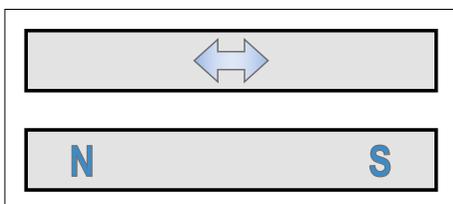


Figure 2. Principe de l'étiquette AM.

tion de ces étiquettes magnétiques (ou acousto-magnétiques), ça peut toujours servir... Et sur l'internet, j'ai trouvé assez d'explications pour expérimenter et les utiliser moi-même.

L'étiquette est constituée de deux ou plusieurs rubans métalliques (**fig. 1**). Un ruban magnétique souple (deux dans certains cas) de longueur définie présente une résonance mécanique à 58 kHz. Le ruban est d'un métal amorphe, très spécial, également appelé *verre métallique*. Pour pouvoir coupler cette résonance à un champ magnétique alternatif externe, il faut polariser le résonateur. C'est la fonction d'un second ruban en matériau magnétique dur que l'on magnétise (**fig. 2**). Dans cet état, le résonateur répond à un champ magnétique de 58 kHz, peut entrer en oscillation et rayonne lui-même un champ alternatif détectable, pour déclencher une alarme. Pour empêcher ce déclenchement, la caissière actionne un démagnétiseur.

Activation

C'est donc une étiquette inactivée que l'on emporte. Pour la réactiver, il suffit de passer l'étiquette au-dessus d'un aimant puissant qui rétablit la résonance. L'effet est facilement démontrable (**fig. 3**) au moyen d'un générateur de signaux, d'une bobine de quelque 100 spires et d'un oscilloscope. On constate un pic de résonance à 58 kHz, alors qu'une étiquette démagnétisée ne réagit pratique-

ment pas. Les oscillations mécaniques produisent un rayonnement sonore perceptible avec un détecteur à ultrasons.

J'ai imaginé une serrure électronique commandée par une telle étiquette, construite avec l'Arduino Uno. Première idée : un générateur à synthèse numérique directe ou DDS (*Direct Digital Synthesis*) asservi + un redresseur de mesure complexe, lu par le μC . Il y a plus simple : l'ATmega328 met à disposition, sur sa sortie Timer OCC1A, un oscillateur de caractéristiques adéquates, pilotable avec précision. On se passe de redresseur de mesure en appliquant une tension alternative (max. 0,3 V) directement sur une entrée analogique, mesurée assez souvent et en moyennant les résultats. Toutes les tensions négatives sont considérées comme nulles, on dispose donc d'un redresseur de mesure quasi idéal. L'ensemble du circuit (**fig. 4**) est constitué de la carte Arduino Uno et du *shield* d'Elektor [1]. L'écran LCD affiche les résultats de mesure. Le signal de commutation de la serrure électronique apparaît sur PC2 et est visualisé par la LED2 du *shield*.

Il nous faut encore une résistance et une bobine convenable. La bobine utilisée (**fig. 5**) est une bobine Ondes Moyennes pour un bâtonnet ferrite de 10 mm – un peu aplatie – qui correspond exactement à la taille de l'étiquette. C'est la même bobine que celle du présélec-

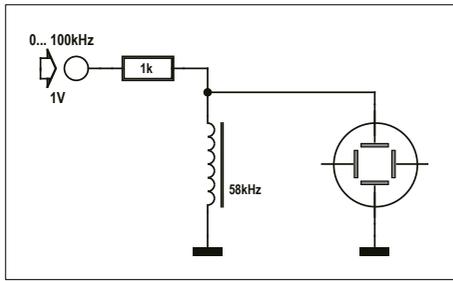


Figure 3. Mesure de résonance avec générateur de signal et oscilloscope.



Figure 5. La bobine Ondes Moyennes.

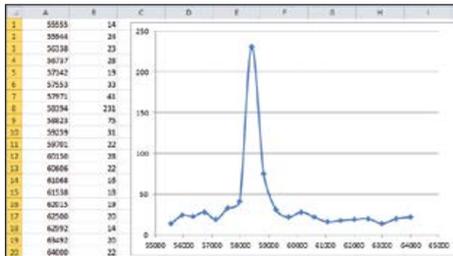


Figure 6. La courbe de résonance.

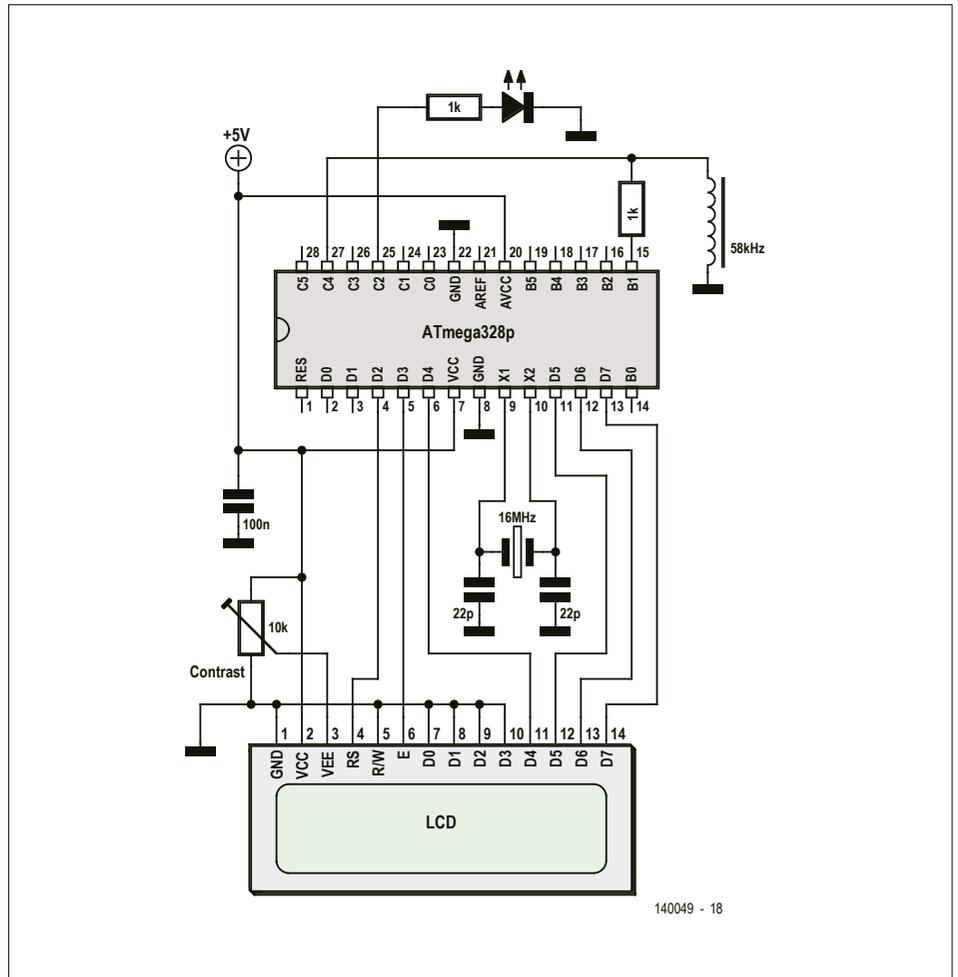


Figure 4. Analyse par l'ATmega328.

teur du récepteur de radio logicielle SDR d'Elektor [2]. Vous pouvez fabriquer votre propre bobine de 80 à 100 spires de CuL de 0,2, ou dépiauter une vieille radio. L'important est que la bobine soit bobinée à capacité mini, qu'elle ait approximativement l'inductance requise et qu'elle ne présente pas déjà une résonance propre dans la plage concernée.

Le logiciel

Le logiciel (**listage 1**), écrit en Bascom, utilise le Timer 1. Le programme délivre une série de 20 fréquences croissantes autour de la plage de 58 kHz. Pour chaque fréquence, on mesure sur l'ADC4 64 valeurs et on calcule leur moyenne. Les résultats sont stockés dans une matrice. Au cours de la mesure, la fréquence et l'amplitude d'oscillation mesurée sont envoyées à l'ordinateur par l'interface série, afin de pouvoir évaluer la courbe de résonance (**fig. 6**). Si vous souhaitez suivre

sur l'afficheur ce qui se passe, réactivez dans le code source les instructions maintenant transformées en commentaires.

Après la mesure, le programme identifie dans la matrice des valeurs de mesure le maximum de l'amplitude et la fréquence correspondante. La fréquence de résonance subit un nouveau réglage de façon à ce que l'on puisse, le cas échéant, voir le résultat sur un oscilloscope. Les valeurs de mesure identifiées sont affichées. Il reste à décider de la validité de la clé. Si la fréquence de résonance se situait entre 57 et 59 kHz et

qu'un seuil d'amplitude déterminé a été atteint, la serrure s'ouvre et la LED rouge s'allume. La mesure est alors répétée et donne le même résultat, tant que l'étiquette reste engagée dans la bobine.

Une clé ne conserve son utilité qu'aussi longtemps que personne d'autre n'en connaît l'existence ou ne la possède. Si vous êtes le seul électronicien pur-sang du quartier, vous pouvez utiliser cette serrure en toute sécurité, sans oublier toutefois qu'il y a beaucoup de clés poten-



tielles en circulation. Pour personnaliser la vôtre, raccourcissez-en un peu les rubans, leur fréquence de résonance augmentera. Ceci permet aussi de réaliser une serrure

à plusieurs fonctions activées chacune par des clés différentes. ◀

(140477 – version française : Guy Raedersdorf)

Liens

[1] www.elektor-magazine.fr/140009

[2] www.elektor-magazine.fr/090615

Listage 1. Mesure de la fréquence de résonance.

```

'-----
'UNO_AMetikett.BAS  58 kHz
'-----
$regfile = "m328pdef.dat"          'ATmega328p
$crystal = 16000000                '16 MHz
$baud = 9600
$hwstack = 32
$swstack = 32
$framesize = 64

Dim D As Long
Dim F As Long
Dim N As Byte
Dim U As Word
Dim Um As Word
Dim Ux(50) As Word
Dim I As Word

Led1 Alias Portc.2
Led2 Alias Portb.2
S1 Alias Pinc.0
S2 Alias Pinc.1
Portc.0 = 1                        'Pullup
Portc.1 = 1
Config Led1 = Output
Config Led2 = Output

Config Lcdpin = Pin , Db4 = Portd.4 , Db5 = Portd.5
      , Db6 = Portd.6 , Db7 = Portd.7 , E = Portd.3 ,
      Rs = Portd.2
Config Lcd = 16 * 2
Cls
Cursor Off

Config Timer1 = Pwm , Prescale = 1 , Pwm = 10 ,
      Compare A Pwm = Clear Up

Tccr1a = &B10000010                'Phase-correct PWM, Top=ICR1
Tccr1b = &B00010001                'Prescaler=1

Config Adc = Single , Prescaler = Auto , Reference =
      Internal_1.1
Start Adc

Do
  For I = 1 To 20
    D = 145 - I
    F = 16000000 / D
    F = F / 2
    Print F ;
    Print " ";
    Print Chr(9);

    ' Locate 1 , 1
    ' Lcd F
    ' Lcd " Hz      "
    Icr1 = D
    Ocr1a = D / 2
    Waitms 10
    Um = 0
    For N = 1 To 64
      U = Getadc(4)
      Um = Um + U
    Next N
    Um = Um / 32
    Ux(i) = Um
    Print Um
    ' Locate 2 , 1
    ' Lcd Um
    ' Lcd " mV     "
    ' Waitms 500
  Next I
  Max(ux(1) , Um , I)
  D = 145 - I
  F = 16000000 / D
  F = F / 2
  Locate 1 , 1
  Lcd F
  Lcd " Hz      "
  Icr1 = D
  Ocr1a = D / 2
  Locate 2 , 1
  Lcd Um
  Lcd " mV     "
  If F > 57000 And F < 59000 And Um > 50 Then
    Waitms 50
    For N = 1 To 255
      U = Getadc(4)
      Um = Um + U
    Next N
    Um = Um / 127
    If Um > 50 Then Led1 = 1
    Lcd F
    Lcd " Hz      "
    Icr1 = D
    Ocr1a = D / 2
    Locate 2 , 1
    Lcd Um
    Lcd " mV     "
  Else
    Led1 = 0
  End If
  Waitms 1000
Loop

```

e-BoB BL600

(2^e partie)

Éditer, compiler, transférer un programme avec le module Bluetooth Low Energy

Jennifer Aubinais (Paris) elektor@aubinais.net

À en croire certaines prévisions, on ne parlera plus bientôt de *l'internet des objets*, mais de *l'internet de tous les objets*. Cela impliquera une généralisation de la communication sans fil et de la faible consommation d'énergie des circuits mis en œuvre pour connecter ces objets. Précisément la mission de cette carte de communication radio à consommation ultra-faible, que l'on peut utiliser comme un ordinateur miniature, programmable aisément en *smartBASIC*.

Dans le premier de cette série d'articles consacrés à l'e-BoB BL600 [1], nous avons présenté le matériel : d'un côté le port UART, pour communiquer avec un objet connecté, dans notre cas un PC via l'e-BoB FT232 ; de l'autre côté sept entrées/sorties utilisables à notre guise : deux entrées ADC, le port I²C, le port SPI. Puis nous avons vu comment, grâce à la fonction OTA (*Over The Air*), transférer via les ondes radio à notre e-BoB BL600, un premier programme UART, téléchargé avec un téléphone Android.

Ici nous proposons un premier programme d'application après nous être familiarisés avec les outils d'édition, de compilation puis de transfert. Ce sera un simple chenillard, mais un chenillard *Bluetooth Low Energy* ! Nous avancerons pas-à-pas et même orteil par orteil, car si les possibilités de ce module sont remarquables, il faut l'appivoiser. Dans le 3^e article, le mois prochain, nous découvrirons le principe de la programmation des événements en *smartBASIC*, grâce auquel vous créerez vos propres projets.

Éditer

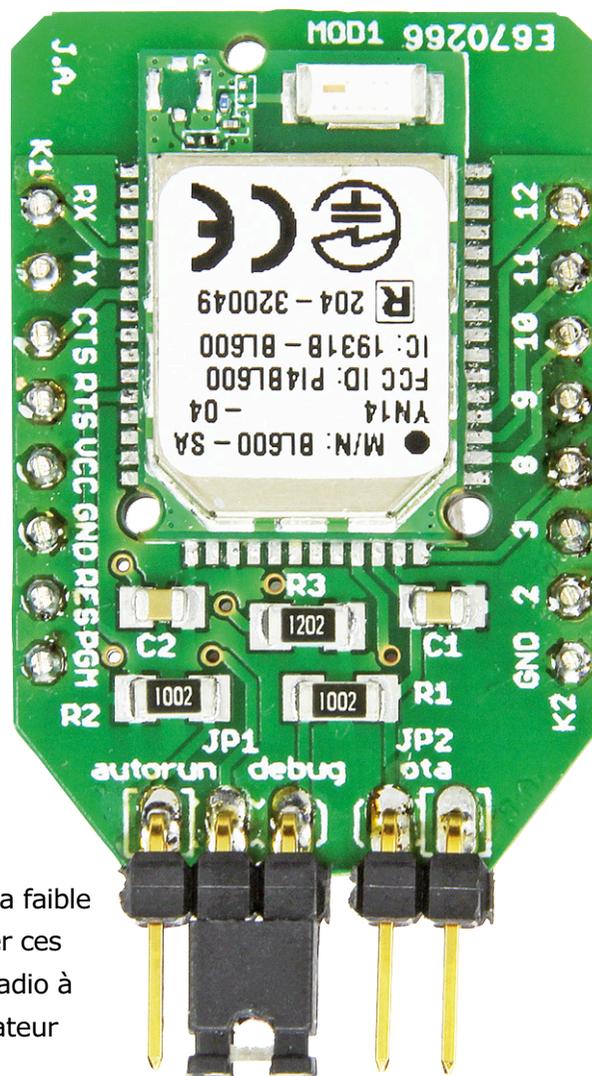
Nous commençons par télécharger le progiciel à jour du BL600 (*Firmware Files version 1.5.70.0 - Revision 5* [2]) ainsi qu'un éditeur ; Laird Technologies conseille Notepad++ (fig. 1) [3] qui connaît la syntaxe du langage *smartBASIC*. Pour obtenir un *login* dans la section *SOFTWARE DOWNLOADS* du site de Laird Technologies, cliquez sur *If you need*

credentials, please click here et patientez quelques jours.

Après vous être connecté, téléchargez *Firmware Files version 1.5.70.0 - Revision 5* qu'il faudra décompresser. Il contient (fig. 2) :

- des exemples de programmes dans le répertoire *smartBASIC_Sample_Apps*
- le programme *UwTerminal.exe* dans le répertoire *smartBASIC_Sample_Apps*
- la bibliothèque dans le répertoire *smartBASIC_Sample_Apps/lib*
- des exemples (*UserManualExampleCode*)
- la configuration spécifique de Notepad++ pour le *smartBASIC* (*smartBASIC(notepad++.xml)*)

Une fois l'éditeur Notepad++ installé, lancer l'application. Dans le menu, choisir *language* (fig. 4) puis cliquez sur *Define your language*. Cliquez sur *Import*, choi-



sissez le fichier *smartBASIC(notepad++)*. *xml* du fichier ZIP et relancez Notepad++. Choisissez enfin *language*, cliquez sur *User language (fig. 5)* et choisissez *smartBasic* et fermez.

Notepad++ est configuré pour afficher en couleurs la syntaxe du code smartBASIC. Essayez par exemple le programme *LedChaser.sb (fig. 6)* sur lequel nous reviendrons à la fin de cet article. En revanche, nous ne l'utiliserons pas maintenant pour apprendre à compiler dans le prochain paragraphe ; nous utiliserons pour cela le programme *upass* déjà utilisé dans le premier article du e-BOB BL600.

Compiler

Avant de poursuivre, il faut savoir que l'e-BoB BL600 doit être connecté au PC par l'intermédiaire d'une interface UART/USB. Pour cela, je recommande d'utiliser l'e-BoB FT232. C'est le premier de la série des e-BoB d'Elektor, une simple et très efficace passerelle série-USB proposée dans le numéro de septembre 2011 [4]. Le tandem formé par nos deux e-BoB apparaît sur la figure 14 et sur la photo adjacente. Nous reviendrons sur cet agencement quand nous parlerons de notre exemple d'application, le chenillard. Mais d'abord, le moment est venu de se familiariser avec la compilation. Pour cela, nous utiliserons aussi le programme *UwTerminal.exe* qui se trouve dans le paquet de logiciels téléchargé chez Laird.

À la fin de l'article du mois dernier, notre e-BoB BL600 était resté en mode *AutoRun*, dans lequel il exécute le programme UART préalablement chargé dans le module. Pour compiler (et transférer), le cavalier JP1 doit être en position *cmd* (nommée *debug* dans la version précédente de l'e-BoB BL600) tandis que JP2 (*ota*) n'est pas implanté. Initialisez le module (*reset*) soit avec le bouton, soit en coupant brièvement l'alimentation. Lancez *UwTerminal.exe* puis choisir le bon port COM de l'e-BoB FT232 et un débit de 9600 bauds (fig. 7).

Tapez sur *Entrée* lorsque l'écran noir apparaît (fig. 8). Même s'il est possible de tout faire en même temps, nous procéderons pas à pas et commençons par compiler. Faites un clic droit sur le fond noir. Une fenêtre s'ouvre. Choisissez *XCompile (fig. 9)* puis le fichier *\$autorun\$.upass.vsp.sb*. Une fenêtre



Figure 2. Les options sur la page d'accueil de Laird dont vous serez bientôt un visiteur familier.

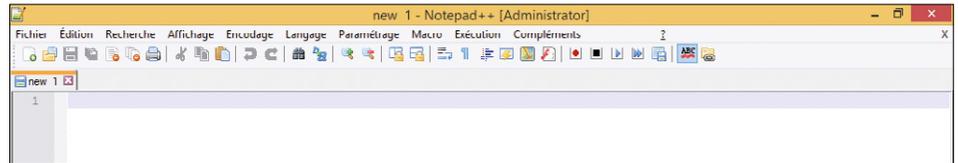


Figure 3. L'éditeur de Notepad++ avec lequel vous écrirez vos applications.

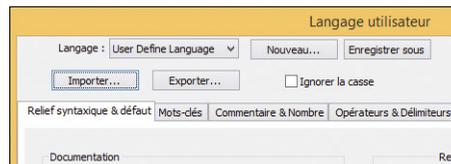


Figure 4. Pour que Notepad++ apprenne la syntaxe de smartBASIC...

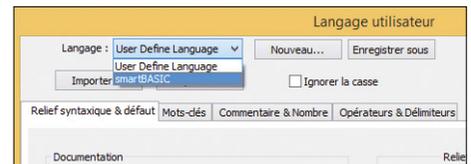


Figure 5. ... il suffit de charger le fichier de configuration adéquat fourni par Laird.

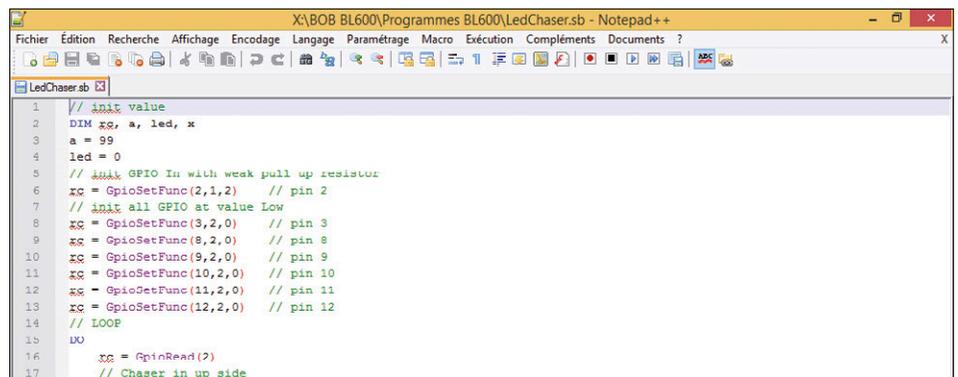


Figure 6. La fenêtre d'édition du code (voir le listage complet à la fin de l'article).

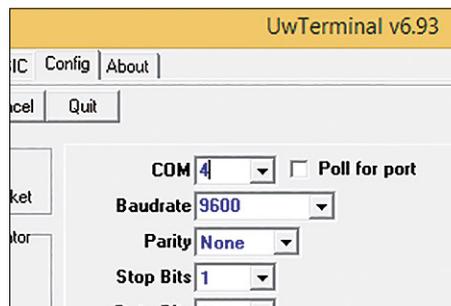


Figure 7. Paramétrage du programme de communication *UwTerminal*



Figure 8. Les deux 00 sur l'écran noir d'*UwTerminal*, c'est bon signe !

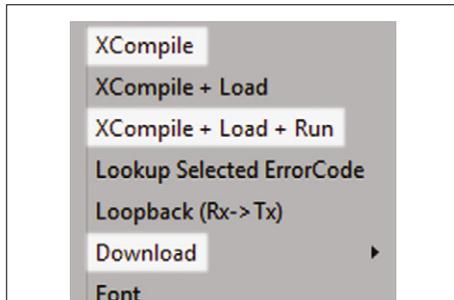


Figure 9. Menu Xcompile | Download | XCompile + Load + Run

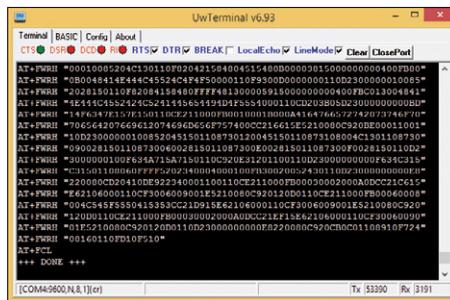


Figure 11. Le transfert du programme compilé se termine par le message +++ DONE +++.

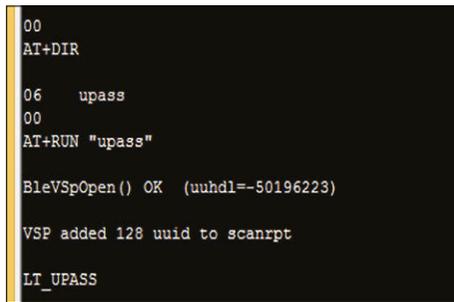


Figure 10. Le résultat de la compilation s'affiche dans la fenêtre UwTerminal.



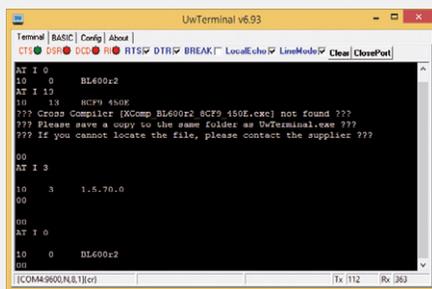
Figure 12. AT+DIR donne la liste des programmes et AT+RUN lance votre programme.



Les possibilités de ce module sont remarquables, il faut l'appriivoiser. Avançons pas-à-pas et même orteil par orteil...

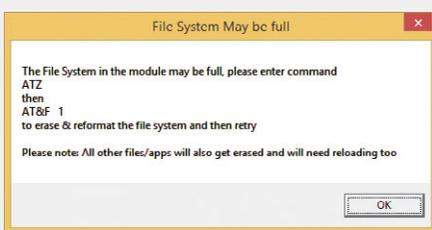
Erreur : Cross Compiler [XComp_BL600.....exe] not found ?

Cette erreur intervient lorsque le progiciel (*firmware*) de votre BL600 ne correspond pas au paquet BL600 téléchargé du site de Laird Technologies. Les commandes pour identifier le progiciel de votre BL600 :



AT I 3 pour le progiciel
AT I 0 pour la révision
Une fois identifiés le progiciel et la révision de votre BL600, téléchargez le paquet correspondant du site de Laird Technologies [2]. Refaites la compilation décrite.

Erreur : The File System in the module may be full



Cette erreur est due au manque de mémoire pour le nouveau programme. Effacez la mémoire du BL600 par les commandes ATZ puis AT&F 1 et téléchargez à nouveau votre programme smartBASIC.

de compilation s'ouvre, puis le résultat de la compilation s'affiche (fig. 10). La compilation crée le fichier \$autorun\$.upass.vsp.uwc fourni en téléchargement avec l'article précédent [1]. Vous voici déjà en mesure de le refaire vous-même. Et bientôt, quand vous serez familier de la procédure décrite, vous le modifierez comme bon vous semble.

Transférer

Nous n'allons pas utiliser le service BLE Over The Air pour télécharger notre programme précédemment compilé sur notre e-BoB BL600. Comme ce dernier est connecté au PC, c'est le moyen le plus rapide pour transférer le programme compilé. De plus, il s'agit pour moi de vous présenter en détail les étapes qui vous aideront à écrire vos propres programmes et... à faire vos propres erreurs, car c'est ainsi que l'on apprend le mieux. La configuration des cavaliers est JP1 en mode cmd et pas de cavalier pour JP2 (OTA). Le logiciel UwTerminal.exe étant lancé sur votre PC, il suffit d'appuyer sur la touche ENTER pour vérifier que l'e-BoB BL600 répond correctement par 00 sur l'écran noir de UwTerminal. Au besoin, une initialisation (reset) de l'e-BOB BL600 sera nécessaire.

Si vous aviez déjà téléchargé un programme dans le BL600, p. ex. avec le premier article (Over The Air), il faut effacer maintenant la mémoire de programme en tapant la commande suivante AT&F 1 qui redémarrera aussi le module. Choisissez Download (fig. 9). Puis BASIC, puis Load Precompiled BASIC et enfin le programme compilé upass.vsp.uwc. Le transfert du programme compilé se termine par le message +++ DONE +++ (fig. 11).

Pour obtenir la liste des programmes téléchargés, tapez la commande AT+DIR et pour lancer votre programme, la commande AT+RUN « upass » (fig. 12)

Nous avons utilisé UwTerminal.exe pour compiler, transférer puis exécuter le programme sur votre e-Bob BL600. Ces trois étapes peuvent être regroupées en une seule, avec la commande XCompile + Load + Run (fig. 9).

- Rappel des commandes courantes :
- AT I 0 : numéro de la révision du BL600
 - AT I 3 : version du progiciel du BL600

- AT+DIR : liste des programmes dans le BL600
- ATZ : remise à zéro du BL600
- AT&F 1 : effacement de la mémoire et redémarrage du BL600
- AT+RUN « xxxx » : exécution du programme xxxx

Commande de chenillard sans fil

Ce petit chenillard de 6 LED commandées par le BL600 change de sens de défilement quand on appuie sur le bouton. Il utilise de façon simple des entrées/sorties logiques du BL600. Nous n'aborderons pas cette fois ni la conversion analogique numérique ni la modulation de largeur d'impulsion, toutes deux possibles avec le BL600.

La liaison série entre l'e-BoB FT232 et notre nouvel e-BoB BL600 a été décrite dans l'article du mois dernier. Pour rappel, la tension de service doit être de 3,3 V. Pour les entrées/sorties de l'e-BoB BL600, rien de spécial : 6 LED avec chacune sa résistance talon de 470 Ω (valeur approximative).

Pour ce qui la programmation en BASIC, nous nous concentrerons ici sur les fonctions des entrées et sorties polyvalentes (GPIO) en étudiant le programme de ce chenillard (**listage**).

GPIOSETFUNC(nSigNum, nFunction, nSubFunc)

Cette fonction définit en deuxième argument le rôle des entrées/sorties de la broche GPIO identifiée par le premier argument.

a. Configuration en entrée logique

Le poussoir est connecté à la broche GPIO 2 polarisée au niveau haut par une résistance interne :

```
rc = GpioSetFunc(2,1,2)
```

- nSigNum = 2 : broche GPIO 2
- nFunction = 1 : port en entrée
- nSubFunc = 2 : résistance interne

b. Configuration en sortie logique

Chaque LED est connectée avec une résistance-talon à une des sorties (3, 8 à 12). Configuration de la sortie 12 p. ex. :

```
rc = GpioSetFunc(12,2,0)
```

- nSigNum = 12 : broche GPIO 12
 - nFunction = 2 : port en sortie
 - nSubFunc = 0 : sortie à l'état bas
- rc est le code retour (= 0x0000 lors du succès de la fonction).

Listage LedChaser.sb

```
// init value
DIM rc, a, led, x
a = 99
led = 0
// init GPIO In with weak pull up resistor
rc = GpioSetFunc(2,1,2) // pin 2
// init all GPIO at value Low
rc = GpioSetFunc(3,2,0) // pin 3
rc = GpioSetFunc(8,2,0) // pin 8
rc = GpioSetFunc(9,2,0) // pin 9
rc = GpioSetFunc(10,2,0) // pin 10
rc = GpioSetFunc(11,2,0) // pin 11
rc = GpioSetFunc(12,2,0) // pin 12
// LOOP
DO
  rc = GpioRead(2)
  // Chaser in up side
  IF (rc == 1) THEN
    IF (led == 0) THEN : GpioWrite(12,0) : GpioWrite(3,1) : ENDIF
    IF (led == 1) THEN : GpioWrite(3,0) : GpioWrite(8,1) : ENDIF
    IF (led == 2) THEN : GpioWrite(8,0) : GpioWrite(9,1) : ENDIF
    IF (led == 3) THEN : GpioWrite(9,0) : GpioWrite(10,1) : ENDIF
    IF (led == 4) THEN : GpioWrite(10,0) : GpioWrite(11,1) : ENDIF
    IF (led == 5) THEN : GpioWrite(11,0) : GpioWrite(12,1) : ENDIF
    led = led + 1
  IF ( led == 6) THEN : led = 0 : ENDIF
ELSE
  // Chaser in down side
  IF (led == 6) THEN : GpioWrite(3,0) : GpioWrite(12,1) : ENDIF
  IF (led == 5) THEN : GpioWrite(12,0) : GpioWrite(11,1) : ENDIF
  IF (led == 4) THEN : GpioWrite(11,0) : GpioWrite(10,1) : ENDIF
  IF (led == 3) THEN : GpioWrite(10,0) : GpioWrite(9,1) : ENDIF
  IF (led == 2) THEN : GpioWrite(9,0) : GpioWrite(8,1) : ENDIF
  IF (led == 1) THEN : GpioWrite(8,0) : GpioWrite(3,1) : ENDIF
  led = led - 1
  IF ( led == 0) THEN : led = 6 : ENDIF
ENDIF
// tempo : speed
for x = 0 to 2000
  next
DOWHILE (a != 0)
```

Liens :

- [1] e-BoB BL600 | Elektor n° 441, mars 2015, p. 34, www.elektor-magazine.fr/140270
- [2] https://laird-ews-support.desk.com/?b_id=1945
- [3] <http://notepad-plus-plus.org/>
- [4] www.elektor-magazine.fr/110553
- [5] <https://www.youtube.com/watch?v=SxwaVIOKkk8>
- [6] thermomètre sans-fil Bluetooth LE, Elektor n° 439-440, janv.-fév. 2015 p. 72 www.elektor-magazine.fr/140190

GPIOREAD(nSigNum)

Lecture (entrée logique) sur la broche GPIO identifiée par l'argument.

Le poussoir est connecté sur la broche GPIO 2 configurée en entrée avec résistance de polarisation interne. La lecture de cette entrée se fait comme suit :

`rc = GpioRead(2)`

- **nSigNum = 2** : broche GPIO 2
 - **rc** retourne la valeur du signal en entrée de la broche 2 : 0 ou 1
- Note : si l'entrée GPIO est configurée en entrée analogique, **rc** retourne la tension aux bornes de cette entrée. Dans un prochain article, nous utiliserons le convertisseur A/N.

GPIOWRITE(nSigNum, nNewValue)

Écriture (sortie logique) sur la broche GPIO identifiée par le premier argument (si le numéro de broche est invalide, rien ne se passe).

Les broches 3 et de 8 à 12 sont configurées en sorties. Dans le programme, il faudra mettre les différentes sorties à l'état 0 ou 1 :

`GpioWrite(12,0)`

- **nSigNum = 12** : broche GPIO 12
 - **nNewValue = 0** : met à l'état bas
- Note : la broche GPIO peut être configurée pour la modulation de largeur d'impulsion (*Pulse Width Modulation*). Cette fonction est limitée pour le BL600 à deux broches au choix. La valeur de **nNewValue** se situe dans la plage de 0 à N, où N est l'indice de modulation max. correspondant à un rapport cyclique de 100 %. Dans un prochain article, nous profiterons de cette possibilité du module BL600-SA utilisée à titre d'exemple d'application sur un petit véhicule télécommandé via Bluetooth [5].

Au terme de ce deuxième article consacré à un si petit module — en fait le troisième si je compte le thermomètre Bluetooth sans-fil [6] — il me reste encore bien des choses à en dire. Cela fait plus d'un an et demi que je m'amuse avec et me réjouis de partager mon expérience avec vous.

Je remercie au passage toute l'équipe de Laird Technologies pour son aide et j'espère vous retrouver le mois prochain pour programmer le BL600. ◀

(150014)

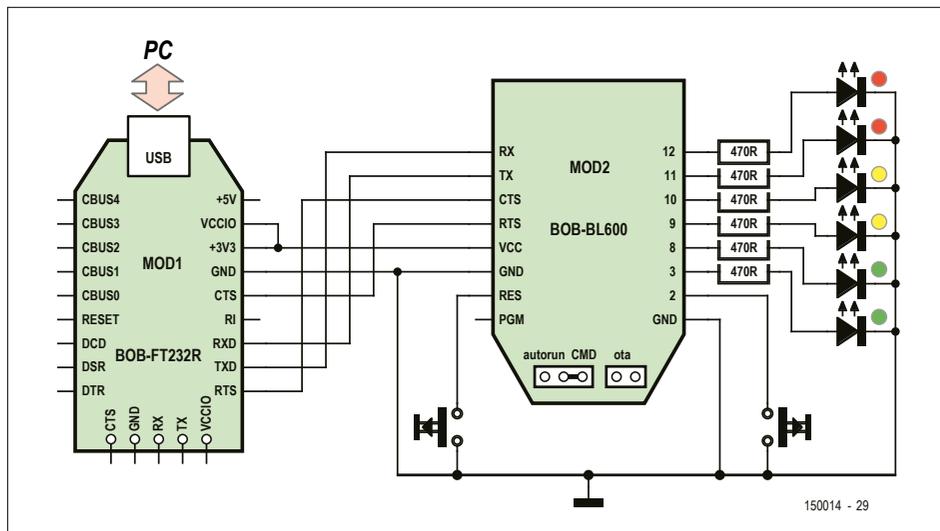


Figure 14. L'e-BoB BL600 est autonome mais, pour le programmer depuis le PC, je l'utilise en tandem avec la passerelle USB/série e-BoB FT232.

Liste des composants

Résistances :

R1 à R6 = 470 Ω

Semi-conducteurs :

D1 à D6 = LED 3 mm (couleur au choix)

Divers :

K1, K2 = poussoir

MOD1 = e-BoB FT232 assemblé 110553-91 (www.elektor.fr)

MOD2 = e-BoB BL600 assemblé 140270-91 (www.elektor.fr)

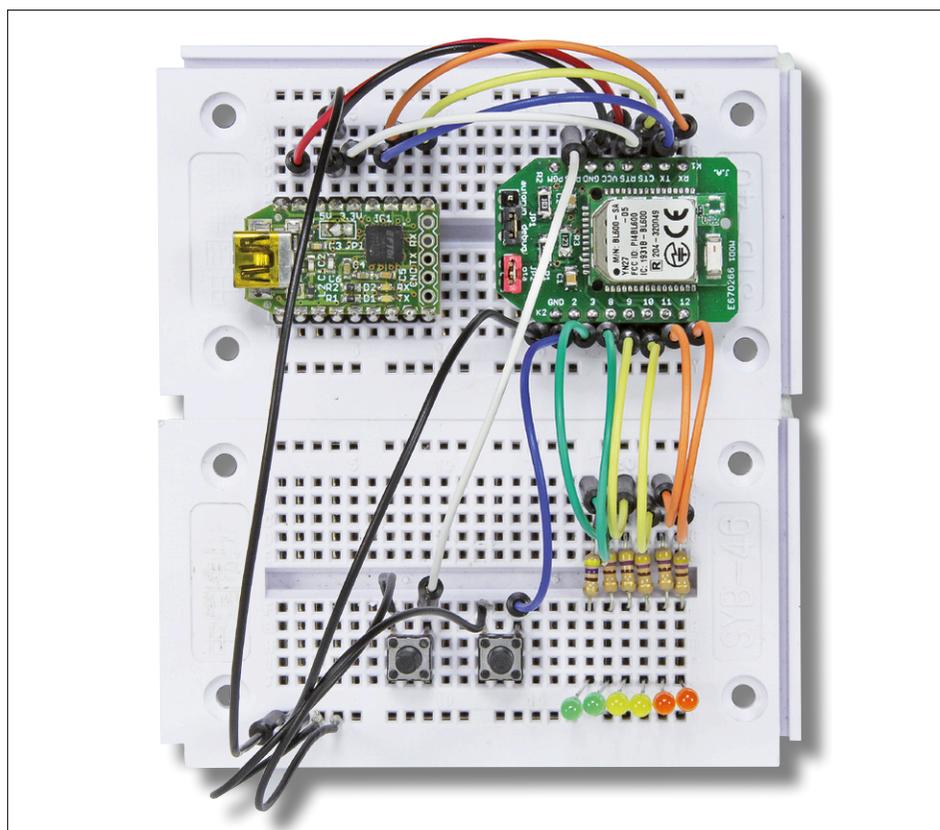


Figure 15. Le chenillard câblé sur une plaque d'expérimentation à trous.

Arduino en émetteur AM sur P.O. par induction

Burkhard Kainka
(Allemagne)

Les radios à tubes possèdent un charme particulier.

Non seulement elles sont de toute beauté, mais elles ont un son hors pair et, de plus, rappellent souvent des souvenirs agréables. D'où le nombre de fans qui collectionnent, réparent ou restaurent avec passion ces antiquités. Malheureusement, de moins en moins de stations (radio) émettent sur les bandes de MA (Modulation d'Amplitude) classiques.

Dans mon bureau, j'ai une radio à tube(s) de 1957, que j'écoute souvent sur les Petites Ondes (P.O., on parle aussi d'Ondes Moyennes). Même si la modulation d'amplitude sur les P.O. a une bande passante limitée, cela ne lui en donne pas moins sa sonorité chaude et spéciale. La nuit, vous pouvez écouter de nom-

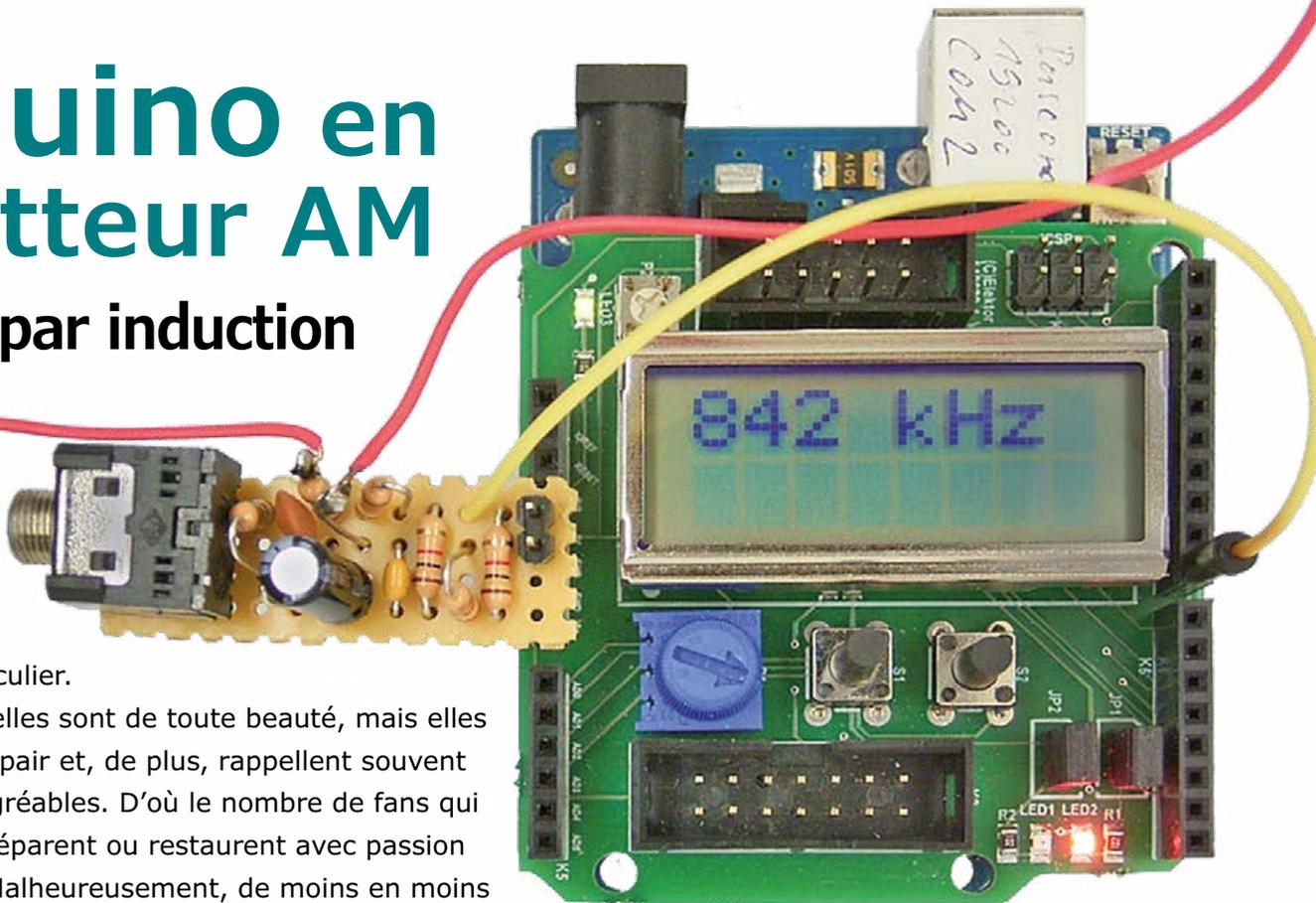
Cependant une solution légale est la transmission du signal par induction, d'où une portée faible, mais peu de risque de gêner les voisins. Il existe des règlements sur l'allocation générale des fréquences dans la gamme de 9 à 30 000 kHz pour les applications inductives de radio. Pour la gamme de fréquences de 148,50

légal donc. Dans la pratique, cela signifie que l'« antenne d'émission » devra être montée, par exemple, sous la table ou au mur et à proximité immédiate de la radio. Dans la boucle de fil, la réception est bonne ; dès que l'on en sort, l'intensité du champ diminue rapidement. On peut aussi réaliser deux spires avec la même longueur de fil ; l'intensité du champ est alors plus élevée sur une zone plus restreinte et elle chute encore plus vite en fonction de la distance.

Comment réaliser une station à MA ? Un signal de porteuse met un tube d'émission en conduction à 100% (étage de sortie de classe C), le signal BF module sa tension d'anode. À faible puissance d'émission cela marche aussi avec un transistor, qu'il soit bipolaire ou FET. Maintenant, il nous faut juste un signal RF, stabilisé par quartz, de la fréquence appropriée.

Microcontrôleur en source RF

C'est là qu'entre en scène, le μC . L'Arduino Uno possède un quartz dont on peut, par diviseur programmé, dériver des fréquences sises dans la bande P.O. Cette



▶ Malheureusement, de moins en moins de stations radio émettent sur les bandes de modulation d'amplitude classiques

breuses stations d'Europe. Malheureusement, nombre de stations locales ont mis la clé sous la porte, même la BBC a pris congé des P.O. On a bien envie alors de trouver une nouvelle utilisation aux fréquences libérées. Sus à ma station à MA privée ! Elle me permet de recevoir la BBC sur Internet et de l'écouter en P.O. comme au bon vieux temps.

Est-ce légal ? En règle générale, « on » n'aime guère les émetteurs pirates.

à 5 000 kHz il faut respecter, à une distance de 10 m, une limite de 15 dB $\mu\text{A}/\text{m}$.

L'intensité du champ magnétique n'est pas facile à mesurer, mais on peut la calculer à partir du courant d'antenne dans une boucle inductive : pour une boucle de fil de $r = 1$ m (c.-à-d. de 2 m de diamètre) à une seule spire et un courant RF limité à 0,3 mA, l'intensité du champ magnétique à une distance de 10 m n'est plus alors que de -15 dB $\mu\text{A}/\text{m}$. Parfaitement

solution simple a un petit inconvénient : l'émission ne respecte pas le pas de 9 kHz classique (10 kHz aux USA). Il ne devrait pas être sorcier de trouver quand même une fréquence libre adéquate. Nous réaliserons un petit panneau de commande avec affichage de la fréquence, à l'aide du *shield* d'Elektor [1].

Ce n'est pas tout : l'Uno possède d'origine l'amplificateur de puissance d'émission nécessaire ! On utilisera à cet effet n'importe quelle broche de port, mais avec un petit truc. On ne commande pas le registre du port, mais le registre de direction de données, pour la broche PB0 par exemple. Avec $PORTB.0 = 0$ et $DDRB.0 = 1$ (sortie), on crée un état bas à basse impédance. Si l'on fait passer ensuite $DDRB.0 = 0$ (ce que l'on fait normalement pour lire des signaux numériques), le port passe à haute impédance. Ces deux états correspondent exactement à un FET à drain ouvert. Nous pouvons maintenant appliquer à PB0 une tension de « drain » quelconque (à condition qu'elle demeure entre 0 et 5 V). Avec $DDRB.0 = 1$, elle se voit forcée à la masse, pratiquement au rythme du signal RF. Et bien sûr, cette tension de drain peut même être modulée, par un signal BF par exemple.

C'est presque prêt ! Il nous reste à peaufiner quelques aspects et à faire un rien de programmation.

Le circuit

La sortie casque d'une source de signal typique (lecteur de CD, carte son de PC, etc.) fournit habituellement jusqu'à 1 V_{eff}, soit près de 3 V_{cc}. De toute façon, c'est assez puisque l'étage de sortie doit être attaqué dans la plage de 0 à 5 V. Pas besoin donc d'amplificateur de modulation. Il suffit de quelques résistances et condensateurs pour conditionner le signal BF (**fig. 1**, à droite). Le schéma montre à gauche l'ATmega328 de l'Arduino Uno et les composants qui se trouvent sur le *shield*. On pourra bien entendu également monter l'ensemble sur une plaque d'essais.

La sortie de l'antenne comporte un filtre passe-bas simple pour atténuer les harmoniques. Des mesures et des considérations théoriques nous apprennent qu'il circule un peu moins de 0,3 mA dans la boucle d'antenne. On respecte ainsi

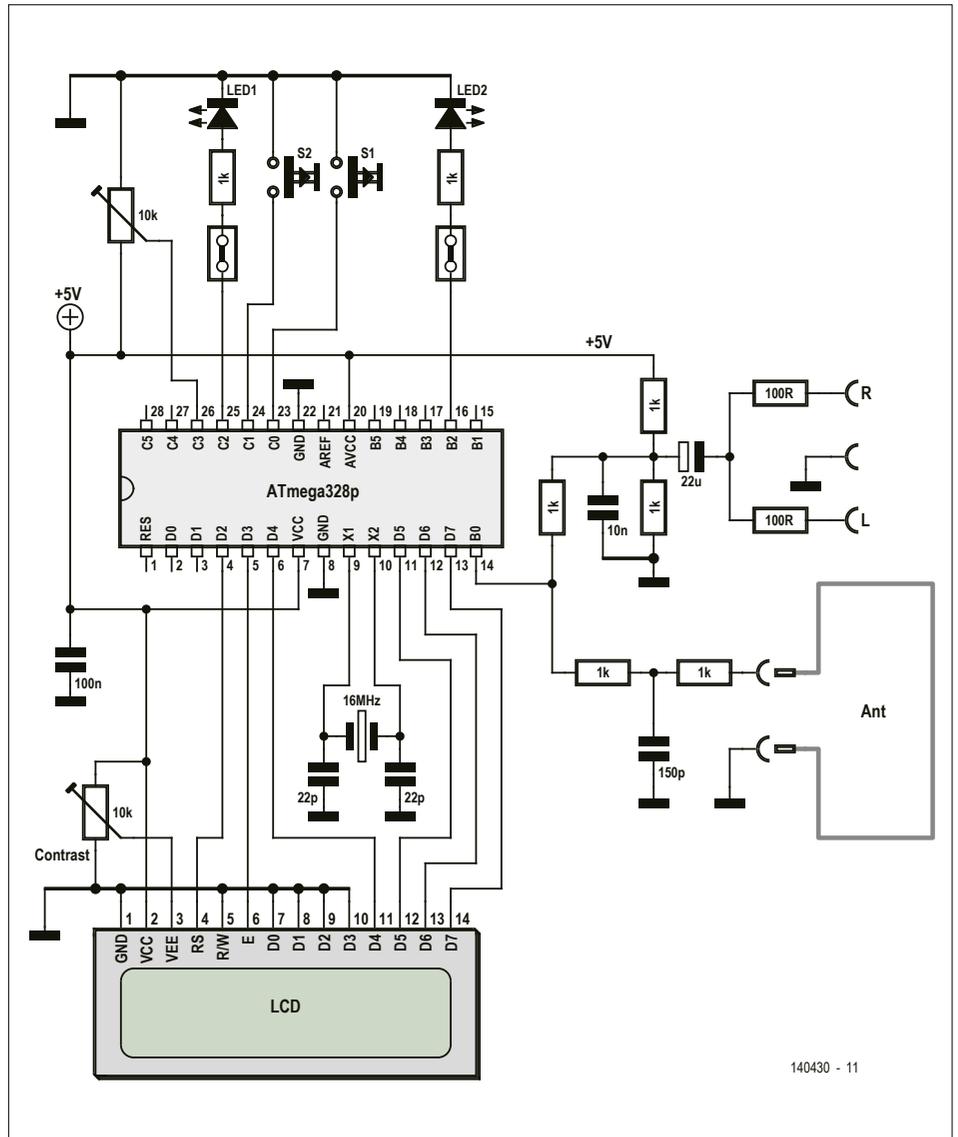


Figure 1. Schéma de l'émetteur à MA. Tous les composants de la partie gauche se trouvent sur l'Arduino Uno et le *shield* d'Elektor.

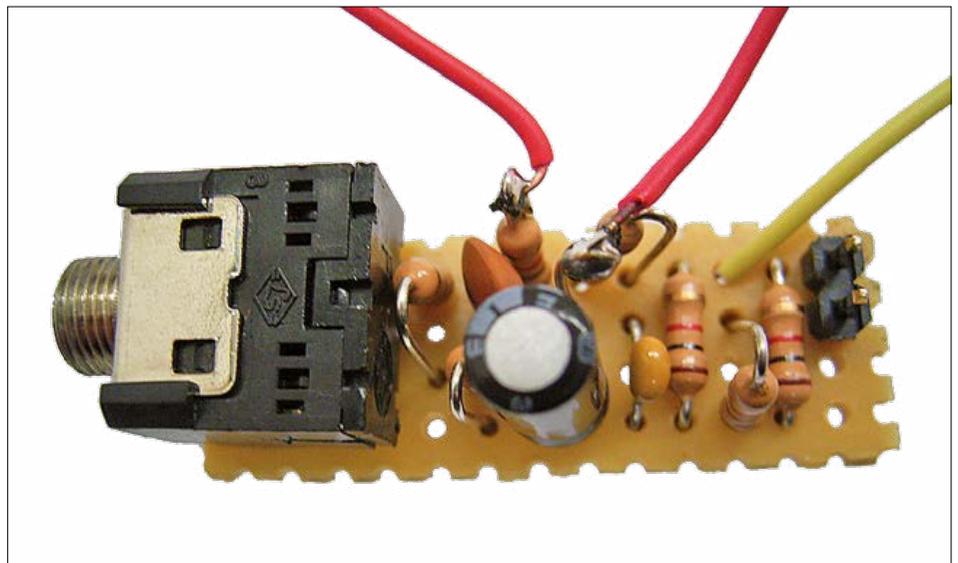


Figure 2. Les composants additionnels montés sur un morceau de platine à trous.

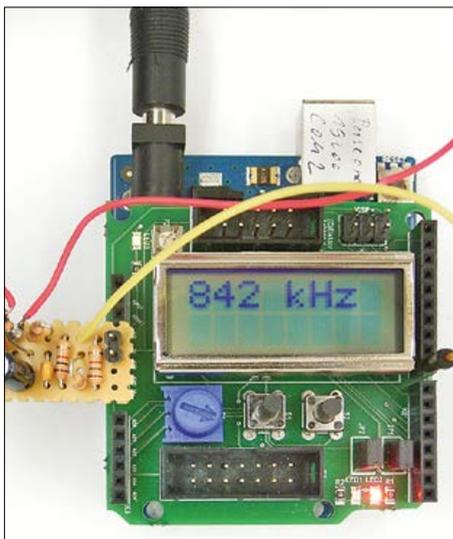
Listage 1. Production du signal RF.

```

D11:
Do          '/11, 1454 kHz
  Ddrb.0 = 1
  nop      '+4 nop
  nop
  nop
  Ddrb.0 = 0
  nop
Loop

D10:
Do          '/10, 1,6 MHz
  Ddrb.0 = 1
  nop      '+3 nop
  nop
  nop
  Ddrb.0 = 0
Loop

```



les directives de manière fiable dans la mesure où la longueur totale du fil de la boucle de l'antenne ne dépasse pas 3 m. Pour les premiers essais, utilisez une boucle de fil bien plus petite (juste 20 cm de fil par ex.).

Avec une sortie de carte son standard (3 Vcc environ), le taux de modulation est de quelque 50%, et donc pratiquement pas de surmodulation. Comme le signal à MA ne transmet que du mono, on interconnecte les canaux gauche et droit. Finalement, on a une modulation très propre sans la moindre distorsion. L'écoute sur une radio à tube

permet de retrouver le magnifique son chaud de l'ancien temps.

Le logiciel

Pour produire la fréquence porteuse, il suffit d'activer et de désactiver très rapidement le bit DDRB correspondant. Dans une simple boucle Do (**Listing 1**), programmée en Bascom, il faut sept cycles d'horloge à l'ATmega328. Ajoutons trois NOP et la fréquence d'horloge de 16 MHz se trouve divisée par dix. On émet ainsi dans le haut de la bande des P.O., à 1,6 MHz.

Ajoutez une autre instruction NOP, la boucle divise par onze, ce qui donne 1 454 kHz, etc. Dans notre programme (téléchargeable ici [2]), chaque fréquence a sa propre boucle, identifiée à chaque fois par une étiquette qui indique le facteur de division correspondant. La fréquence la plus basse, 500 kHz, est délivrée par D32 (16 MHz divisés par 32). À noter que les NOP sont réparties de façon à obtenir le rapport cyclique (impulsion/pause) le plus égal possible ; un signal rectangulaire symétrique ne comporte, intrinsèquement, que très peu d'harmoniques.

On a le choix entre un total de 22 fréquences P.O. différentes. Les fréquences proches de la limite inférieure du spectre, 500 kHz, 516 kHz, 533 kHz, etc. sont plus rapprochées que celles situées à la limite supérieure.

Mise en œuvre

Une fois dans la boucle RF, tout retour en arrière est exclu. On pourrait imaginer une solution à base d'interruption. La procédure d'exploitation ci-après (**Listing 2**) est plus simple : on commence, à l'aide du potentiomètre, par choisir une fréquence adéquate, qui s'affiche à l'écran. On appuie ensuite sur S1, ce qui met l'émetteur en marche. Dans le même temps, LED2 s'allume pour signaler l'entrée en fonction. L'écran continue d'afficher la dernière fréquence sélectionnée. Une action sur le potentiomètre reste sans effet, la fréquence est stable. C'est exactement comme avec des émetteurs de puissance à tubes d'antan, où il n'était pas non plus possible de modifier la fréquence en cours de fonctionnement. Si l'on veut quand même changer de fréquence, il faut appuyer sur le bouton *reset* de la carte Arduino, ce qui arrête l'émet-

Listage 2. Choix de la fréquence d'émission.

```

Do
  U = Getadc(3)
  D = U / 46
  D = D + 10
  F = 16000 / D
  Locate 1 , 1
  Lcd F
  Lcd " kHz "
  Waitms 500
  If S1 = 0 Then
    Led2 = 1
    If D = 10 Then Goto D10
    If D = 11 Then Goto D11
    If D = 12 Then Goto D12
    If D = 13 Then Goto D13
    If D = 14 Then Goto D14
    If D = 15 Then Goto D15
    If D = 16 Then Goto D16
    If D = 17 Then Goto D17
    If D = 18 Then Goto D18
    If D = 19 Then Goto D19
    If D = 20 Then Goto D20
    If D = 21 Then Goto D21
    If D = 22 Then Goto D22
    If D = 23 Then Goto D23
    If D = 24 Then Goto D24
    If D = 25 Then Goto D25
    If D = 26 Then Goto D26
    If D = 27 Then Goto D27
    If D = 28 Then Goto D28
    If D = 29 Then Goto D29
    If D = 30 Then Goto D30
    If D = 31 Then Goto D31
    If D = 32 Then Goto D32
  End If
Loop

```

teur. On peut alors choisir une nouvelle fréquence et l'activer par S1. Le bouton S2 présent sur le *shield* n'a aucune fonction ici. ◀

(140430 - version française : Guy Raedersdorf)

Liens

[1] www.elektor-magazine.fr/140009

[2] www.elektor-magazine.fr/140430

convertisseur USB ↔ série multiprotocole avec FT232H

Ton Giesberts et Clemens Valens (Elektor Labs)

Dans un projet, il arrive souvent qu'un ordinateur ait à communiquer avec le port série d'un microcontrôleur ou avec un dispositif doté d'un port série RS-232, RS-485, I²C ou encore SPI. Comment faire lorsqu'on ne possède que de connecteurs USB ?

Elektor développe depuis longtemps des solutions, projets et palliatifs pour le port série traditionnel, tous basés sur des puces de conversion USB-série (voir les encadrés *Qu'est-ce qu'un port série ?* et *Une longue histoire*). Mais qu'en est-il des autres types de ports série ?

Les électroniciens ont de plus en plus souvent affaire à des composants dotés d'interfaces I²C et SPI. Puisque ces interfaces sont également des ports série, il semble logique qu'elles aient elles aussi leurs convertisseurs USB. Il existe bien pour cela notre passerelle USB/série BoB-FT232R, mais elle exécute le protocole série de façon logicielle et ce *bit-banging*, pour reprendre le terme anglais, est hélas plus lent qu'une implantation matérielle. Nous avons donc exploré d'autres voies possibles.

FT232H

La puce FT232H nous a semblé la meilleure piste à suivre. Le « H » de sa référence évoque la FT232R, mais ces deux puces (compatibles USB 2.0) sont très différentes. D'abord par leur débit, 12 Mbit/s pour la version « R » *USB Full Speed*, contre 480 Mbit/s pour la version « H » *USB High Speed*. Ensuite parce qu'en matière de conversion la « R » est décrite comme une USB/UART (avec quelques options supplémentaires d'interface en mode *bit-bang*), tandis que la

« H » est décrite comme une puce *USB haute vitesse vers UART/FIFO à usages multiples*. La « H » n'est donc pas uniquement UART, elle prend aussi en charge les interfaces I²C ou SPI ainsi que les ports parallèles. Toutes ces caractéristiques reposent sur le moteur appelé *Multi-Protocol Synchronous Serial Engine* (MPSSE).

MPSSE

D'après la documentation FTDI, le moteur MPSSE permet d'interfacer de façon flexible des dispositifs série synchrones avec un port USB. Notez le terme synchrone. Les protocoles de communication série synchrone utilisent au moins deux fils (un bus), un pour les données, et un pour le signal d'horloge. Dans un système synchrone, un dispositif fournit l'horloge, généralement le maître ; ce signal d'horloge synchronise l'émission et la réception des bits de données échangés entre les dispositifs reliés au bus. Une synchronisation très précise est alors inutile puisque chaque participant du bus « regarde » la même horloge. Dans les communications asynchrones, le signal d'horloge est au contraire combiné aux données et un seul fil suffit. L'inconvénient est que l'extraction de la donnée d'horloge parmi les autres données requiert une synchronisation précise des participants du bus. C'est en quelque sorte comme si les participants d'un système synchrone

utilisaient l'horloge du village, tandis que ceux d'un système asynchrone disposaient chacun de leur propre montre. Les interfaces SPI, I²C et JTAG reposent sur des communications synchrones. Le port série d'un microcontrôleur et son UART recourent à des communications asynchrones.

Comme le MPSSE est multiprotocole, la puce permet de communiquer avec différents types de dispositifs synchrones ; les protocoles les plus courants sont SPI, I²C et JTAG. Le formatage des données et la synchronisation peuvent être configurés de plusieurs façons afin de répondre à la plupart des exigences, pour un débit maximal de 30 Mbit/s. Le MPSSE peut aussi établir des communications série asynchrones, mais selon un seul protocole.

Le module MPSSE de la puce FT232H est également capable d'établir des communications synchrones parallèles de plusieurs manières. L'une est le mode FT1248, une interface utilisant un bus directionnel de 1, 2, 4 ou 8 bits de largeur. *A priori*, le MPSSE peut aussi émuler le célèbre (célèbre disent certains) bus multiplexé (adresses et données) des 8048 et 8051, mais il est difficile de trouver des informations sur ce mode d'utilisation. Le MPSSE permet aussi d'exploiter les modes FIFO synchrones et asyn-

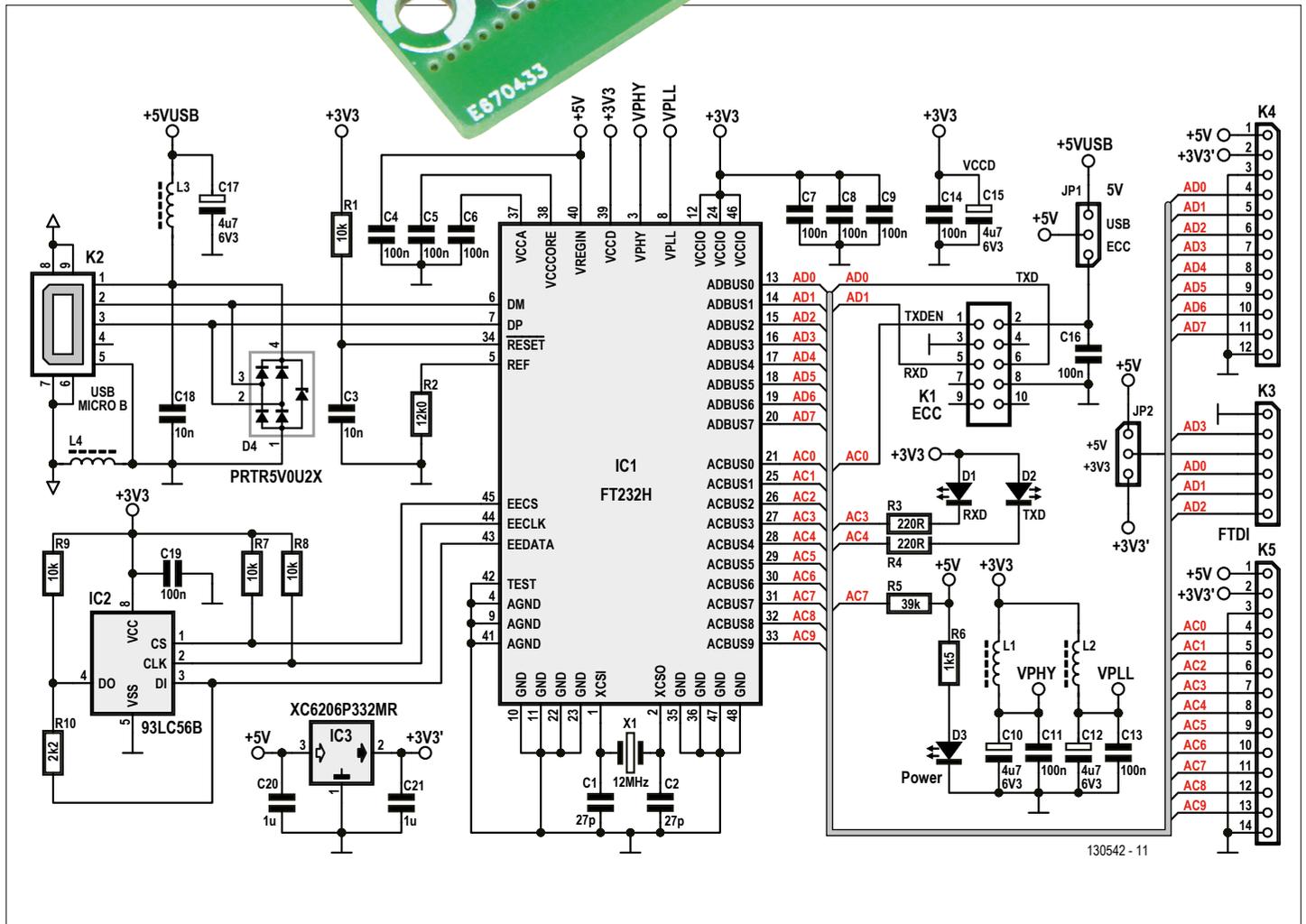
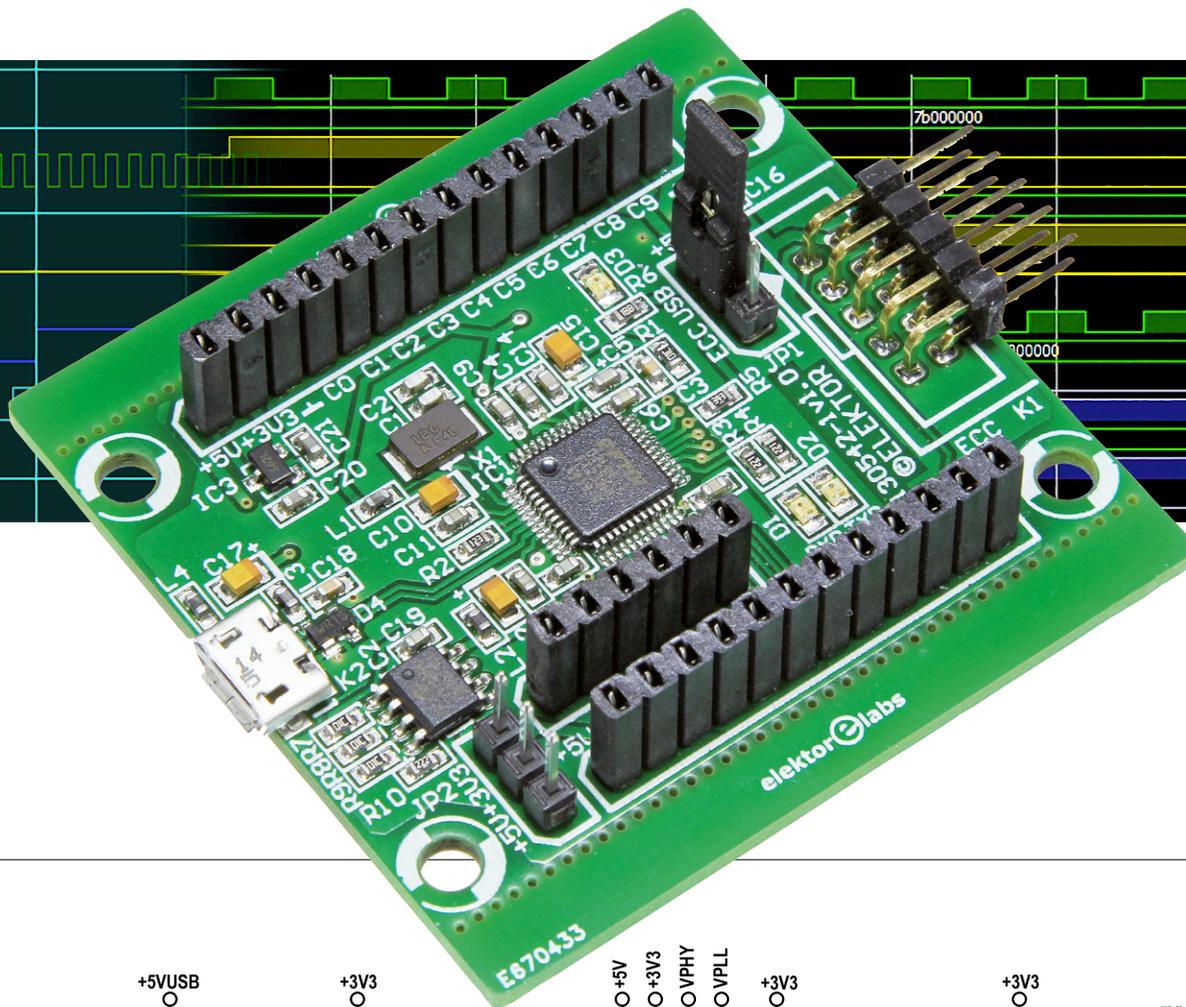


Figure 1. Schéma de la carte FT232H.

FTDI à tout prix ?

Vous l'avez sans doute remarqué, cet article ne cite que des puces de FTDI. Il est vrai qu'au labo Elektor nous utilisons souvent des convertisseurs USB-série de FTDI, mais FTDI n'est pas le seul fabricant de semi-conducteurs à vendre des puces à interface USB. En voici quelques-uns :

- Prolific et son célèbre PL2303.
- Cypress, qui a toujours été très actif dans le domaine de l'USB. Citons sa célèbre puce FX2LP, ainsi que la CY7C65211, similaire à la FT232H sinon qu'elle a un mode

chrones de la FT245 (une puce USB vers FIFO parallèle de FTDI).

Outre les protocoles série et parallèle, des signaux GPIO supplémentaires sont disponibles et le *bit-banging* sans contrainte de standard est aussi supporté.

Le MPSSE est actuellement implanté dans quatre puces : FT2232D, FT2232H, FT4232H (le premier chiffre indique le nombre d'UART) et FT232H (1 UART). Le MPSSE de la FT232H est celui qui prend en charge le plus de protocoles.

Le circuit

Si nous avons choisi la FT232H pour construire notre convertisseur à usages

multiples, c'est aussi parce qu'elle offre plus d'options que les autres puces de la même famille. Faites abstraction des bobines et condensateurs de filtrage et de découplage du schéma (**fig. 1**), vous verrez que le circuit est très simple.

Puisque nous voulions réaliser une sorte de carte de liaison (BoB, *break-out board*), les broches potentiellement intéressantes sont reliées aux connecteurs K4 et K5. Nous avons ajouté deux autres connecteurs, K1 et K3.

K1 est un ECC (*Embedded Communication Connector*), un connecteur Elektor utilisé sur certaines de nos cartes à micro-contrôleur. K1 peut alimenter la carte afin

pleine vitesse.

- Texas Instruments, le plus important des fabricants de semi-conducteurs. Lui aussi propose quelques convertisseurs, p. ex. le TUSB3410 à contrôleur 8052.
- Le chinois Nanjing QinHeng Electronics fabrique notamment la CH340, qu'on retrouve dans des programmeurs d'EEPROM très bon marché.
- Microchip propose de son côté le convertisseur USB-série MCP2200, en fait un PIC18F14K50 exécutant une pile USB.

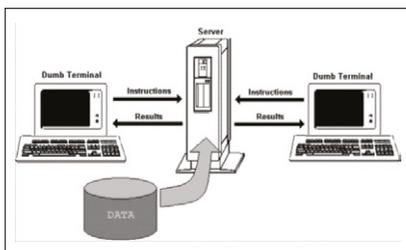
de ne pas surcharger le port USB. Pour cela, il faut court-circuiter les broches 1 et 2 du cavalier JP1.

K3 est compatible avec les câbles FTDI, donc vous pouvez utiliser la carte FT232H à leur place. Notez bien que tous les signaux de K3 (et par ailleurs ceux de tous les connecteurs) sont de 3,3 V, pas de 5 V. Cela dit les broches de la puce tolèrent le 5 V, donc la carte peut être utilisée sans crainte avec un système à 5 V. Le cavalier JP2 permet de sélectionner la tension sur la broche VCC du « câble » FTDI. Cette tension est toujours de 5 V

Qu'est-ce qu'un port série ?

Aux débuts de l'informatique, lorsque les ordinateurs étaient encore chers et volumineux, une façon de réaliser des économies était d'acheter un ordinateur central, de le placer en sous-sol et d'y relier plusieurs terminaux. Ce terminal n'était alors rien d'autre qu'un clavier et un écran, d'où le terme de terminal passif (*dumb terminal*) pour désigner ce type d'équipement de saisie. Cette façon de procéder par liaison physique distante avec l'unité centrale a finalement donné naissance à la norme RS-232.

La norme RS-232 définit une interface de liaison sur au plus 25 points, dont un fil d'envoi des données (TXD) et un fil de réception des données (RXD). Près d'un demi-siècle plus tard, ce type d'interface a quasiment disparu et seule subsiste



aujourd'hui une interface à trois broches pour les signaux RXD, TXD et GND. Les niveaux de tension d'origine ont été remplacés par 5 V (TTL) ou moins. C'est cette incompatibilité de niveaux qui explique que vous ne pouvez pas relier directement un microcontrôleur à un vrai port RS-232. Pourquoi l'appelle-t-on interface *série* ? Rappelons d'abord qu'à l'intérieur d'un ordinateur les données sont représentées

par des bits. Un bit ne peut prendre que deux valeurs, 0 ou 1, et ne peut donc représenter, ou coder, que deux objets. Pour représenter de façon unique plusieurs objets, par exemple les 26 lettres de l'alphabet, il est donc nécessaire d'utiliser des séquences de bits. L'informatique a privilégié la représentation des données par séquences de 8 bits, ou octets.

Matériellement, ces octets sont transmis via un port d'une « largeur » de 8 fils (en oubliant le fil de masse). Un tel port est dit parallèle puisqu'il transmet les bits en parallèle. Envoyer un octet au moyen d'un seul fil est huit fois moins rapide et oblige à transmettre les bits un par un, autrement dit en série. Et voilà expliqué le « série » de notre interface RS-232.

De l'eau a coulé sous les ponts depuis l'apparition du premier port série et, comme cela arrive à chaque fois que des ingénieurs s'intéressent à une technique, de nombreuses versions des ports série ont vu le jour. Bien qu'il puisse aujourd'hui être associé à plusieurs protocoles (SPI, I²C, USB, JTAG, I²S, Ethernet, SATI, SWD...), le terme *port série* désigne toujours un port transmettant l'information bit par bit.



sur un vrai câble FTDI, mais dans le cas des systèmes à 3,3 V il est souvent plus pratique (et plus sûr) d'avoir 3,3 V sur cette broche. Le régulateur de tension IC3 délivre 3,3 V à K3, ce qui évite de charger inutilement le régulateur interne d'IC1. La carte est équipée de trois LED témoin. La LED D3 s'allume si la carte est alimentée. JP1 joue un rôle important ici.

Les LED D1 et D2 s'allument lorsque les lignes RX et TX sont actives et que la carte est utilisée comme convertisseur USB-série « normal ». Dans les autres modes, elles n'ont pas de fonction prédéfinie.

Une EEPROM externe est nécessaire si l'on veut exploiter toute la puissance de la FT232H. Parmi les paramètres de configuration de l'EEPROM figurent les codes USB VID et PID (vendeur et produit), la source d'alimentation (auto-alimentation ou par bus) ainsi que la fonction des broches ACBUS. La fiche technique décrit tous les paramètres de configuration, consultez-la au besoin. Quelques mots sur le mode *USB Suspend*, un état de basse consommation dans lequel le bus ne tire pas de courant du port USB. Si ce mode est activé, la puce y entre par mise au niveau bas de la broche ACBUS7. La configuration par défaut de la broche ACBUS7 dans l'EEPROM est une entrée avec résistance de rappel vers le bas ; si cette broche n'est pas connectée, la puce s'endormira. La résistance R5 relie ACBUS7 au +5 V pour éviter que cela n'arrive automatiquement. La valeur de R5 est donnée dans la fiche technique.

La FT232H peut fonctionner sans EEPROM, mais au prix de certaines limitations. Sinon ce sont les valeurs par défaut de l'EEPROM qui sont utilisées. L'utilitaire FT PROG décrit ci-après permet de modifier ces valeurs.

La résistance R2 (1 %) et le quartz X1 (± 30 ppm) sont des « pièces de précision » indispensables au fonctionnement d'IC1. Les condensateurs C1 et C2 garantissent une oscillation correcte de X1. Leurs valeurs dépendent du quartz utilisé, mais conviendront dans la plupart des cas.

K2 est un connecteur micro-USB, vous pourrez donc éventuellement employer le câble de votre chargeur de téléphone si celui-ci est détachable. La diode D4 protège les entrées de la puce contre les

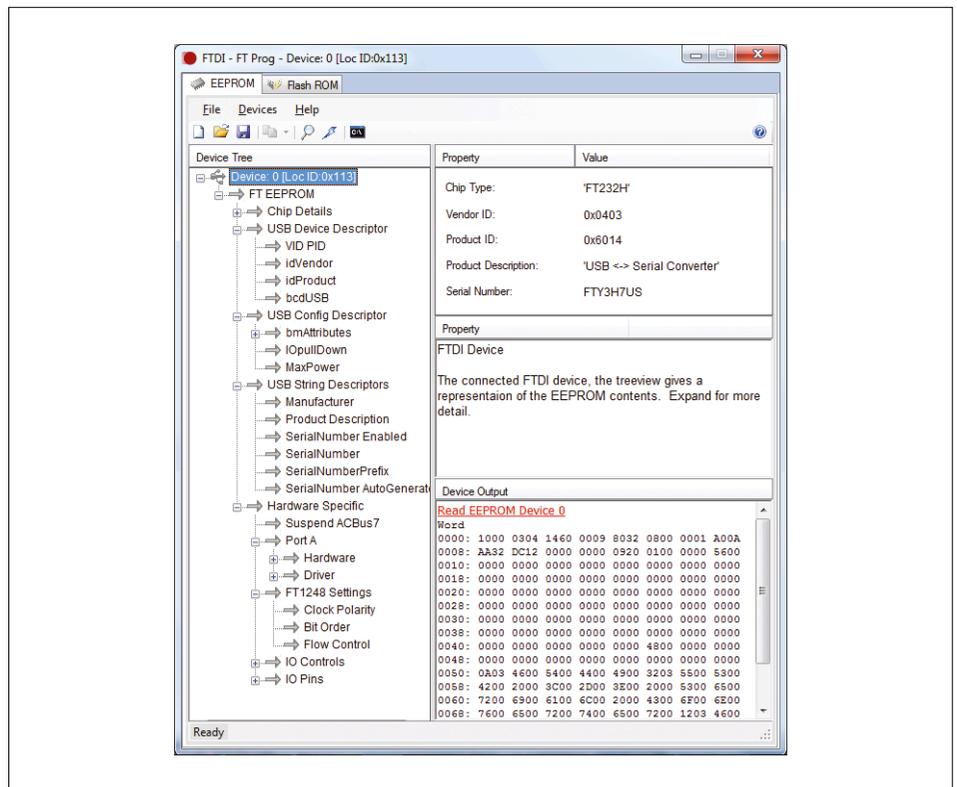


Figure 2. Détection de notre FT232H par FT PROG.

bruits et surtensions potentiellement dangereux. Les composants L3, L4 et C18 protègent contre les bruits de haute fréquence des câbles d'alimentation et de blindage.

FT PROG

L'utilitaire FT PROG (disponible sur le site de FTDI) permet de configurer la FT232H

(IC1) et de programmer et lire l'EEPROM (IC2). Reliez la carte à un PC, lancez FT PROG, et cliquez sur *Scan and Parse* (icône en forme de loupe). Vous devriez obtenir l'écran de la **figure 2**. On peut y activer/désactiver certaines fonctions, et configurer le pilote USB ainsi que les broches et interfaces de la partie asynchrone de la puce.

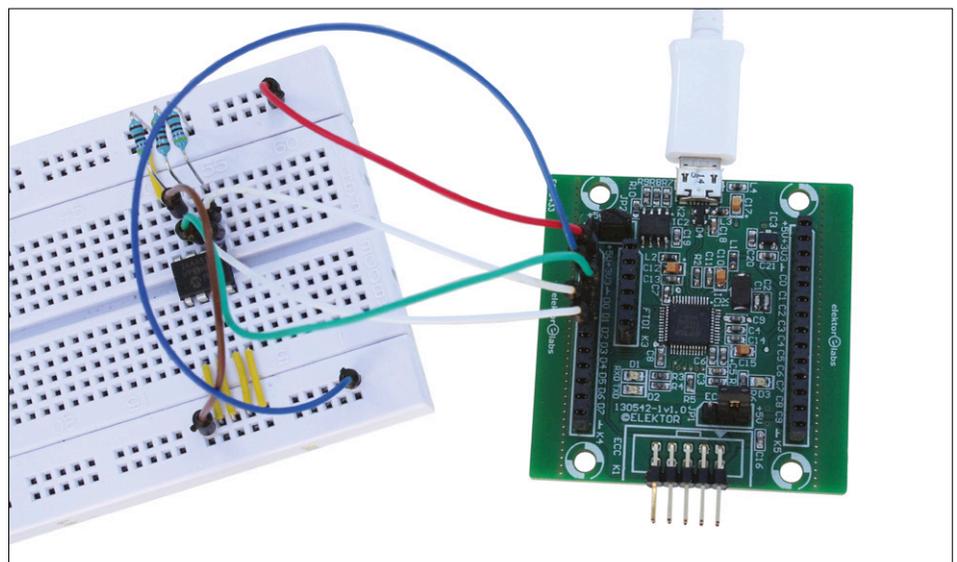


Figure 3. La carte FT232H reliée à une EEPROM I²C montée sur une plaque d'essai. Notez les deux fils blancs reliant AD1 et AD2 à la broche SDA.

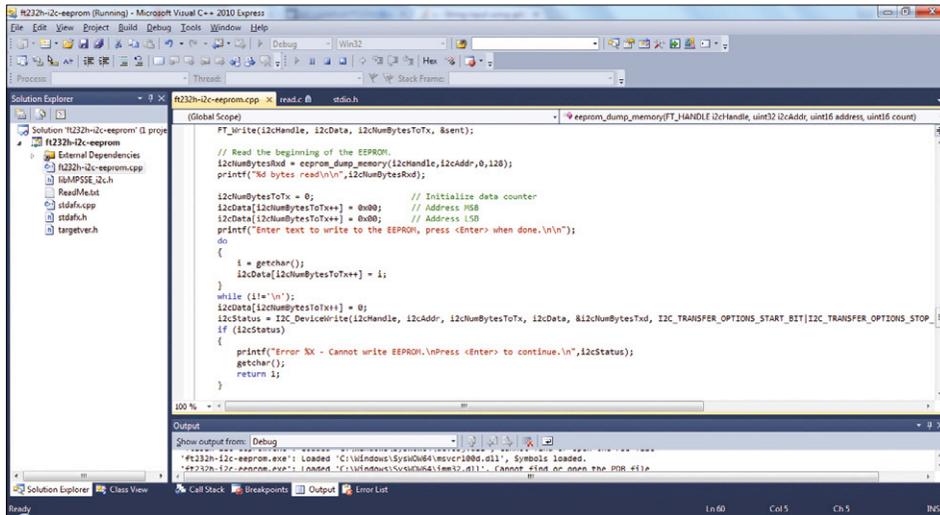


Figure 4. Code montrant comment écrire des données sur l'EEPROM I2C de la plaque d'essai.

Tableau 1. Cavalier JP1	
Court-circuit	Fonction
1-2	alimentation par K1 (ECC)
2-3	alimentation par USB

Tableau 2. Cavalier JP2	
Court-circuit	Fonction
1-2	5 V sur la broche 3 de K3
2-3	3,3 V sur la broche 3 de K3

Liste des composants

Résistances

Par défaut : CMS 0603, 0,1 W
 R1, R7, R8, R9 = 10 kΩ
 R2 = 12 kΩ 1 %
 R3, R4 = 220 Ω
 R5 = 39 kΩ
 R6 = 1,5 kΩ
 R10 = 2,2 kΩ

Condensateurs

Par défaut : CMS 0603
 C1, C2 = 27 pF, 50 V, C0G/NP0
 C3, C18 = 10 nF, 50 V, X7R
 C4-C9, C11, C13, C14, C16, C19 = 100 nF, 50 V, X7R
 C10, C12, C15, C17 = 4,7 μF, 6,3 V, tantale, CMS boîtier R (0805)
 C20, C21 = 1 μF, 6,3 V, X5R

Inductances

L1-L4 = perle de ferrite, 600 Ω @100MHz, 25 %, 0,15 Ω @CC, 1,3 A (CMS 0603)

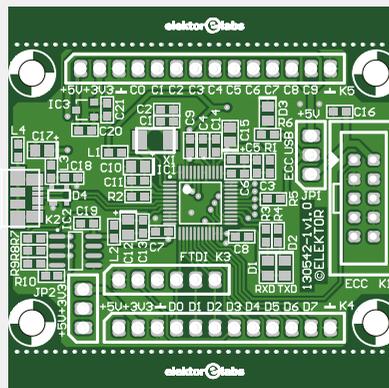
Semi-conducteurs

D1, D2, D3 = LED, verte (CMS 0805)
 D4 = PRTR5V0U2X (SOT-143B)
 IC1 = FT232HL (LQFP-48)
 IC2 = 93LC56B-I/SN (SOIC-8)
 IC3 = XC6206P332MR (SOT-23)

Divers

K1 = barrette DIL à 10 broches (2x5), droites
 K2 = micro-USB embase femelle type B, CMS
 K3 = embase SIL à 6 contacts, pas de 2,54 mm
 K4 = embase SIL à 12 contacts, pas de 2,54 mm
 K5 = embase SIL à 14 contacts, pas de 2,54 mm
 JP1, JP2 = barrette à 3 broches, pas de 2,54 mm
 JP1, JP2 = cavalier, au pas de 2,54 mm
 X1 = quartz 12 MHz, 18pF CL, CMS 5 x 3,2 mm circuit imprimé nu v1.0, e-choppe 130542-1

boîtier Hammond Manufacturing, type 1455C802
 carte partiellement assemblée, e-choppe 130542-91



FT PROG ne sert pas à contrôler le MPSSSE ; activer et utiliser p. ex. le port I2C ou SPI se fait de façon logicielle.

L'UART se paramètre depuis *Hardware Specific*. Le Port A correspond aux lignes identifiées par ADBUSx et ADx sur notre schéma. Vous pouvez sélectionner l'interface à utiliser sur ce port (UART par défaut, le meilleur choix sauf besoin spécifique) ainsi que le pilote que chargera le SE. Le pilote VCP (choix par défaut) vaut pour les applications employant un port traditionnel ou historique.

Le Port C du sous-menu *IO Controls* correspond aux lignes notées ACBUSx et ACx sur notre schéma. Ce port commande les LED et certains signaux comme TXDEN (pour les interfaces RS-485 et RS-422). Soyez attentif ici, les dix broches ne sont pas identiques, certaines offrent plus de possibilités que d'autres.

Au risque d'embrouiller un peu plus le tableau, on retrouve les ports A et C sous *IO pins*, mais maintenant appelés groupes AD et AC. Ce sous-menu permet de sélectionner le taux de montée (*slew rate*), le type d'entrée et le courant des broches concernées.

Une fois la configuration terminée, l'EEPROM se programme en cliquant sur *Program Devices* (icône en forme d'éclair). Pour reproduire nos essais avec le port série, synchrones ou asynchrones, tous les paramètres peuvent être laissés sur leurs valeurs par défaut.

Programmes d'exemples

Comme le laisse entrevoir ce qui précède, la relative complexité de la FT232H permet de l'utiliser de nombreuses façons. Pour aider l'utilisateur à en tirer le maximum, FTDI a publié plusieurs notes d'application [1] ainsi que des programmes d'exemple. Les notes les plus intéressantes sont les 180, 188 et 190. Ces documents se réfèrent au matériel FTDI, mais peuvent être utilisés directement avec notre carte. Tous les exemples reposent sur les pilotes D2XX de FTDI.

La note 180 contient un code remarquable en Visual Basic (2008). Il montre comment relever des tensions et intensités avec deux convertisseurs analogique-numérique MCP3201, une interface SPI et quelques broches d'E/S à usage général (GPIO).



Figure 6. La carte FT232H se glisse dans un boîtier Hammond en laissant ressortir K1.

Boîtier

L'utilité de la carte FT232H est indéniable, nous l'avons donc dimensionnée pour qu'elle puisse tenir dans un boîtier standard (**fig. 6**). Le boîtier Hammond en

aluminium que nous avons choisi est un peu long, mais il est facile de le couper à la bonne longueur. Vous pouvez aussi découper la carte FT232H selon les pointillés. Glissez-la dans le boîtier, vous aurez

entre les mains un robuste convertisseur USB-série. ◀

(130542 - version française : Hervé Moreau)

Liens

- [1] FT232H : www.ftdichip.com/Products/ICs/FT232H.htm
- [2] Carte Elektor DSP audio universelle à ADAU170 : www.elektor-magazine.fr/130232
- [3] Bibliothèque Python GPIO d'Adafruit : <https://learn.adafruit.com/adafruit-ft232h-breakout/>
- [4] Ces câbles sont toujours disponibles dans l'e-choppe : 080213-71 (version 5 V) ; 080213-72 (version 3,3 V).
- [5] Platine RS-232, réf. e-choppe 080470-1.
- [6] Passerelle USB/série BoB-FT232R, Elektor septembre 2011, www.elektor-magazine.fr/110553. Réf. e-choppe 110553-91.
- [7] Convertisseur USB/UART RS-232/422/485 & concentrateur USB, Elektor novembre 2014, www.elektor-magazine.fr/140033. Réf. e-choppe 140033-91.
- [8] www.elektor-magazine.fr/130542

**Jaime González-Arintero**

jaime.glez.arintero@eimworld.com

Retour de balancier

Ici, le partage et l'échange sont au coeur de nos préoccupations. Dans beaucoup de circuits intégrés, et plus généralement dans tout circuit d'une certaine complexité – le signal du partage par excellence est celui de l'horloge. Un signal *Clock* ou CLK (= hor-

loge) gouverne l'ensemble. La survenue de tout événement est assujettie à cette horloge. Cette règle implicite nous semble évidente, à tel point que le fait de se passer d'horloge est considéré comme une pratique déviante en électronique.

Et si l'usage des horloges constituait un obstacle au progrès ?

Tentons une analogie

Imaginez une grande entreprise dont les employés prennent et quittent leur travail à la même heure et sont parfaitement synchronisés. Certains se montreront plus rapides ou plus habiles que d'autres dans l'exécution de leur tâche, ils seront néanmoins obligés de ralentir et/ou de patienter jusqu'à ce que l'équipe entière ait accompli sa tâche. Certaines équipes devront en attendre d'autres avant que le produit final soit prêt.

Imaginez maintenant une jeune pousse formée d'une équipe réduite dont la production consiste en flux de travaux individuels ou collectifs, sans synchronisation stricte. En fin de compte, les bénéfices de la grande entreprise n'excèderaient pas nécessairement les gains réalisés par la jeune pousse. De ce point de vue, un μC est un peu une grande entreprise...

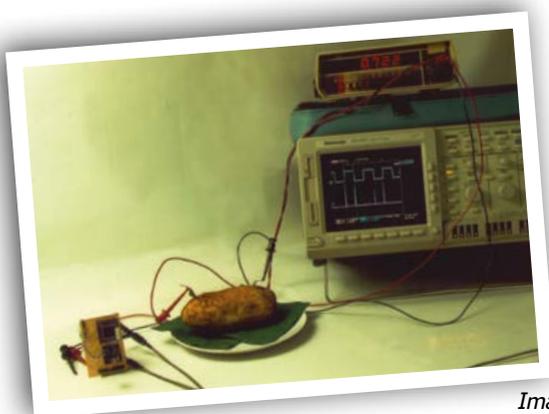
Si tout fonctionne, pourquoi se passer d'horloge ? Parce que les microcontrôleurs asynchrones présentent certains avantages par rapport à leurs homologues synchrones. Leur consommation est très faible, leur bruit électromagnétique infime, et ils peuvent être à la fois plus rapides et plus efficaces.

De plus les périphériques d'un dispositif asynchrone ne consomment presque pas de courant lorsqu'ils ne sont pas sollicités, et il en va de même du processeur en veille. Une puce sans horloge est également susceptible d'accélérer ou de ralentir en fonction des conditions extérieures, la température ambiante par exemple.

Ces avantages ne doivent pas être dédaignés, certains projets à la pointe de la technologie adoptent une approche sans horloge. Les puces neuromorphiques du projet SyNAPSE de DARPA (*Defense Advanced Research Projects Agency*) comportent un nombre record de transistors et fonctionnent en mode asynchrone.

Alors, surcadencage, sous-cadencage, ou nada-cadencage ? Certains concepteurs pensent que l'avenir privilégiera l'absence totale d'horloge, d'autres que la plupart des logiciels de conception ne sont pas prévus pour faire de la logique asynchrone et que le manque de formation des électroniciens dans ce domaine allongerait beaucoup trop les temps de conception. Dans le monde universitaire, si l'on excepte quelques groupes de recherche, le sujet est à la limite de l'ésotérisme.

Quant aux interfaces entre puces synchrones et asynchrones, c'est comme la politique, mieux vaut éviter le sujet !



Une puce qui a la patate

Chacun sait que les puces dépourvues d'horloge peuvent fonctionner plus rapidement et plus efficacement que les puces synchrones. En 1989, Mika Nyström, du groupe Asynchronous VLSI du Caltech, a montré que le CAM (*Caltech Asynchronous Microprocessor*), le tout premier microcontrôleur asynchrone, pouvait fonctionner cadencé à environ 300 kHz avec une alimentation pour le moins originale : deux pommes de terre reliées en série (créant une DDP d'environ 0,9 V).

Image: Asynchronous VLSI Group, Caltech

restauration d'un voltmètre à tubes (2)

Chuck Hansen (USA)

Voici la suite de la 1^{ère} partie publiée dans le numéro de mars 2015. Le texte fera référence au schéma du 400H, également reproduit ici.

Que le test commence !

J'ai remplacé V9, le tube régulateur 5651A, par un neuf et mis sous tension le 400H graduellement en faisant monter la tension secteur avec mon alimentation AC variable B-K 1655 sur une période de 15 mn. Le bus régulé B+ était maintenant à +245 VDC, mais les tensions

d'écran et de plaque des quatre tubes amplificateurs V2-V5 étaient 15 à 37 % trop hautes.

J'ai ensuite remarqué que la tension redressée des filaments des tubes V1-V4 était un peu basse à 10,5 VDC. R66 sert à ajuster la tension de filament vers les 12,6 V spécifiés pour une tension secteur de 115 VAC, mais, tournée à fond, ne permettait d'atteindre que 11,5 VDC. Les enroulements correspondants produisaient la bonne tension de 13,6 VAC.

Avant que CR3 ne tombe en panne en émettant un nuage toxique et nauséabond, je le remplacerai par un pont redresseur double-alternance au silicium. Il faudra ajouter une résistance en série : la tension de chute des diodes au silicium est habituellement plus faible que celle du sélénium, surtout quand ce dernier a vieilli. Ensuite je pourrai ajuster la polarisation de l'amplificateur avec R119 pour tenter d'obtenir les bonnes tensions sur ces tubes.

Par précaution, je remplacerai C30, le quadruple 20 μ F 450 VDC, par un quadruple 20 μ F 475 VDC (AES).

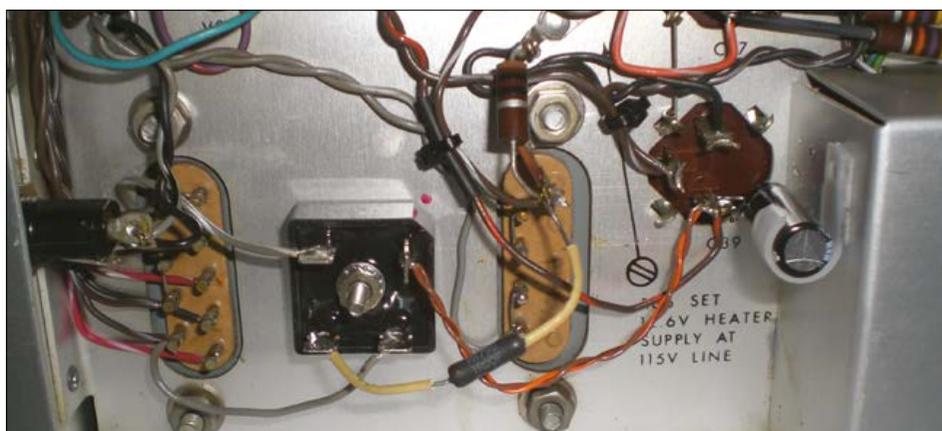
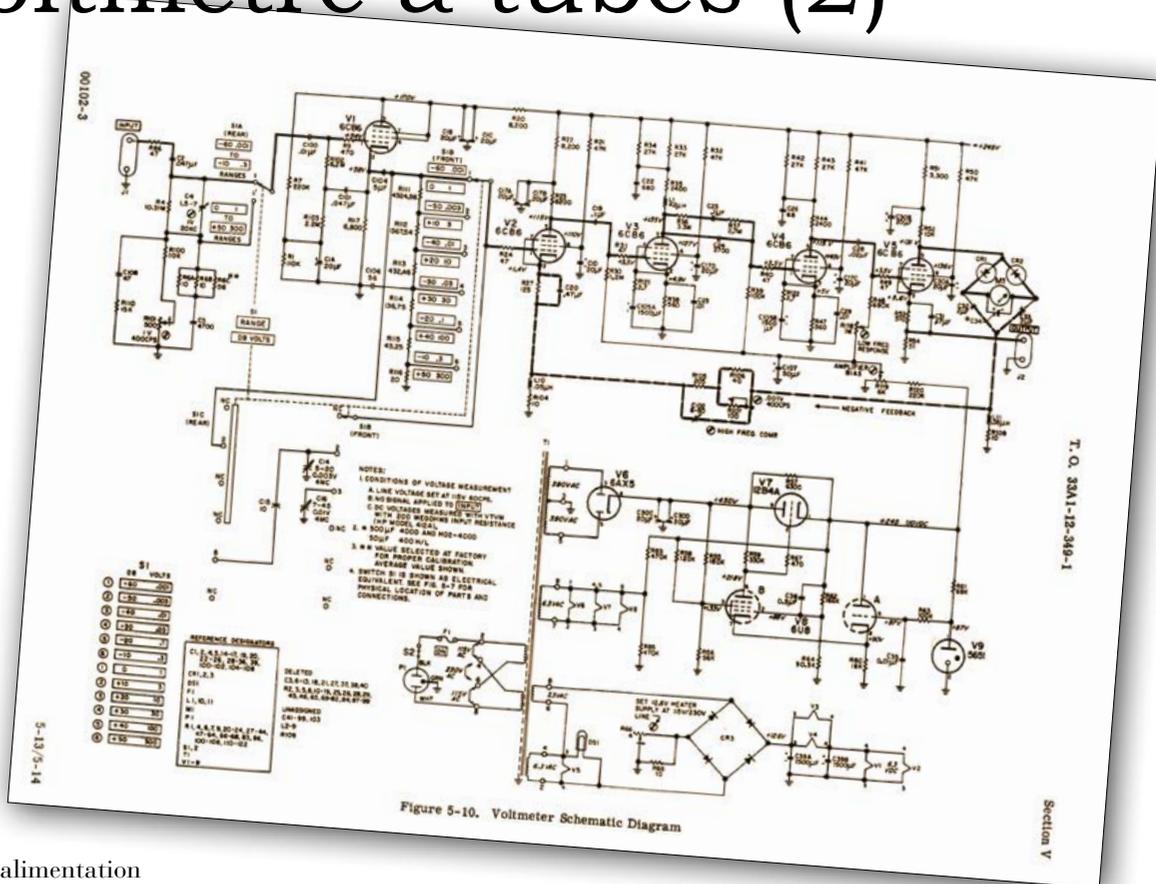


Figure 7. CR3 remplacé par un pont au silicium, la résistance de 1 Ω ajoutée et le condensateur 105°C, 1500 μ F en parallèle avec C39C (sur le blindage de V1-V2 à droite).

EST[®] 2004

www.elektor.tv



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

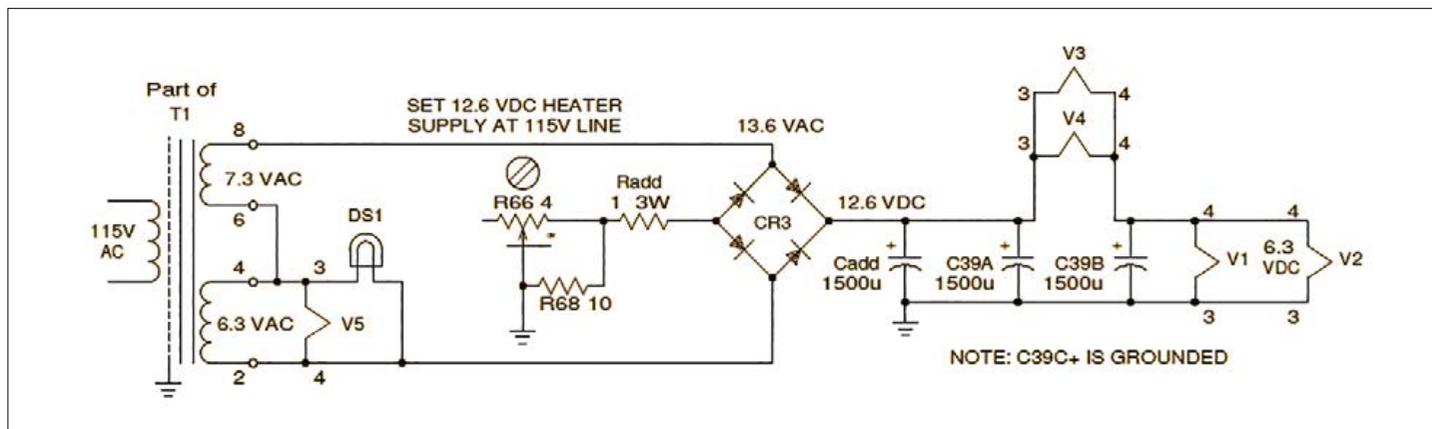


Figure 8. Schéma de l'alimentation DC pour filaments modifiée.

C'est C30 qui voit les tensions B+ les plus élevées et sûrement celui qui a le plus souffert lors de la panne du tube régulateur.

Réparation du HP-400H et autres tests

Après avoir remplacé CR3 par un pont redresseur 50 V 15 A au silicium, j'ai mesuré la tension de chute à presque 2,5 V. J'ai ajouté une boîte à décade de résistances de puissance en série avec le négatif du pont. Avec le HP-400H remis sous tension, mon oscillo montrait une très forte ondulation sur la tension continue de filament. C39 semble endommagé, mais AES ne propose pas d'équivalent dans son catalogue en ligne (tubesandmore.com). J'ai ajouté un condensateur 1500 µF discret en parallèle avec le condensateur de filtrage de la tension de filament C39A. Je l'ai fixé avec un passe-câble plastique modifié sur la gauche du blindage de V1-V2 (fig. 7). J'ai également dû ajuster la résistance de la boîte à décade sur 1 Ω

pour réduire la tension de filament à 12,6 VDC avec R66 en position médiane. J'ai ensuite remplacé la boîte à décade par une résistance bobinée Dale RS2B, 1 Ω 3 W.

J'ai dû rallonger un peu les fils gris et gris/blanc en provenance des enroulements pour filament du transformateur pour pouvoir atteindre l'entrée AC du pont redresseur. Puis, j'ai dû joindre le fil marron/orange qui allait du positif du redresseur au sélénium au fil du filament pour compléter ce circuit. Sans fil marron/orange pour relier le + du pont à C39A, j'ai torsadé du fil marron AWG-22 avec de l'orange AWG-24 Teflon® pour conserver les couleurs.

Le schéma de mon alimentation pour filaments DC modifiée se trouve en **figure 8**.

Avec l'entrée du HP 400H court-circuitée, l'aiguille dévie complètement sur les deux plus bas calibres : 0,001 V et 0,003 V. Elle devrait être à une petite division du zéro sur le calibre 0,001 V. Il y a

deux condensateurs aluminium discrets Sprague 50 µF, 6 V ; C34 en parallèle avec le galva et C107 pour la polarisation de l'amplificateur. J'ai commandé deux condensateurs 47 µF, 16 V Vishay spécifié pour 105 °C chez Mouser.

D'abord il fallait résoudre le problème de la haute tension résiduelle qui bloquait le galva sur la gamme 0,003 V. J'ai remplacé C34, la capa aux bornes du galva par une de celles commandées chez Mouser (fig. 9). Le galva affichait maintenant 0,18 mV sur la gamme 0,001 V avec l'entrée en court-circuit et le boîtier retiré. J'ai testé le vieux condo avec un LC-mètre Sencore LC102 : seulement 25 µF à 5,4 VDC. L'ESR était haute avec 15,7 Ω. L'absorption du diélectrique à 31%, avec plus de 700 µA de courante de fuite, tous deux déclarés MAUVAIS par le LC-mètre. La tension de polarisation de tous les tubes amplificateurs 6CB6 était trop haute, d'où le remplacement de C107 par un des condensateurs Vishay. Malheureusement, C107 est planqué sous un ensemble de

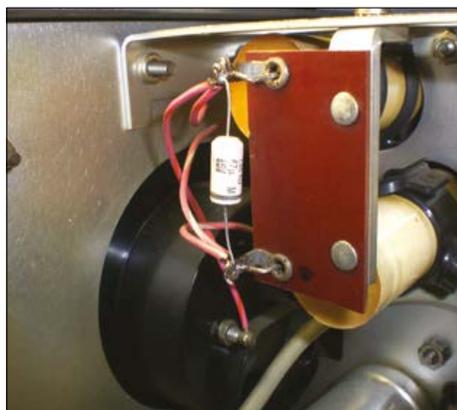


Figure 9. Le remplaçant de C34, la capa aux bornes du galva.



Figure 10. C107 (boîtier argenté) est caché sous des composants et un cavalier.



Figure 11. Le remplaçant de C107 monté au-dessus de C23.

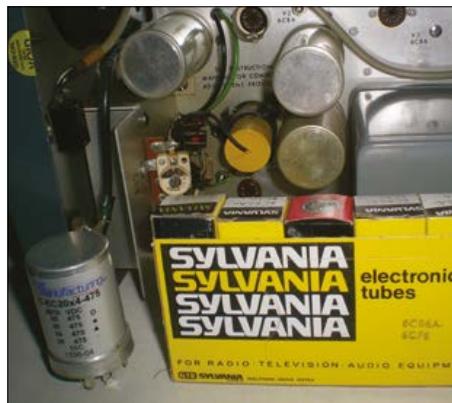


Tableau 1.
HP 400H Voltage Error Test

Calibre VTVM	Fréq.	V appliquée	V erreur
300	60 Hz	300	0%
		270	0%
		30	-3,3%
100	60 Hz	100	0%
		90	0%
		10	5,0%
30	60 Hz	30	0%
		27	0%
		3	0%
10	60 Hz	10	0,8%
		9	0,22%
		1	4,0%
3	1 kHz	3,0	0%
		2,7	0%
		0,3	1,3%
1	1 kHz	1,0	0,5%
		0,9	0,6%
		0,1	1,6%
0,3	1 kHz	0,30	2,3%
		0,27	2,5%
		0,03	4,3%
0,1	1 kHz	0,10	2,0%
		0,09	2,2%
		0,01	8,0%
0,03	1 kHz	0,030	1,7%
		0,027	2,2%
		0,003	23%
0,01	1 kHz	0,010	3,0%
		0,009	5,9%
		0,001	28%
0,003	1 kHz	0,0030	2,3%
		0,0027	4,1%
		(voir texte)	
0,001	1 kHz	(voir texte)	

Figure 12. Le condensateur C30 et les tubes 6CB6A, en feu.

Figure 13. Le C30 d'origine et composants environnants.

composants et un cavalier (fig. 10) ; comment réduire le nombre de composants à retirer pour le remplacer ?

Facile : couper le vieux C107 et installer le nouveau au-dessus du condensateur à film noir Sprague 160P C23 (fig. 11). J'ai vérifié le vieux condensateur : 51,6 μF , avec une ESR de 3,9 Ω . Le courant de fuite de 17 μA et l'absorption du diélectrique de 12% étaient toujours respectables. Tous deux étaient considérés comme BON par le Sencore LC102. Le nouveau condensateur Vishay faisait 45,1 μF avec une ESR de 3,7 Ω , bien en dessous de la limite max de 5,4 Ω .

Jouer sur R119 a corrigé la polarisation de grille sur tous les tubes. Cependant, la tension de plaque de V3 était maintenant de 154 VDC au lieu de 121 VDC. J'ai échangé V3 et V4 : V4 avait alors une tension de plaque de 152 V au lieu de 119. Comme au moins l'un des tubes était faible, j'ai commandé des pentodes 6CB6 pour V1-V5 chez Antique Electronic Supply.

En vérifiant les composants dans les circuits des amplificateurs, j'ai vu que quelqu'un d'autre avait déjà travaillé

sur ce VTVM. Il y avait plusieurs soudures sèches (du travail d'amateur). J'ai également trouvé les modifications suivantes :

1. C24 remplacé par un modèle à film polyester 0,0025 μF 600 VDC Sprague 220P Orange Drop au lieu d'être un Sangamo 1800 μF 500 VDC au mica, ou un 2700 μF au mica d'après mon schéma ultérieur.
2. R27 remplacée par un modèle bobiné Dale CS-2 125 Ω 3 W au lieu de 125 Ω 1/2 W Dale. Pas de mention de la série Dale CS dans mon catalogue Dale Electronics de 1991. Elle ressemble exactement à celles de la série RS.
3. C105, le condensateur qui avait fui sur le boîtier, remplacé par un condensateur Sprague identique avec un code de date plus récent de 1957. Tous les autres condensateurs dataient de 1952. Lorsque C105 a été remplacé, les pattes de R122 coupées trop courtes ont été rallongées avec des bouts de fil étamés.
4. Des marques marron sur le plastique de C30 montraient que lui aussi avait

Composants de remplacement utilisés sur le HP 400H

Achetés chez Antique Electronic Supply (tubesandmore.com) :

• C30	condensateur alu 20 μF x 4, 475 V	C-EC20X4-475
• V1-V5	pentodes	6CB6-A/6CF6
• V9	régulateur	5651

Achetés chez Mouser (mouser.com) :

• CR3	pont redresseur silicium 12 A, 100 V	512-GBPC1201
• C34, C107	condo alu 47 μF , 16 V, 105°C	594-2222-138-25479
• C ajouté	condo alu 1500 μF , 16 V, 105°C	667-EEU-FR1C152LB
• R ajouté	résistance bobinée 1 Ω , 3 W, 1%	71-RS2B-1.0



Figure 14. C30 retiré du boîtier.

souffert de la panne du tube régulateur V9. J'ai installé les cinq tubes 6CB5A de remplacement puis ai remplacé C30 par le modèle 475 V commandé chez AES. Voir les **figures 12 à 16**.

Les tensions de plaque sont maintenant correctes sauf pour V4 et V5 pour lesquels elles sont toujours un peu élevées. La permutation des tubes n'a pas produit d'effet notable sur les tensions. La résistance de 47 k Ω en série avec la grille de V5 mesure 52 k Ω .

Le galva indique maintenant 0,017 mV sur le calibre 0,001 V avec l'entrée court-circuitée et le boîtier fermé.

Vérification de la précision

Après ces réparations, j'ai calibré le HP 400H comme indiqué dans le manuel HP en utilisant mon multimètre Fluke 8050A (calibré avec une source AC Fluke 5200 en juin 2014) en parallèle avec le HP 400H. La source de signal est mon oscillateur de test de distorsion HP-339A.

D'abord j'ai testé la précision en fréquence en injectant un signal sinusoïdal de 1,000 V_{rms} entre 10 Hz et 50 kHz. J'ai réglé le HP 400H sur la gamme 1 V : 0,95 V à 10 Hz et 0,96 V à 20 Hz. La lecture était ensuite la même de 50 Hz à 20 kHz, et il donnait 1,005 V à 50 kHz, ce qui correspond à la fréquence max pour mon 8050A. Je n'ai pas les appareils nécessaires pour tester le HP 400H jusqu'à sa limite en fréquence de 4 MHz.

Ensuite, j'ai vérifié la précision en tension à 10%, 90% et 100% de la pleine échelle sur chacun des douze calibres de tension. J'ai utilisé mon oscillateur HP-339A réglé sur 1 kHz, en faisant grimper la tension de 2,7 mV_{rms} à 3 V_{rms}. Le plus faible niveau en sortie de mon HP-339A est 2,5 mV, je n'ai



Figure 15. Le remplaçant de C30 installé et les composants alentour remis en place.

donc pas pu tester la gamme 0,001 V du HP 400H. L'erreur en tension à 10% de la pleine échelle augmentait à mesure que le niveau du signal diminuait. Cependant, l'appareil respectait largement sa tolérance de $\pm 1\%$ à 90% et 100% de la pleine échelle jusqu'à la gamme 1 V. En dessous, c'est le niveau du bruit qui semblait affecter la précision de la mesure. Le **tableau 1** contient un résumé des mesures.

Le manuel de l'HP 400H conseille d'installer un adaptateur de prise sans terre pour briser les boucles de masse et améliorer la précision sur les faibles calibres. Je ne vous le recommande pas, surtout sur un appareil de plus de 50 ans dont on ne connaît pas la tension d'isolation du boîtier.

Le niveau max en sortie de l'oscillateur HP-339A est de 6,75 V_{rms}. Pour la gamme 10 V et au-delà, j'ai utilisé une alimentation AC 60 Hz variable entre 0 et 325 V. À 10% de la pleine échelle l'erreur était de moins de $\pm 5\%$. À 90% et 100% de la pleine échelle, le HP 400H s'en est beaucoup mieux sorti : le plus souvent il donnait la valeur exacte (tableau 1).



Figure 16. Le remplaçant de C30 installé, côté boîtier.

Pour finir

Il faut changer la poignée de cuir en piteux état mais AES ne propose plus que des modèles en caoutchouc moulé pour amplis de guitare. Il reste deux quadruple capas de 20 μ F, 450 V que je devrais probablement remplacer : C1 et C17.

Je serais très heureux si je parvenais à trouver de quoi remplacer les deux condensateurs basse tension 1500 μ F x 3, 15 VDC, C34 et C107. J'ai rencontré le même problème en rénovant mon ampli intégré Scott 222C (*audioXpress* mai à septembre 2006). AES avait à son catalogue des remplaçants pour les capas haute tension mais rien à proposer pour le quadruple 75 μ F, 75 V utilisé pour la polarisation de grille (C207-C210). J'ai dû joindre les capas internes par des condensateurs à l'aluminium discrets 100 μ F, 100 VDC 105°C.

Je suis plutôt satisfait des performances de cet HP 400H. C'est un appareil que j'ai trouvé intéressant à diagnostiquer et réparer. J'ai aussi aimé voir comment mes prédécesseurs avaient résolu les problèmes qu'ils avaient rencontrés. ◀

(150034 - version française : Kévin Petit)

L'auteur



Chuck Hansen, ingénieur en électronique, détenteur de cinq brevets, travaille comme consultant pour l'industrie aéronautique. Il a écrit deux livres pour *Audio Amateur Publications* et quelque 260 articles. Chuck a commencé à construire des appareils audio à tubes à vide à la fac. Il aime la voile et joue de la guitare jazz. Il

modifie les amplis de guitare et effets pour y réduire le bruit et la distorsion. Il construit et restaure les appareils de test audio.

épluchage de pomme

Il est plus facile d'ouvrir une huître qu'un produit Apple, à moins de disposer d'un outil spécial. Or, chez Elektor, nous détestons les boîtes noires fermées. Nous voulons savoir ce qu'il y a à l'intérieur. Comme vous sans doute, je portais encore des culottes courtes lorsque la curiosité m'a poussé à ouvrir mes premières radios et télévisions, sans crainte des mauvais génies électriques qui y rôdaient. Alors comment un boîtier en plastique aurait-il pu m'arrêter ?

Thijs Beckers (Elektor Labs)

Si vous me ressemblez, tout ce qui est électrique vous intéresse, et même vous aimante dès que c'est cassé. Quoi qu'il en soit, lorsque j'ai suspecté de défaillance l'adaptateur secteur de 90 W de mon écran *Cinema HD Display* d'Apple, j'ai été pris d'une irrésistible envie de l'ouvrir.

À première vue, le boîtier semblait pouvoir s'ouvrir en glissant un tournevis dans la rainure centrale, mais en fait les deux coques qui le composent sont soudées entre elles. Qu'importe, pour ça j'avais ma Dremel, un outil pour lequel le plastique est juste du beurre. Attention toutefois, certains composants sont placés très près du boîtier et peuvent être endommagés si le disque abrasif va trop loin dans le plastique.



Les fabricants chinois payés par Apple pour assembler la plupart de leurs produits utilisent une sorte de colle pour décourager un peu plus les fouineurs. J'ai donc dû retirer cette colle pour dégager entièrement la coque supérieure. Là je suis tombé sur le blindage, une plaque de cuivre qui enveloppe tout le circuit. J'ai retiré le fil reliant ce blindage et le circuit

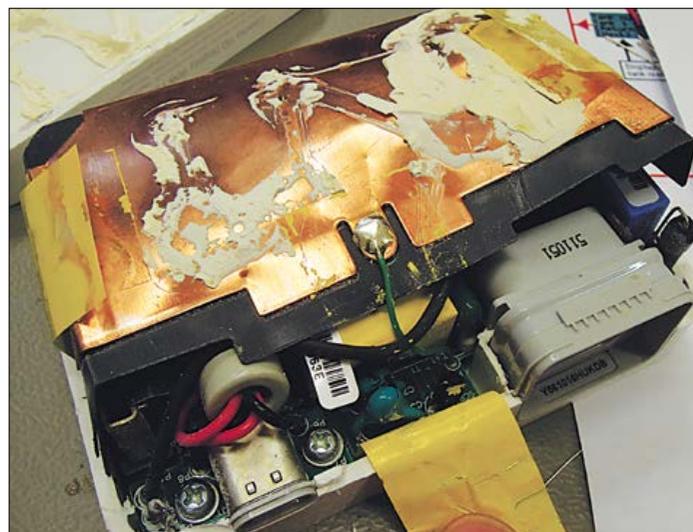


imprimé, dégagé la plaque, et pu découvrir l'alimentation à découpage qu'elle recouvrait.

J'ai d'abord remarqué que le circuit imprimé n'était pas vissé dans le boîtier mais y était simplement calé. Ensuite j'ai découvert que le connecteur d'écran était monté sur une petite carte à part. Le connecteur d'alimentation CA n'est pas non plus soudé directement à la carte principale. C'est plutôt astucieux, car ces connecteurs sont les seuls composants à subir des contraintes mécaniques. Les monter sur une carte séparée ou ne pas les monter sur la carte principale est donc la garantie qu'ils ne s'en détacheront pas. La carte qui supporte le connecteur d'écran est pour sa part vissée dans le boîtier. La configuration de l'alimentation à découpage est standard. On trouve des condensateurs et bobines de filtrage à l'entrée CA, un pont redresseur, ainsi qu'un grassouillet condensa-

teur chargé de fournir la haute tension CC au transformateur. Viennent ensuite deux MOSFET de puissance pour le découpage de la tension, puis le transformateur. Les MOSFET sont commandés par un L6571B de ST Microelectronics, un pilote de MOSFET haute tension à configuration en demi-pont et oscillateur intégré, un choix qui semble logique ici. Le circuit se poursuit et se termine côté secondaire avec un redresseur (doté d'un radiateur) et des condensateurs de filtrage de sortie. On trouve aussi bien sûr l'habituelle valetaille de composants passifs, autrement dit divers condensateurs et résistances. L'ensemble est commandé par un circuit intégré de STMicro (à la référence illisible ici).

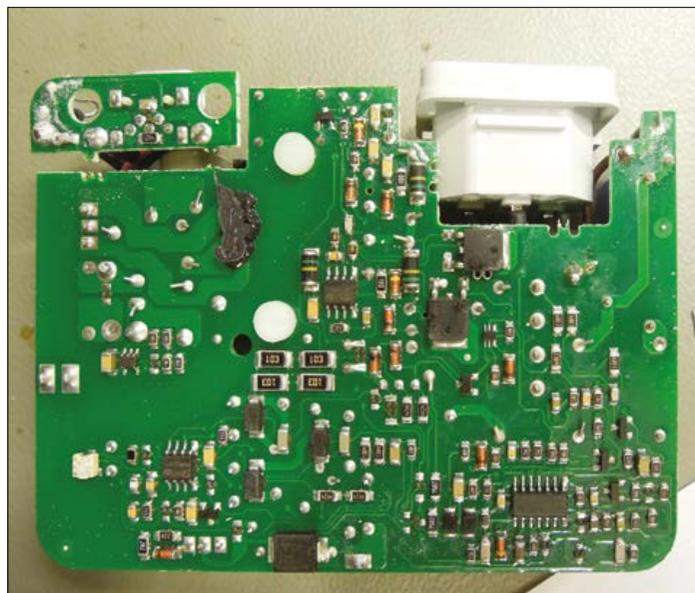
J'ai également découvert un circuit de correction du facteur de puissance (PFC, *Power Factor Correction*). Le transformateur partiellement recouvert d'adhésif jaune en fait partie.





La chaleur est dissipée par deux bandes métalliques placées sur deux bords de la carte. L'une refroidit la diode de redressement du circuit de sortie, l'autre ce que je présume être un FET de commutation pour le circuit PFC, par ailleurs basé sur un L6561D de ST Microelectronics. Je n'ai pas cherché à lire la référence du FET, pour cela il aurait fallu retirer un transformateur et cela ne m'a pas semblé en valoir la peine.

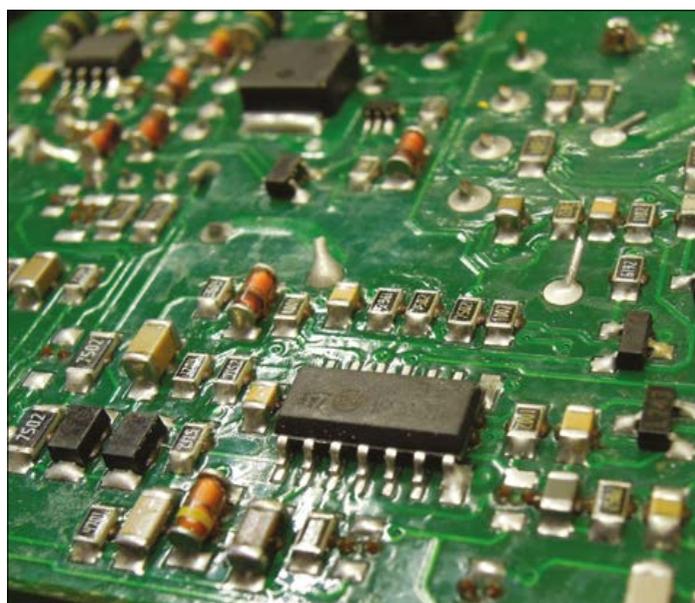
Les principaux MOSFET sont des STD5NM50 de STMicro en boîtier DPAK, de 500 V, 0,7 Ω et 7,5 A. Ils ne sont pas assemblés sur des radiateurs mais montés en surface, autrement dit leurs pattes reposent à même la carte. Il n'y a aucune surface de cuivre sous ces composants. Le circuit doit donc vraiment être très efficace pour réussir à convertir les 90 W des 100-240 V CA en ± 24 V CC (la tension nominale de l'adaptateur) tout en étant monté dans un boîtier en plastique hermétique. Une situation pas vraiment idéale pour dissiper l'excès de chaleur... Finalement l'adaptateur délivrait bien les 24 V attendus. Sur



Internet, j'ai lu qu'en raison de leur ancienneté les lampes CCFL utilisées pour le rétroéclairage des écrans *Cinema Display* avaient tendance à absorber trop de courant. Côté adaptateur, ce drainage peut donc déclencher l'activation de la protection contre les surintensités, et ainsi interrompre l'alimentation. Dans la plupart des cas, remplacer l'adaptateur de 90 W par un 130 W semble prolonger quelque peu l'espérance de vie de l'écran, mais l'issue reste fatale.

J'ai arrêté là mon investigation. Mon écran avait déjà vécu sept ans et je venais d'en commander un neuf, donc ce vieil adaptateur pouvait bien passer par pertes et profits. L'expérience n'en aura pas moins été aussi utile qu'intéressante. La pomme peut nourrir de plusieurs façons ! ◀

(150022 - version française : Hervé Moreau)



.LABorama **Big Brother** pourra jouer du tambour pour Spock

Florilège de projets [elektor.labs](http://www.elektor-labs.com) (www.elektor-labs.com)

Clemens Valens a choisi ceux qui lui ont semblé amusants ou intéressants. Allez-voir vous-même sur le site www.elektor-labs.com, vous en trouverez d'autres qui méritent aussi votre attention.

Mozart ou Tata Yoyo ?

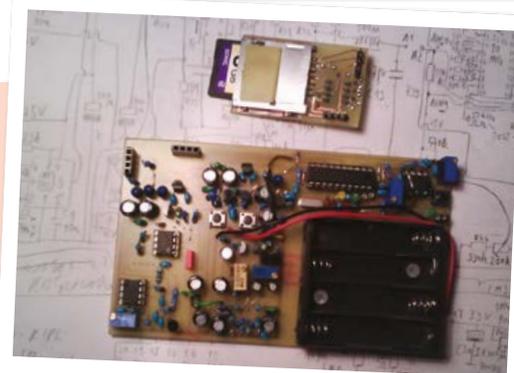
C'est vous qui voyez, ce sont vos visiteurs qui jugeront. Cette sonnette de porte programmable repose sur le microcontrôleur PIC16F873. Enregistrez la mélodie de votre choix (la musique de Psychose par exemple pour les plus sociopathes d'entre vous), la sonnette la jouera. Bonus, le bouton (un capteur capacitif en fait) est sans fil, ce qui facilite l'installation.



<http://po.st/doorbell>

Une idée survenue de façon aléatoire ?

Même lorsqu'ils ne fonctionnent pas correctement, les objets de l'électronique ont en général un comportement prévisible. En informatique, le seul aléatoire qui ait cours est le pseudo-aléatoire (en mettant de côté les comportements étranges dus aux problèmes de mémoire). Ce générateur de nombres aléatoires exploite le bruit d'avalanche créé par les zeners. Le bruit est nettoyé, en partie filtré, puis stocké sur une carte SD sous forme de séquence binaire. Sceptique sur ce soi-disant aléatoire vrai ? Demandez à MATLAB !



<http://po.st/random>

Allô ? C'est PING-Cube...

Voici un projet quelque peu mystérieux. La photo floue qui montre le PING-Cube semble avoir été prise par une caméra espion de la série Le Prisonnier, et le cube a un petit côté boîte aux lettres secrète du capitaine Kirk. Quoi qu'il en soit, ce PING-Cube est décrit comme étant un dispositif de surveillance de réseau LAN. L'état des composantes du réseau peut être consulté depuis un navigateur Internet, mais le PING-Cube peut aussi vous avertir par téléphonie mobile qu'un problème est survenu et vous indiquer sa nature. Ce qui serait encore plus cool, ce serait de pouvoir lui répondre : « Eh bien, pourquoi tu ne l'as pas déjà réparé ? »

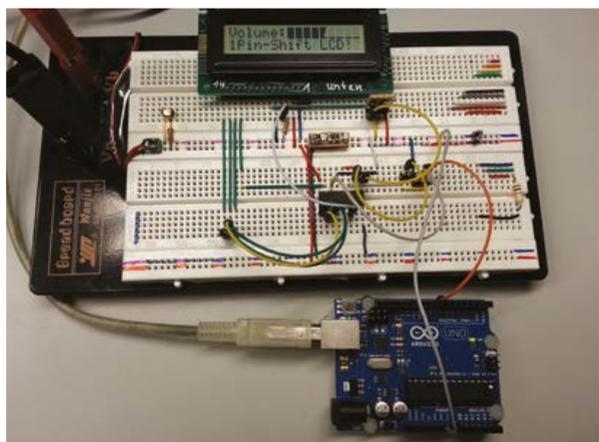


<http://po.st/PINGcube>

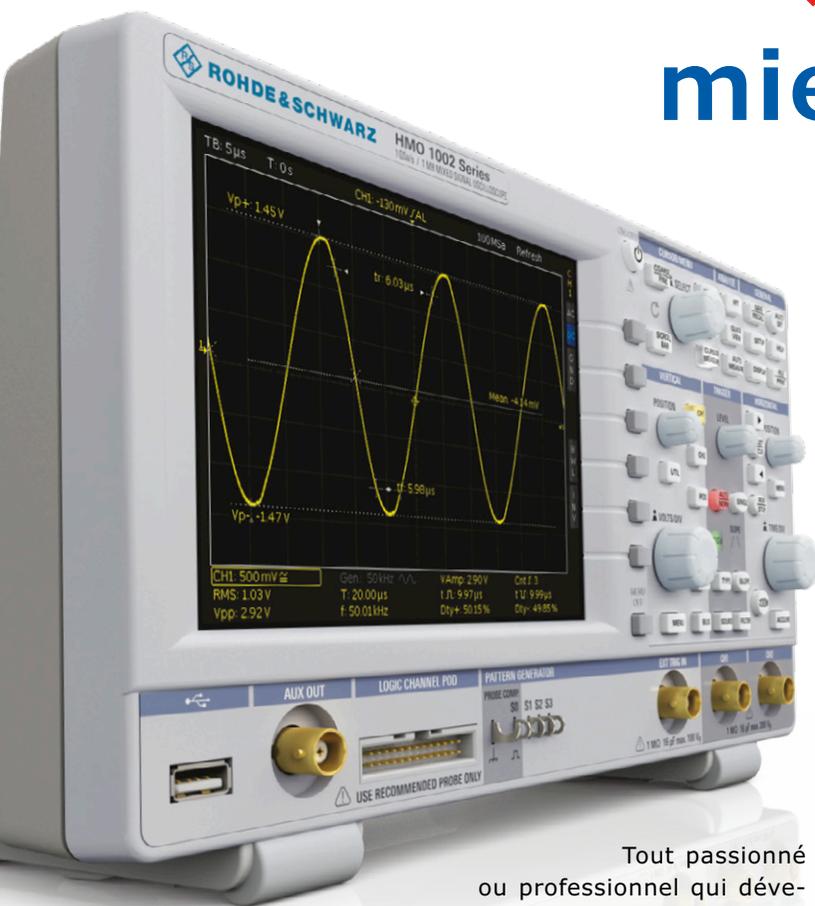
Interface à un fil

Réaliser que l'on manquera de ports d'E/S pour implanter la nouvelle fonction réclamée par son chef de projet n'est jamais un grand moment de la vie. Il n'y a guère d'autre choix alors que de migrer vers un microcontrôleur doté de plus d'E/S ou d'ajouter

un duplicateur de ports série-parallèle. Ce projet pousse la dernière solution à l'extrême puisqu'il réduit l'interface série à un seul fil pour commander un LCD standard avec une seule broche, un registre à décalage 74HC595 et quelques composants passifs. L'interface est moins rapide qu'une interface parallèle à 8 bits, mais suffit pour pas mal d'applications à bas débit.



<http://po.st/onewire>



Tout passionné ou professionnel qui développe ou répare des circuits électroniques dispose bien sûr de quelques instruments de mesure. Un multimètre, pour commencer, mais aussi une alimentation universelle et bien vite un oscilloscope. Dans le temps, les vrais étaient hors de prix pour un amateur, mais aujourd'hui, on en trouve déjà de beaux petits pour 200 €. Un appareil autonome convenable à deux canaux, en provenance de Chine, coûte à peine 500 €.

Quand vous entrez en possession d'un oscilloscope, il est judicieux de vous familiariser avec toutes les fonctions et possibilités que l'appareil vous offre. Et surtout de savoir exactement ce que l'oscilloscope fait des signaux mesurés avant de les porter à l'écran pour vous permettre d'interpréter convenablement les résultats. C'était déjà vrai pour les oscilloscopes analogiques, mais à présent, c'est encore plus important avec les versions numériques.

mieux mesurer à l'oscilloscope des bases aux astuces

Harry Baggen (Elektor)

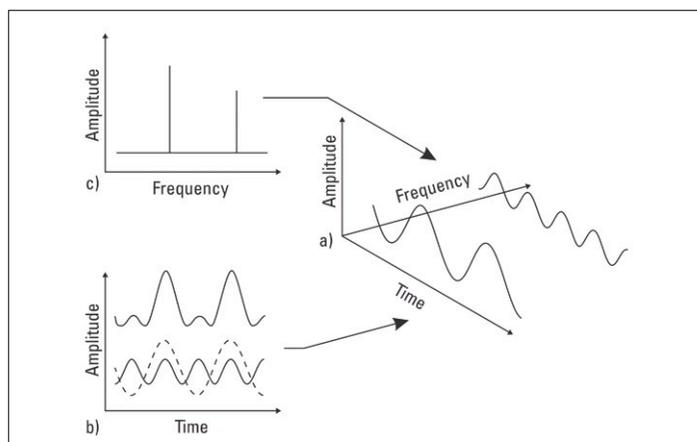
Un oscilloscope est un instrument de mesure très pratique pour qui sait exactement comment s'en servir. On trouve sur internet quantité de descriptions du fonctionnement et de la manipulation de l'oscilloscope. Mais n'est-ce pas d'abord aux grands fabricants de nous expliquer tout cela le mieux possible ? Elektor est allé enquêter sur ce qu'ils pouvaient nous apprendre.

Récemment, j'étais au labo occupé avec un concepteur à observer un signal d'aspect bizarre, alors que nous attendions une tout autre forme de signal sur ce point de mesure. L'image inadéquate semblait provenir d'une interférence entre le signal mesuré, agité de brusques changements de fréquence, et la fréquence d'échantillonnage de l'oscilloscope. C'est alors qu'il faut être bien conscient des traitements que l'appareil effectue avant que le signal ne soit visible à l'écran.

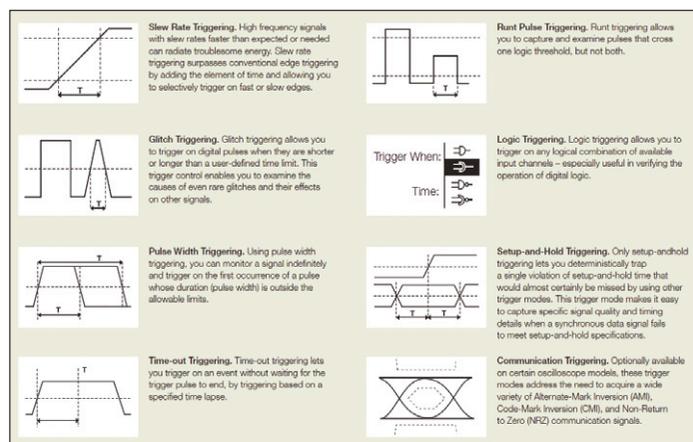
Vous dénicheriez sur internet beaucoup d'explications, de trucs et d'astuces si vous y cherchez des informations générales. Ce sont surtout les grandes firmes d'oscilloscopes qui fournissent sur leur site des indications que chacun peut lire ou télécharger gratuitement. Il ne s'agit pas seulement du mode d'emploi pour votre appareil, mais en creusant un peu, on trouve aussi des notions de base et des conseils.

Keysight

Le fabricant américain Keysight, anciennement Agilent, a constitué toute une base de données d'articles. Sous l'onglet *Technical*



1



2

support, il y en a des milliers [1] parmi lesquels des spécifications et des manuels pour les appareils de la marque, mais aussi beaucoup de documentation et d'information générale. Nous en avons épinglé quelques-unes intéressantes. L'application en deux parties *Fundamentals of Signal Analysis* fournit une introduction généraliste sur la mesure des signaux dans les circuits électroniques. La première partie [2] est une bonne étude sur la mesure dans les domaines de fréquence et de temps et explique le fonctionnement des appareils et des circuits, souvent en se servant d'analogies mécaniques notoires. La deuxième partie [3] va plus loin et aborde des sujets comme la FFT, l'échantillonnage et le repliement (**fig. 1**). Des notions importantes quand on se demande comment marche un oscilloscope numérique. Dans la bibliothèque de documents de Keysight, on trouve encore pas mal d'autres titres intéressants. Recommandons les notes d'application dites *8 Hints*. Ce sont des groupes de huit conseils sur des thèmes déterminés. Citons *8 Hints For Successful Impedance Measurement* [4] et *8 Hints for Making Better Measurements Using Analog RF Signal Generators* [5].

Tektronix

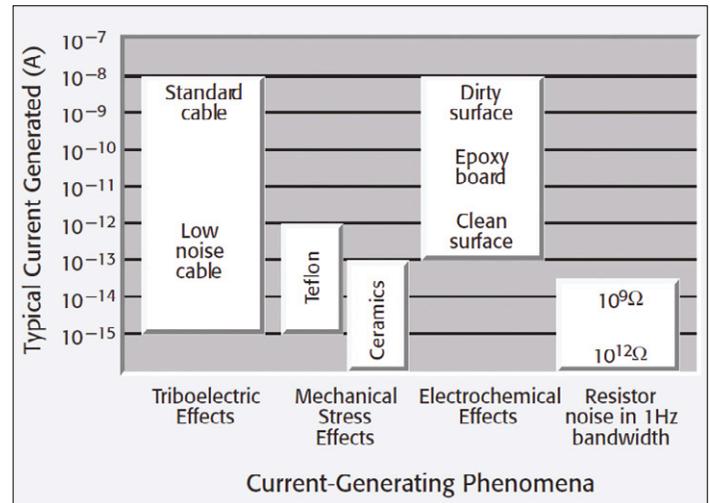
Tektronix a même réservé une place à part pour ses brochures d'information et notes d'application [6]. La première dans la liste est certainement un best-seller, c'est *Oscilloscope Primer* [7], qui s'appelle en fait *XYZs of Oscilloscopes*. Pour y accéder, Tektronix demande bien de décliner son adresse de courriel ou de s'inscrire pour pouvoir télécharger l'un ou l'autre PDF, mais cela en vaut la peine. Ce livre électronique d'info de 59 pages donne beaucoup d'explications sur l'utilisation et le fonctionnement des oscilloscopes. Il décrit au début très simplement les principales formes d'onde existantes, mais va très vite en profondeur sur les possibilités des instruments modernes, en particulier les modes de déclenchement (**fig. 2**).

Un autre document intéressant sur un sujet que les électroniciens sous-estiment souvent, c'est *Probes Primer*, l'ABC des sondes [8]. On y découvre en 60 pages comment les sondes sont construites, quelle influence elles ont sur le point de mesure, de quel type il s'agit et comment s'en servir dans différentes situations.

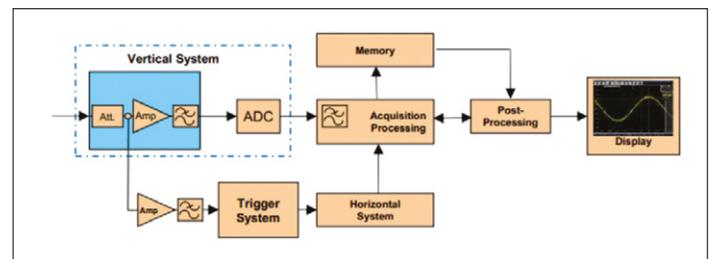
Par le site de Tektronix, on arrive aussi à Keithley, qui fait à présent partie de la firme. Nous y avons dégotté un intéressant e-book qui ne concerne pas directement les oscilloscopes, mais fournit en 240 pages des infos détaillées et des explications (cf. **fig. 3**) sur la mesure des petits signaux : *Low Level Measurements Handbook*. Utile aussi pour les mesures à l'oscilloscope. Un livre à succès, mais pour l'obtenir, vous devrez ici aussi léguer votre adresse.

Un peu plus loin...

Les autres fabricants d'oscilloscope mettent aussi à disposition des articles d'information pour tous ceux que le sujet intéresse. Chez Teledyne LeCroy, on trouve entre autres un article utile sur l'utilisation de sondes de mesure : *Probing Tutorial* [10]. Il traite principalement de l'influence de la sonde sur le point de mesure, avec différents exemples d'application. Un autre article, plus destiné aux utilisateurs « avertis » de l'oscilloscope, est le PDF *Why Differential* [11] dans lequel on expose les avantages d'exécuter les mesures en mode différentiel. Évidemment, Rohde & Schwarz ne pouvait pas manquer à l'appel dans ce rassemblement. Nous y avons également trouvé



3



4

des lectures intéressantes avec des informations de base au sujet des oscilloscopes [12]. Le plus agréable dans ce document, c'est qu'il explique l'ensemble de l'appareil à l'aide de diagrammes fonctionnels (**fig. 4**) jusqu'aux fonctions spéciales, avec un bout d'historique et une digression sur l'association avec les sondes.

Visiblement, quelques recherches suffisent à découvrir de nombreuses informations techniques de première main que chacun peut télécharger. Mais surtout, on peut être assuré que le contenu des articles techniques cités ici est correct, chose qui n'est pas garantie d'office avec tout ce qu'on trouve sur internet. ◀

(150032 – version française : Robert Grignard)

Liens

- [1] www.keysight.com/main/facet.jspx
- [2] <http://literature.cdn.keysight.com/litweb/pdf/5988-6765EN.pdf>
- [3] <http://literature.cdn.keysight.com/litweb/pdf/5988-6774EN.pdf>
- [4] <http://literature.cdn.keysight.com/litweb/pdf/5968-1947E.pdf>
- [5] <http://literature.cdn.keysight.com/litweb/pdf/5967-5661E.pdf>
- [6] www.tek.com/learning/oscilloscope-tutorial
- [7] <http://info.tek.com/www-xyzs-of-oscilloscopes-primer.html>
- [8] <http://info.tek.com/www-abcs-of-probes-primer.html>
- [9] <http://info.tek.com/KI-Low-Level-Measurements-Handbook-LP.html>
- [10] http://cdn.teledynelecroy.com/files/appnotes/lecroy_probing_tutorial_appnote016.pdf
- [11] http://cdn.teledynelecroy.com/files/whitepapers/wp_differential_measurements.pdf
- [12] www.rohde-schwarz-usa.com/rs/rohdeschwarz/images/Oscilloscope-Fundamentals_v1.1.pdf

compilées par **Beatriz Sousa**

Elektor rapproche les jeunes talents de leur futur employeur

Selon *Harvard Management Review*, aujourd'hui un grand PDG sur quatre est ingénieur. La demande est forte, les électroniciens talentueux sont choyés par les entreprises, et ils se retrouvent à des postes de grande responsabilité, techniques ou non. C'est pourquoi Elektor lance pour ses membres

elektor•talents

un nouveau service de recrutement : les entreprises recruteuses d'électroniciens seront interrogées par Elektor, les profils recherchés par elles seront présentés en toute indépendance par la rédaction. Si vous possédez les compétences requises, vous contactez Elektor qui jouera les entremetteurs. Si votre entreprise cherche à embaucher, contactez-nous pour évaluer les modalités de votre campagne de recrutement.

Merci de la visite

L'équipe d'Elektor était présente en nombre au grand salon international [Embedded World](#) du 24 au 26 février à Nuremberg. Comme prévu, l'envergure de l'événement dépassait celle des éditions précédentes. Notre équipe a accueilli nos membres en leur remettant notamment un exemplaire gratuit de notre premier hors-série *Embedded Business Special* ainsi qu'un kit de développement NXP LPC800. Nos visiteurs ont pu manipuler les T-Boards, l'horloge Clic-Clock, notre carte à FPGA, le *shield* d'extension Arduino... Nous espérons vous (re)voir lors d'un prochain salon !

READ ONLY MEMORY

Le magazine Elektor et son éditeur sont fiers de leur histoire. Cette rubrique montre pourquoi.

un ordinateur pour tout le monde !

Le *Junior Computer* d'Elektor date du début des années 80. Tout électronicien habile et persévérant pouvait s'en construire un chez soi. Disponible en kit, il était assemblé à partir de cartes proposées par Elektor.

Son processeur était le 6502 et sa mémoire de masse un lecteur de cassettes !

D'innombrables lecteurs firent ainsi leurs premiers pas en micro-informatique il y a 35 ans.



INDISCRÉTIONS • Bientôt paraîtra chez Elektor le livre de B. van Dam sur l'Edison d'Intel • passent toutes les marchandises commandées chez Elektor, ont battu leur record de livraisons • Johan Bergqvist a terminé le volume 1 de son recueil d'amplificateurs à tubes **The Bottle** publicité pour les Pays-Bas et la Belgique • Alex Politis est le nouveau gestionnaire de

nouvelles de la rédaction :

les éditions spéciales



Le premier sera consacré intégralement aux drones, sous l'hélice vibronnante de son rédacteur en chef Dré de Man, soutenu par une équipe internationale de spécialistes. Dans ce **hors-série drones** vous trouverez tout ce qu'il faut savoir sur ces engins envahissants. Sa date de parution est facile à retenir : ce sera le premier jour de l'été.

La curiosité des électroniciens n'a pas de limites, qu'il s'agisse d'impression 3D, de codage de logiciel, de sécurité, de LED ou de robotique. C'est pourquoi la rédaction d'Elektor proposera bientôt des éditions spéciales (ou hors-série) du magazine sur des sujets brûlants de l'actualité.



Surveillez votre boîte aux lettres électronique :

nos membres auront accès en priorité à cette édition spéciale, avec 10 % de réduction bien entendu.

Si vous avez vous-même des idées de thèmes auxquels consacrer de telles éditions spéciales, faites-les nous connaître.

Nos collègues Jan et Theo, entre les mains de qui journalières lors de la campagne Raspberry Pi **Builder** • Thom Wolf est le nouveau chef de l'approvisionnement d'Elektor •

PROFIL D'EXPERT

Elektor est au cœur d'un réseau de plus de 1.000 experts et d'auteurs engagés dans la publication de livres, d'articles, de DVD, de webinaires et autres événements. Coup de projecteur !



Nom : **Günter Spanner**

Âge : **49**

Études : **doctorat en physique**

Publications : **papiers scientifiques dans beaucoup de revues à comité de lecture. Plusieurs brevets dans le domaine de l'ingénierie biomédicale. Plus de dix livres et**

publications sur la technologie des capteurs et la microélectronique.

Enseignement : **maître de conférence dans plusieurs instituts et universités techniques en Europe.**

Qui est Günter Spanner ?

Après avoir obtenu mon doctorat à l'université de Munich, j'ai rejoint le *Corporate Research Center* d'ABB en Suisse ainsi que l'ETH Zurich en tant que chercheur. Ensuite chez Siemens, j'ai poursuivi une carrière d'expert scientifique dans les télécommunications par fibre optique et les applications des capteurs. J'habite et je travaille dans un petit village au pied des Alpes bavaroises.

Quels sont vos principaux domaines d'expérience ?

Applications des fibres optiques, technologie des lasers et capteurs optiques. Chez Siemens, j'étais responsable de la conception des systèmes, des technologies de production et de la conformité. J'ai aussi dirigé plusieurs groupes de R&D ainsi que dans les achats et la gestion de la technologie.

Quels seront les prochains développements clés ?

La bio-électronique, je pense. Deux gros projets sont en cours : BRAIN aux États-Unis et le *Human Brain Project* en Europe. J'espère qu'ils seront à l'origine de nouvelles ruptures technologiques en bio-électronique et en neuro-électronique.

Sur quels sujets allez-vous écrire à l'avenir ?

En ce moment, je travaille sur les microcontrôleurs, la technologie des capteurs et l'informatique physique. Ma prochaine publication sera un cours détaillé sur les ATXmega, une impressionnante nouvelle famille de puces Atmel.

Si Elektor vous donne 100 €, qu'achèteriez-vous ?

Je les investirais sans doute dans l'Internet des Objets sans fil. Son potentiel est énorme pour la domotique ainsi que pour beaucoup d'autres applications industrielles. J'espère qu'Elektor voit les choses comme moi...

Quels sont les plus grands concurrents d'Arduino ?

Le Raspberry Pi a capté une bonne partie des utilisations d'Arduino. Il y a beaucoup d'applications du Raspberry que l'Arduino ne sera jamais en mesure de concurrencer. Au contraire, je ne vois guère de choses réalisables avec un Arduino qui ne le seraient pas avec un Raspberry.



hexadoku votre bain parfumé à la fleur d'hexadécimal !

Déshabillez-vous ! Oui, oui, le haut, le bas, enlevez tout, la cravate surtout, vous vous sentirez mieux. Gardez vos chaussettes si vous voulez, mais pas le fer à souder. Prenez votre bouée si vous en avez une, une couverture chaude (pour après), un crayon, mais *n'oubliez pas la gomme et l'indispensable taille-crayon*. Enjambez prudemment le bord, ne dérapez pas et laissez-vous glisser doucement dans votre bain mensuel parfumé à la fleur d'hexadécimal. Vous en sortirez dégrassé(e), rajeuni(e). Remplissez la grille selon les règles, envoyez-nous votre solution, et vous gagnerez peut-être l'un des cadeaux offerts 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** fois dans chaque rangée, 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 cinq des réponses internationales correctes arrivées dans les délais ; leurs auteurs recevront un bon d'achat de livres Elektor d'une valeur de **50 €**.
À vos crayons !

Où envoyer ?

Envoyez votre réponse (les chiffres sur fond grisé) avec vos coordonnées par courriel, avant le **1^{er} mai 2015** à l'adresse

hexadoku@elektor.fr

Les gagnants

La solution de la grille du numéro de janvier-février est **3146A**

Les cinq bons d'achat Elektor d'une valeur de 50 € vont à :

Yngve Bordoy (Norvège) | **Harrie Rouwhorst** (Pays-Bas) | **Hartwig Wander** (Allemagne)

Jairo Rotava (Brésil) | **Jean Verdois** (France)

Bravo à tous les participants et félicitations aux gagnants !

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

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

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.

LA CULTURE DU SERVICE, LA TECHNOLOGIE COMME MOTEUR

PCB prototypes
& petites séries

Délais courts & tenus
de 5 jours à 24 h

Multicouches,
perçages borgnes,
enterrés

Homologation UL
certification ISO 9001

Acteur d'innovations
avec CIRCOM
« black box inside »

Matériaux HTG, RF,
SMI, semi-flex
et cuivre épais

Plus d'informations sur www.cirly.com

email : commercial@cirly.com

Tél : 04 72 71 59 59

Selectronic

L'UNIVERS ÉLECTRONIQUE

**60 000
RÉFÉRENCES**

Robotique

Outils

Appareils de mesure

Imprimante 3D

Composants
électronique

Depuis près de 40 ans, Selectronic propose une offre unique de +60.000 références de composants électroniques, électriques, mécaniques et informatiques ainsi que tout l'outillage et les appareils de mesure qui permettent d'aborder tous les projets d'électronique et de robotique.

Offre spéciale pour les lecteurs d'Elektor :

5% de remise sur la prochaine commande y compris sur les promotions en cours*

CODE PROMO : ELKTR



LILLE

Selectronic Lille :
16, rue Jules Verne - 59790 Ronchin



PARIS

Selectronic Paris :
11, place de la Nation - 75011 Paris



Service client
Conseil technique
0 970 82 01 66**



Service livraison
en **24h**

www.selectronic.fr