

nouvelle formule !

juillet / août 2017 | n° 466

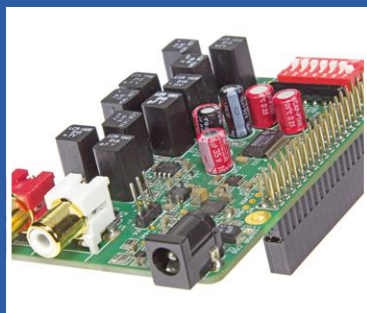
www.elektormagazine.fr

elektor

DÉCOUVRIR

CRÉER

PARTAGER



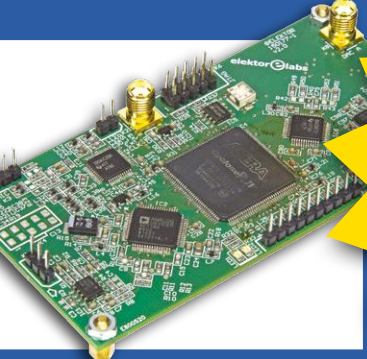
CN/A audio pour RPi

lecteur audio sur réseau
avec Volumio



des circuits vraiment imprimés

imprimante de carte/four à
refusion Voltera V-One



carte DSP à FPGA pour SDR à bande étroite

émetteur-récepteur
complètement programmable

nWatch un système de développement au poignet



ou comment fabriquer votre montre intelligente

Dans cette édition :
7 projets du labo
8 projets de lecteur
Bluetooth, HF,
LED RVB, FPGA, DSP,
MicroPython...

gradateur à deux bornes • conception de circuit imprimé

• atténuateur HF commutable • ESP32 (mutation de
l'ESP8266) • BBC micro:bit pour électroniciens (2) •

l'électronique dans l'espace • simulateur de murmure
de la mer • Rétronique : oscilloscopes Tektronix 556
et 565 • tirer 5 V d'une cellule alcaline épuisée • R.

Lacoste : absorption diélectrique des condensateurs •

lampemètre façon Tektronix 570 • questions d'éthique •

MicroPython et PyBoard • composants en astronautique •

MC sur SUPRA 2.0 • commande de feux tricolores • ukulé-LED

• APN : générateur de mire • feux de position

multifonctions • renifleur avec module radio RFM12

L 19624 - 466 - F: 15,50 € - RD



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

ISSN 0181-7450

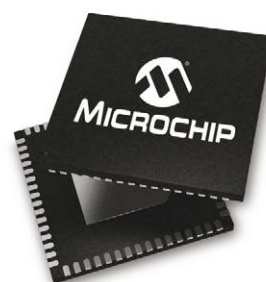
Les microcontrôleurs PIC18F "K40" intègrent un convertisseur A/N intelligent avec des fonctions de filtrage et d'analyse de signal

Les microcontrôleurs PIC® 8 bits sont idéaux pour le tactile et le traitement de signal



Les périphériques indépendants du cœur, ou périphériques CIP (pour Core Independent Peripherals) disponibles sur la famille PIC18F « K40 » de microcontrôleurs PIC® 8 bits permettent le filtrage et l'analyse de signal pour les applications tactiles avancées et le traitement de signal.

Parmi les périphériques CIP analogiques intelligents, on compte un convertisseur A/N avec capacités de calcul (ADC2) permettant des fonctions de moyenne, filtrage, sur-échantillonnage et comparaison automatique de seuils. Les microcontrôleurs intègrent également des CIP de sécurisation et des PWM matériels avec de nombreuses interfaces de communication ainsi que de grandes capacités de mémoire Flash et EEPROM sur puce. Ces caractéristiques, associées à une tension de fonctionnement de 5 V, permettent à la famille PIC18F « K40 » d'améliorer la flexibilité d'un système tout en minimisant son coût.



microchip
DIRECT
www.microchipdirect.com

 **MICROCHIP**

www.microchip.com/EUPIC18FK40

Le nom et le logo de Microchip, le logo Microchip, MPLAB et PIC sont des marques déposées de Microchip Technology Incorporated aux États-Unis et dans d'autres pays. Toutes les autres marques commerciales ci-dessus mentionnées sont respectivement la propriété des sociétés concernées. © 2017 Microchip Technology Inc. Tous droits réservés. DS30010138A, MEC2125Fre05/17

40^{ème} année,
n° 466 – juillet/août 2017

ISSN 0181-7450
Dépôt légal : juin 2017
CPPAP 1113 U 83713
Directeur de la publication : Donatus Akkermans

Elektor est édité par :
PUBLITRONIC SARL
c/o Regus Roissy CDG
1, rue de la Haye
BP 12910
FR - 95731 Roissy CDG Cedex

@ : service@elektor.fr

Tél. : (+33) 01.49.19.26.19
du lundi au vendredi de 10h à 13h

Fax : (+33) 01.49.19.22.37

www.elektor.fr | www.elektormagazine.fr

Banque ABN AMRO : Paris
IBAN : FR76 1873 9000 0100 2007 9702 603
BIC : ABNAFRPP

Publicité :

Margriet Debeij
Tél. : 02 41/955 09 174
Fax : 02 41/955 09 013
Gsm : (+49) 170 550 539 6
margriet.debeij@eimworld.com

DROITS D'AUTEUR :

© 2017 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.



Êtes-vous prêt ?

- *Voulez-vous parler à un humain ?*
- *Non, ça va, merci.*
- *Avez-vous été sarcastique et/ou grossier ?*
- *Pas du tout. C'est une infraction que d'être grossier envers un agent. C'est compris.*

Cette conversation est extraite du film de science-fiction *Elysium* de Neill Blomkamp, sorti en 2013.

Je me suis souvenue de cette scène en parcourant il y a quelques jours un salon de robotique. Pratiquement un stand sur deux proposait un « robot semi-humanoïde d'aide à la personne ». Leurs créateurs vantaient les mérites de leur progéniture : faciliter la vie quotidienne des personnes âgées, accueillir les clients d'un hôtel, vous renseigner dans un lieu public, vous guider dans un musée, vous conseiller dans le choix de vêtements dans un magasin...

Un robot dans une cabine d'essayage ? Et quoi encore ?
J'ai reculé lorsque l'un de ces engins s'est adressé à moi (très poliment) !

Et vous, êtes-vous prêt à confier la garde de vos enfants à un humanoïde ? Ainsi que celle de vos vieux parents ?

Voilà de quoi méditer pendant la pause estivale.

Bonnes vacances à toutes et tous !

Mariline Thiebaut-Brodier



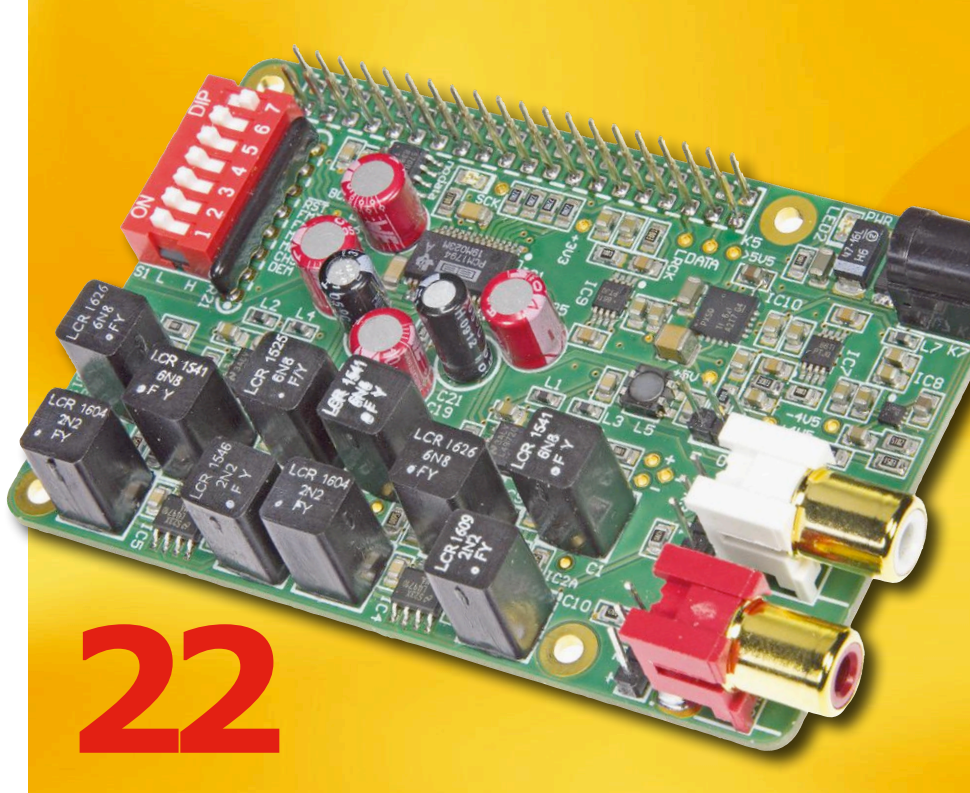
Notre équipe

Rédactrice en chef :	Mariline Thiebaut-Brodier (redaction@elektor.fr)
Rédaction internationale :	Thijs Beckers, Jan Buiting, Jens Nickel
Laboratoire :	Ton Giesberts, Luc Lemmens, Clemens Valens (responsable), Jan Visser
Coordination :	Hedwig Hennekens
Ont coopéré à ce numéro :	Patrick Bechler, Yves Georges, Robert Grignard, Denis Lafourcade, Guy Matt, Jean-Louis Mehren, Denis Meyer, Helmut Müller, Kevin Petit, Xavier Pfaff, Guy Raedersdorf, Alexandre Roy
Service de la clientèle :	Cindy Tijssen
Graphistes :	Giel Dols, Mart Schroijsen, Patrick Wielders
Elektor en ligne :	Daniëlle Mertens

- 5 bientôt dans Elektor
- 13 Elektor : guide de connexion(s)
- 33 agenda juillet 2017
- 59 vol tous azimuts
piloté par Clemens Valens
- 68 projet 2.0
corrections, mises à jour et courrier des lecteurs
- 79 optocoupleurs
drôle de composant n°31
- 85 Q & R (presque) tout ce que vous avez toujours voulu savoir sur...
les composants en astronautique
- 100 Rétronique
oscilloscopes Tektronix 556 et 565 à deux faisceaux
- 126 questions d'éthique
qui invitez-vous chez vous ?
- 128 l'e-choppe d'Elektor
- 130 Zenerdoku
casse-tête pour elektorniciens

en coulisse

- 6 concevoir un circuit imprimé ?
pas si facile !
de l'art de placer des pistes
- 29 trucs et astuces
Appareil-photo numérique en guise de générateur de mire
- 38 le bus I²C
2^e partie : le bus en action
- 44 des circuits vraiment imprimés
imprimante de carte/four à refusion Voltera V-One
- 88 mutation de l'ESP8266
découvrons l'ESP32 avec l'EDI d'Arduino
- 94 l'électronique dans l'espace
la recherche spatiale pour tous
- 99 bruits de labo
« le menu, s'il vous plaît ! »
- 108 hors-circuits de R. Lacoste
des condensateurs qui se rechargent tout seuls !

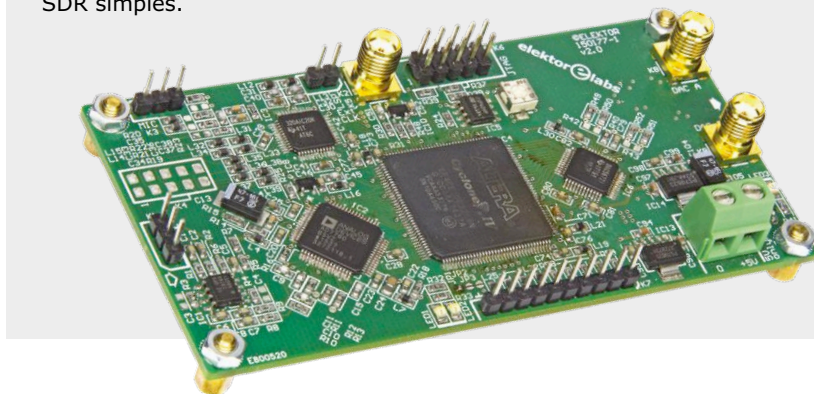


22

carte DSP à FPGA pour SDR à bande étroite

émetteur-récepteur
complètement programmable

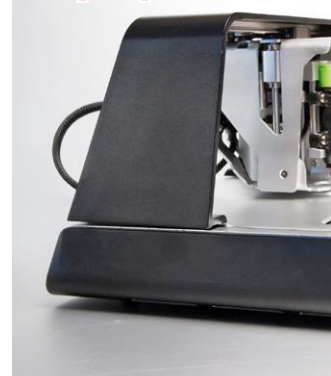
Voici une carte de traitement du signal numérique basée sur un FPGA qui comporte tout le nécessaire pour réaliser le travail en bande de base des modes traditionnels à bande rétrécie, comme SSB, radiotélégraphie (CW) et modulation d'amplitude (AM). Elle apporte une solution à de nombreuses difficultés et limitations rencontrées avec les plateformes SDR simples.



sur la scène : les projets

- 14 gradateur à deux bornes
commande de LED, lampes et chauffages
- 18 atténuateur HF commutable
niveaux de signal précis par atténuation réglable
- 22 CN/A audio pour RPi
lecteur audio sur réseau avec Volumio

44



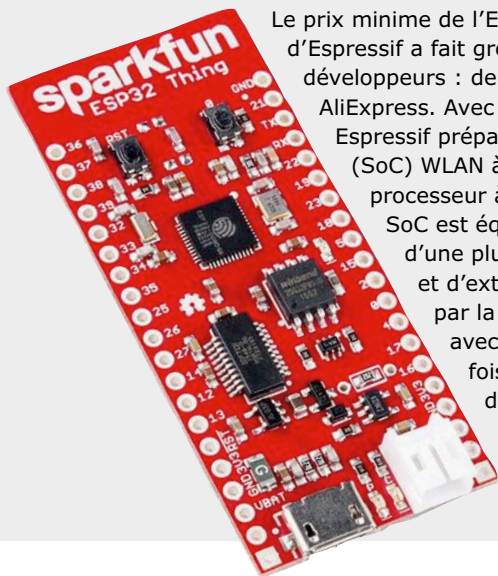
CN/A audio pour Raspberry Pi

lecteur audio sur réseau avec *Volumio*

mutation de l'ESP8266

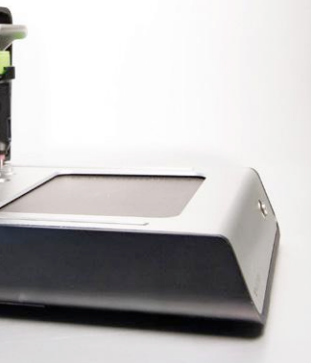
découvrons l'ESP32 avec l'EDI d'Arduino

88



Le prix minime de l'ESP8266 d'Espressif a fait grosse impression chez les développeurs : deux ou trois euros chez AliExpress. Avec sa version évoluée, l'ESP32, Espressif prépare ses systèmes sur puce (SoC) WLAN à l'avenir : en plus d'un processeur à double cœur, ce nouveau SoC est équipé d'un module Bluetooth, d'une plus grande mémoire de travail et d'extensions des E/S. Commençons par la programmation de l'ESP32 avec l'EDI d'Arduino. La prochaine fois, nous verrons les outils de développement ESP-IDF.

imprimante de
carte/four à refusion
Voltera V-One



- 30 BBC micro:bit pour électroniciens (2)**
enregistreur de données et oscilloscope
- 35 simulateur de murmure de la mer**
un doux bruit pour vous aider à vous endormir
- 48 nWatch, un système de développement au poignet**
ou comment fabriquer votre montre intelligente

- 62 tirez 5 V d'une cellule alcaline épuisée**
essorez-la à fond
- 63 ce bon vieux lampemètre Tektronix 570**
on s'en refait un ?
- 70 carte DSP à FPGA pour SDR à bande étroite**
émetteur-récepteur complètement programmable
- 80 MicroPython et PyBoard**
de la LED qui clignote ... au serveur web qui fait clignoter une LED
- 104 renifleur avec un module radio RFM12**
espionnez l'éther
- 112 MC sur SUPRA 2.0**
tout petit signal, mais grande qualité
- 116 commande de feux tricolores**
pour le modélisme ferroviaire
- 118 ukulé-LED**
des LED adressables en série pour apprendre à jouer du ukulélé
- 122 feux de position multifonctions**
pour tout modèle réduit volant



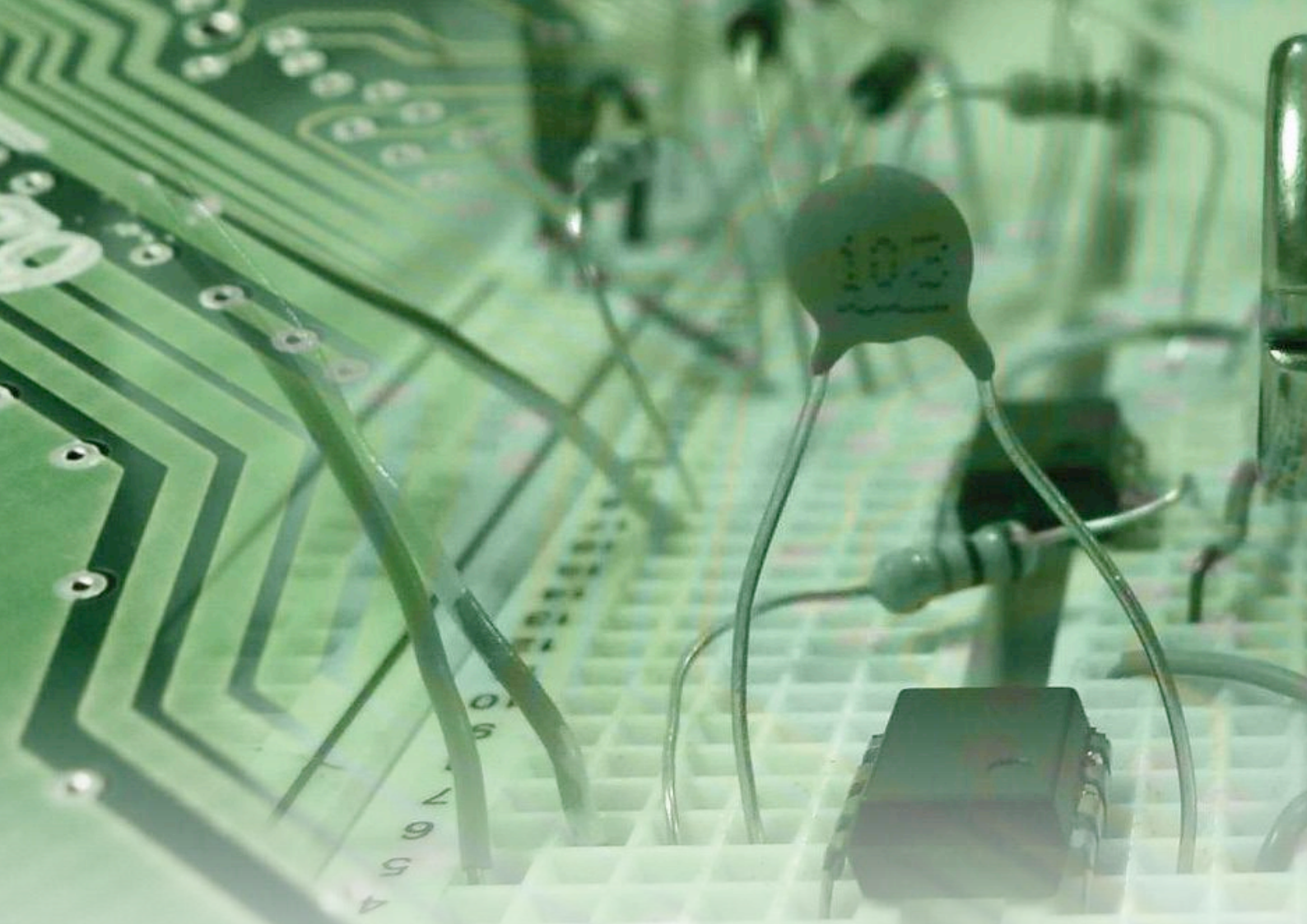
bientôt sur ces pages

Extrait du sommaire du prochain numéro :

- Balance de cuisine avec Arduino
- Carte radio pour la carte DSP à FPGA
- Enregistreur de données pour chauffage
- Générateurs de fonction
- Interface IdO
- Lampe à effet Peltier
- Programmation d'iPhone pour les électroniciens
- Raspberry Pi Zero W en pratique
- Etc.

Sous réserve de modification.

Le numéro de septembre-octobre 2017 paraîtra le 16 août 2017.



concevoir un circuit imprimé ? pas si facile !

de l'art de placer des pistes

Même si une approche aussi naïve peut marcher pour des montages simples, de faible puissance et à basse fréquence, c'est une erreur. Les montages simples ont aussi besoin de bons CI. Non seulement les bons CI font que le montage fonctionne et est fiable, mais ils limitent aussi au maximum les problèmes de CEM, ils ont des points de test pour l'assemblage et la réparation et ils sont faciles à intégrer dans le montage final. Parce qu'une carte de CI est un composant à part entière du système, elle doit être de même qualité que les autres composants. La conception d'un CI n'est pas destinée à plaire à son concepteur ; elle doit plaire à son utilisateur final, quel qu'il soit (**fig. 1**).

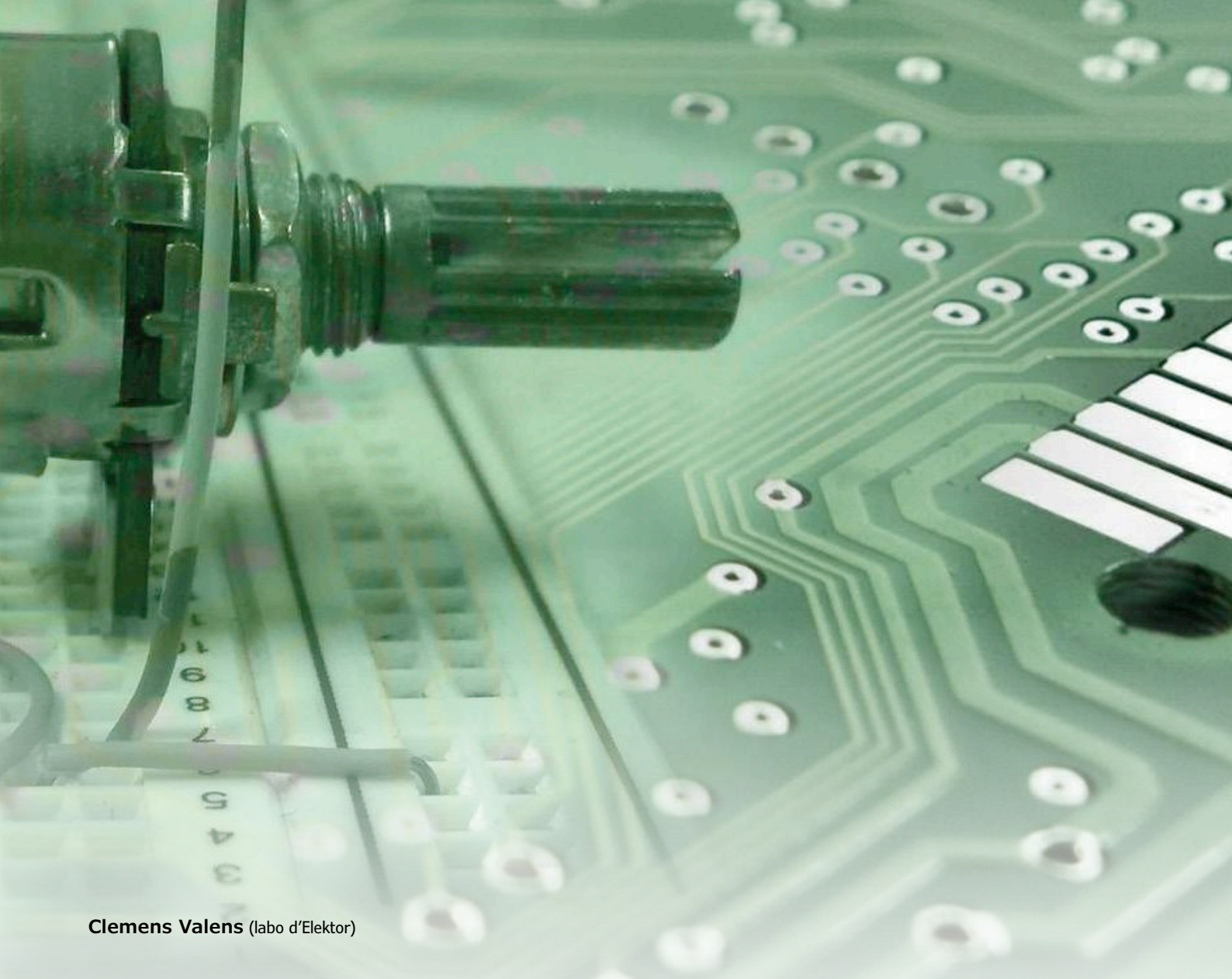
Dans ce qui suit, j'utilise des acronymes et du jargon. Référez-vous au **Glossaire** plus loin dans cet article pour mieux comprendre le texte. De plus, malgré sa longueur, cet article est loin d'être complet. La conception de CI est tout simplement un sujet trop vaste pour être traité intégralement ici.

Intégration dans le système

La première étape – généralement ignorée – implique d'étudier comment intégrer la carte dans le système. Quelle forme lui donner ? Où placer les trous de fixation ? Combien ? Et les câbles et fils de et vers la carte ? D'où viennent-ils, où vont-ils ? Y a-t-il une hauteur limite ? D'éventuels problèmes de chaleur ? Comment interagit l'utilisateur avec la carte ? Y a-t-il des câbles à brancher ? À l'avant, à l'arrière, sur un côté ? Codeur rotatif ou boutons-poussoirs ? Afficheur ? LED ? Des problèmes potentiels avec le matériau du boîtier ? Oh oui – un autre aspect secondaire – l'alimentation électrique ?

Même si le système se limite à votre montage fixé dans un boîtier, avant de concevoir le CI, choisissez un boîtier adéquat et adaptez la taille et la forme du CI à celui-ci.

À moins d'exceller dans le travail du métal, du bois et du plastique, ou de disposer de machines à commande numérique et de découpe laser, limitez au maximum la partie mécanique.



Clemens Valens (labo d'Elektor)

Pour beaucoup, concevoir un circuit imprimé (CI) n'est qu'un détail ou une tâche secondaire. Une fois le montage soigneusement testé, ils l'implantent sans trop réfléchir sur une carte pour le rendre plus facile à déplacer et ne pas avoir de fils qui se débranchent ou de composants qui tombent. Comme ils ont tort...

Fabrication de la carte

Autre question importante : comment est fabriquée la carte ? Gravure à la maison ? Êtes-vous à l'aise avec ça ? Sinon, évitez les vias et restez en simple face. Les cartes double-face exigent un parfait alignement des films du dessus et du dessous, et les vias métallisés peuvent demander un gros travail, surtout s'ils sont nombreux. Ceci dit, beaucoup obtiennent de très bons résultats à la maison avec des cartes double-face ; c'est surtout une affaire de savoir-faire et d'expérience. Vous devriez éviter les plans de masse et autres aplats cuivrés, car vous n'avez pas de masque de soudure pour éviter les ponts de soudure. Trouver et couper des courts-circuits presque invisibles n'est pas non plus un travail agréable. Pourquoi ne pas élargir les pistes pour éviter qu'elles ne soient rongées par le liquide de gravure ? Pour éviter les ponts de cuivre, ne mettez pas les pistes trop proches les unes des autres. Mettez des pastilles plus grandes si vous n'êtes pas trop bon en perçage (**fig. 2**).



Figure 1. Votre carte peut être réalisée dans une usine comme celle-ci, quelque part très loin de votre point de vue. Donc, pour avoir les meilleures chances de succès, concevez-la de façon à ce qu'elle puisse être faite avec le moins d'explications possible.

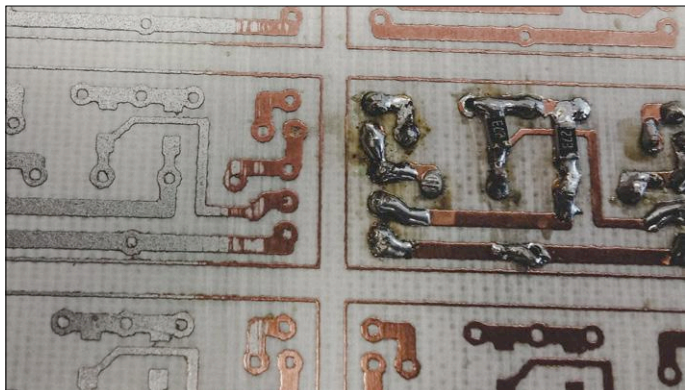


Figure 2. Un panneau gravé à domicile avec beaucoup de trous restant à percer et de découpes à faire. Est-ce que cela vaut réellement la peine de faire ça vous-même ?

Une fraiseuse à CI a des exigences similaires, mais à moins que vous ne soyez un très bon graveur, elle a une meilleure précision. La densité des pistes peut être supérieure, mais pas trop non plus. Si le fraisage est trop profond, des pistes peuvent disparaître. À l'inverse, si le fraisage n'est pas assez profond, des courts-circuits peuvent se créer. Si la carte n'est pas plate, les deux problèmes peuvent survenir simultanément. N'oubliez pas non plus que contrairement à la gravure, le fraisage laisse le cuivre inutilisé sur la carte. Ceci rend le soudage plus difficile (là encore, pas de masque de soudure), mais peut aussi créer des situations dangereuses avec des montages connectés à des hautes tensions (réseau CA) parce que les distances entre pistes ne sont plus respectées. Ceci peut aussi poser problème pour les montages avec des entrées à haute impédance. On peut fraiser le cuivre inutilisé, mais cela allonge le temps de fabrication de la carte.

Vous pouvez fraiser manuellement une carte avec un outil de type Dremel ou avec des embouts de fraisage adaptés comme ces forets arrondis qu'utilise votre dentiste. Comme lui/elle, entraînement et habileté sont nécessaires pour obtenir des résultats sans effort.

Ensuite il y a, bien sûr, la voie professionnelle. Les services

de groupage qui combinent votre projet avec ceux d'autres clients sur une carte de grande dimension sont largement présents sur l'internet. Comme les prix et délais de livraison varient beaucoup, comparez plusieurs fournisseurs. Les prix dépendent surtout de la surface de la carte, de sa classe, du nombre de couches et du délai de fabrication, et l'ajout d'options augmentera le prix. Certains services permettent de supprimer des options, d'autres non et vous pouvez finir avec des caractéristiques inutiles pour vous. Les services de groupage délocalisés semblent attractifs, mais les délais de livraison ne sont pas toujours respectés et les colis peuvent se perdre. Certains services proposent un prix par carte incluant le port, d'autres facturent la préparation et/ou les coûts d'outillage.

Généralement il n'y a aucune raison de dessiner un CI aussi petit que possible. Il sera moins cher à produire, mais la carte sera plus difficile à router, plus compliquée à intégrer et encore plus à réparer.

Un mot sur le fraisage : quelquefois un fraisage non circulaire est nécessaire pour ajuster une pièce ou pour accéder à quelque chose. Malheureusement il faudra encore un autre outil de fraisage, ce qui augmentera sans doute le prix de la carte. Certains fabricants n'ajoutent pas de coûts si le foret de fraisage est le même que celui utilisé pour le contour de la carte. Le fraisage peut ne pas être aussi détaillé et précis que souhaité, alors avant de fraiser des formes, vérifiez auprès du fabricant ce qui est possible ou pas.

Placement des composants

Les composants ne peuvent pas être mis n'importe où ou n'importe comment, même quand le circuit vous permet ce luxe virtuel. Pour faciliter le montage, la vérification et la réparation, les composants doivent être disposés en lignes et colonnes, et orientés uniformément. Cela paraît inutile si vous assemblez vous-même, mais cela vous facilitera la vie, de même qu'aux professionnels de l'assemblage/réparation des cartes.

Pour des raisons électriques, les composants doivent – en général – être placés aussi proche que possible les uns des autres. Bien sûr, il y a bien d'autres critères à prendre en compte ici, comme les chemins de retour de courant, éviter la diaphonie et autres couplages inductifs ou capacitifs non désirés, etc. Bref,

Petit glossaire de la conception de CI

Aplat (cuivre) : voir Plan
 Auto-routeur : Saint Graal des développeurs de logiciel de conception de CI
 Borne : voir Broche
Bottom : couche inférieure de la pile de la carte, nommée aussi côté soudure
 Broche : point de connexion d'un composant comme une pastille, une broche, un contact, etc.
 Chevelu : une représentation visuelle de tous les réseaux non connectés
 CI : Circuit Imprimé
 Classe : résolution ou densité d'un CI ; plus la classe est élevée, plus les éléments du CI et les espacements peuvent être petits
 CMS : Composant à Montage en Surface, utilisant la technologie TMS
 Couche : surface supportant des éléments

du CI comme les pistes, les aplats et les composants
 Court-circuit : connexion involontaire entre deux ou plusieurs broches
 DRC : **Design Rule Check**, vérification que les éléments du CI respectent un jeu de règles de conception comme une largeur de piste minimale, un diamètre de perçage minimal, un espacement minimal entre pastilles, etc.
 Élément de CI : objet imprimé sur une couche, y compris le contour de la carte
 ERC : **Electrical Rule Check**, vérification que les réseaux sont connectés, ne se recouvrent pas, sont sans courts-circuits dus aux aplats de cuivre, etc.
 Espacement : distance entre deux ou plusieurs éléments du CI
 Excellon : format de données pour les machines

numériques de perçage et routage
 Frein thermique : connexion pastille-cuivre (plan ou piste) évitant que la chaleur de la soudure ne soit absorbée par le cuivre environnant
 Gerber : format de données vectoriel ASCII pour des images en deux dimensions et deux couleurs
 IAR : **Inner Annular Ring**, anneau annulaire interne
 Masque de soudure : masque avec des ouvertures où la soudure est permise. Non seulement un masque de soudure évite les courts-circuits, mais il empêche aussi la soudure de glisser de la pastille ce qui pourrait conduire à de mauvais joints de soudure ou au désalignement d'un composant CMS.

préparez avec soin le placement des composants ! Un amplificateur de puissance n'a clairement pas les mêmes exigences qu'un instrument de mesure de haute précision.

Toutefois, la technique de soudage utilisée peut encore imposer un autre jeu de contraintes. L'environnement peut aussi avoir un impact sur le placement des composants. Si l'humidité et/ou la poussière polluent la carte, il faut un espace plus grand autour des pistes et composants pour éviter les courts-circuits et les fuites. Tout ceci est étroitement lié aux normes et standards auxquels votre carte doit répondre. Même si cela peut ne pas sembler un problème pour la plupart des cartes FLTM, il y a de bonnes raisons pour que ces standards existent, et les étudier peut réellement vous apprendre une chose ou deux.

Placez les composants traversants sur le dessus, nommé aussi côté composants. Placez tous les CMS du même côté, dessus ou dessous ; évitez d'en mettre des deux côtés, car cela rend la fabrication plus chère. C'est une bonne pratique d'éviter les CMS des deux côtés, même pour une seule carte ou de petites séries (**fig. 3**).

Les cartes avec des CMS doivent avoir au moins trois mires comme points de référence pour les machines bras-transfert. Les gros circuits intégrés avec beaucoup de contacts à pas fin peuvent aussi nécessiter des mires pour garantir un alignement correct. Les cartes FLTM peuvent, bien sûr, se passer de mires, mais pourquoi ne pas prendre une habitude de professionnel ? Cela ne coûte rien.

Orientez dans la même direction tous les composants polarisés comme les condensateurs et les diodes (à moins que l'intégrité du signal ou une autre bonne raison empêche de le faire). C'est du temps gagné pendant l'assemblage de la carte, l'inspection et la recherche de panne. Indiquez toujours la polarité sur la sérigraphie.

Attention aux empreintes en miroir.

Restez sur la grille

Travaillez sur une grille autant que possible. J'aime placer les composants et faire le routage initial sur une grille de 50 mils. Quand les choses deviennent trop denses, je bascule sur une grille de 25 mils. Quand une carte dense est presque finie, je passe sur une grille de 5 mils pour y serrer les derniers seg-

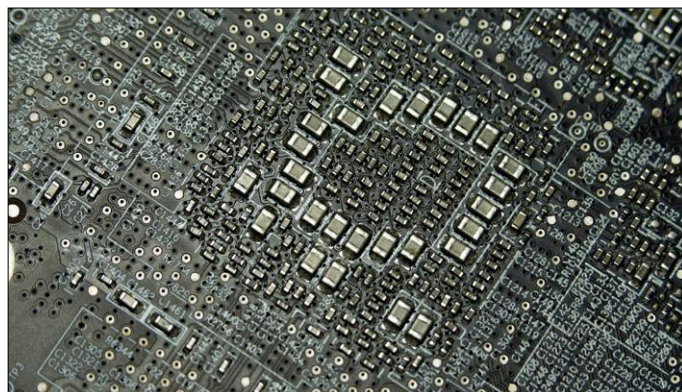


Figure 3. Pour bien découpler les circuits intégrés avec de nombreux contacts, les condensateurs de découplage peuvent être mis de l'autre côté de la carte. Notez la façon intuitive de placer les annotations des composants en dehors de la zone à densité élevée.

ments de piste. Parfois une grille de 1 mil peut être nécessaire pour placer précisément un élément tout en respectant les règles de conception (DRC). Mils, mm, peu importe l'unité choisie, mais il faut vous y tenir.

Les pastilles

Les contacts des composants et les bornes sont soudés à des pastilles sur le CI. Pour beaucoup de composants, les pastilles assurent non seulement la connexion électrique, mais aussi leur tenue mécanique. Il est donc important que la pastille soit assez grande pour ça. En particulier les composants hauts ou lourds avec peu de bornes (par ex. les gros condensateurs) nécessitent de grandes pastilles pour éviter d'être arrachés accidentellement. Le cuivre est collé sur la carte et, en particulier lorsqu'il est chauffé (par le soudage ou le courant), il y a un risque de délaminage.

Les composants CMS ont d'habitude des pastilles oblongues. La soudure sans plomb ne s'écoule pas aussi bien que la soudure 60/40 traditionnelle, en particulier dans les coins, c'est pourquoi on a commencé à utiliser des pastilles à coins arrondis. Les pastilles CMS doivent être assez grandes pour supporter suffisamment de pâte à souder pour la soudure correcte d'une

Métallisé : recouvert avec un matériau conducteur
Mil : un millième de pouce
Millimètre : un millième de mètre
Mire : marque spéciale sur un film, masque, carte, panneau, etc., pour aider l'alignement avec des caméras, pochoirs, machines, entre eux, etc.
OAR : **Outer Annular Ring**, anneau annulaire externe
Pastille : élément de CI pour connecter une broche
Pierre Tombale (effet) : relevage partiel ou total d'un CMS à deux contacts pendant la refusion
Pile : empilage ordonné de couches
Piste : connexion entre deux ou plusieurs contacts sur un CI

Plan : grande surface de cuivre ; nommé plan d'alimentation si connecté à la masse ou à la tension d'alimentation
Pochoir : masque pour appliquer la pâte à souder sur un CI
Push and shove : router une piste en poussant et écartant les éléments environnants du CI pour créer suffisamment d'espace pour la nouvelle piste
Réseau : connexion volontaire entre deux ou plusieurs contacts
Réserve : protection contre la soudure ; voir Masque de soudure
Routage : transformation des réseaux en pistes
Sérigraphie : couche non conductrice de symboles graphiques et de texte, habituellement blanche, nommé aussi impression des composants

TMS : Technologie de Montage en Surface, pour les composants dont les contacts ne sont pas insérés dans des trous
Top : couche supérieure de la pile de la carte, nommée aussi côté composant
Trace : voir Piste
TTT : Technologie à Trou Traversant pour les composants dont les contacts sont insérés dans des trous
Via : trou métallisé connectant des pistes entre deux ou plusieurs couches différentes
Via stitching : utilisation de multiples vias pour connecter un élément de cuivre sur une couche à un autre élément sur une autre couche ; souvent utilisé pour conduire la chaleur ou sur les cartes haute fréquence



Figure 4. Une sélection de pastilles traversantes disponibles dans Eagle, outil répandu de conception de CI.

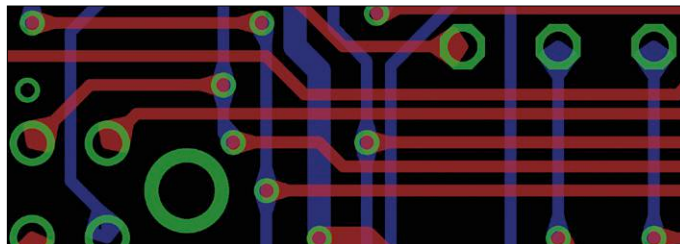


Figure 5. Pastilles effilées dans Eagle, créées avec le script goutte d'eau (ULP).

borne. Consultez la fiche technique du composant pour l'empreinte ou plage d'accueil à utiliser.

Les pastilles traversantes présentent de multiples formes (**fig. 4**). Les pastilles carrées sont souvent employées pour indiquer la broche 1 d'un composant tel un connecteur. Les pastilles octogonales sont répandues, mais les rondes sont mieux, car elles maximisent la surface de cuivre (bon pour la tenue mécanique comme pour la dissipation thermique), tout en minimisant la place requise pour respecter les règles d'espacement.

Il existe aussi des pastilles effilées ou en goutte d'eau, où la pastille se fond progressivement dans la piste connectée (**fig. 5**). En plus de donner une touche rétro à la carte, elles procurent des liaisons pastille-piste plus solides. C'est important pour les cartes qui doivent pouvoir être courbées, comme les CI souples. Tous les outils de conception de CI ne peuvent pas le faire.

Ensuite il y a le frein thermique (**fig. 6**). Si un contact d'un composant est soudé à une piste très épaisse ou à un aplat de cuivre, le cuivre entourant la pastille peut absorber la chaleur,

ce qui rend le soudage difficile. Pour éviter la dissipation de chaleur, les pastilles peuvent être connectées avec des pistes étroites – appelées rayons ou thermiques, d'habitude quatre – au cuivre environnant (c'est là que l'anneau interne « *Inner Annular Ring* », IAR, apparaît). Lorsqu'on fait de la soudure à refusion (c.-à-d. dans un four), le problème est moindre parce que toute la carte est chauffée à la même température. Ainsi il n'y a d'habitude pas de véritable raison d'utiliser les techniques de frein thermique pour les composants CMS qui seront soudés dans un four. Toutefois, l'effet « pierre tombale » – le relevage partiel ou complet d'un composant CMS pendant la refusion – peut survenir quand la masse thermique est très différente des deux côtés d'un petit composant à deux bornes (résistance ou condensateur). Un frein thermique règle ce problème. Les vias n'ont presque jamais besoin de frein thermique, car, en général, ils ne sont pas soudés.

Les trous

Les pastilles ont souvent des trous dedans – les vias aussi. Les trous de fixation peuvent ne pas être métallisés ; les trous

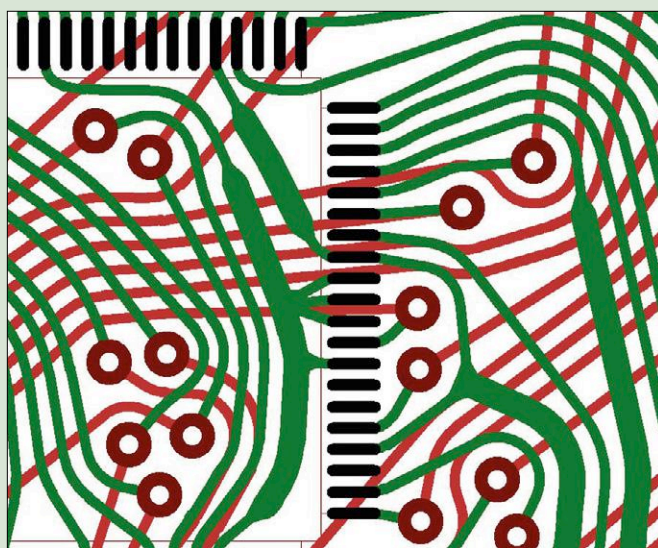
Routage manuel ou automatique ?

Les fabricants d'outils de CAO et les scientifiques passent et ont passé beaucoup, beaucoup, beaucoup d'heures à mettre au point et améliorer des routeurs automatiques et je n'ai encore jamais été satisfait des résultats d'un seul d'entre eux. Je n'ai non plus jamais rencontré quelqu'un qui l'était. Une raison est que je n'ai sans doute jamais eu accès aux meilleurs outils existants, mais ceux que j'ai essayés soit se plantaient soit abandonnaient cinq minutes après que j'aie quitté le bureau, n'ont jamais réussi à router 100% de la carte, ou s'ils atteignaient 100% je devais passer des heures à remettre au propre, ou étaient simplement trop compliqués à mettre en œuvre. Pour toutes ces raisons, je préfère router manuellement. (De plus, je trouve que le routage est une activité très relaxante).

Le routeur automatique parvient souvent à router jusqu'à 90% de la carte, voire mieux, mais n'atteint presque jamais 100% à moins d'un circuit très simple. Quand il se bloque avant d'atteindre 100% cela signifie d'ordinaire qu'il ne reste plus de solution pour finir la carte. Cela n'implique pas que la carte ne peut pas être routée, cela signifie simplement que le routeur a bloqué sur toutes les solutions possibles. Pour s'en sortir, vous devez tant défaire que finalement il aurait été plus rapide de router la carte manuellement depuis le début.

Comme compromis, certains utilisent le routeur automatique pour voir où les problèmes de routage peuvent survenir, puis déplacent quelques composants en espérant résoudre ces poten-

tiels goulots d'étranglement et finissent en routant manuellement. D'autres utilisent le routeur automatique sur des parties (simples) de la carte puis mettent au propre ensuite.



Le « routeur topologique » Eremex TopoR n'a pas de directions de routage préférentielles pour optimiser l'utilisation de l'espace. (Source : Eremex)

dans les pastilles et les vias sont métallisés. Vous spécifiez le diamètre du trou métallisé pour vos vias et pastilles ; c'est le travail du fabricant de CI de garantir que le trou terminé respecte ce diamètre. Pour un trou métallisé, le diamètre de perçage réel doit être plus grand que spécifié en raison de l'épaisseur de la métallisation. Le perçage a une précision limitée et un trou peut être décentré (**fig. 7**). En conséquence, pour garantir qu'assez de cuivre reste autour d'un trou métallisé après avoir percé et gravé la carte, l'anneau externe (« *Outer Annular Ring* », OAR) doit être suffisamment large. L'anneau interne existe aussi (« *Inner Annular Ring* », IAR). Selon mon dictionnaire, « *annular* » signifie « en forme d'anneau », donc, techniquement parlant, des « *annular rings* » sont des « *ring rings* » (ce qui me rappelle cette belle chanson d'Abba en 1973 dont je recommande la vidéo sur YouTube. On ne fait plus de musique comme ça..., ni de tenues, *Swenglish*, guitares, coupes de cheveux, etc., mais je m'égare).

Spécifier une mauvaise taille de perçage pour un contact de composant – trop grande ou trop petite – est une erreur fréquente.

Il y a un problème avec les vias et la soudure à la vague : si un via n'est pas masqué, la soudure peut couler au travers et potentiellement endommager un composant monté au-dessus. Fermer le masque de soudure pour les vias évitera cela en général.

Pistes et plans

Garder les pistes aussi courtes que possible : c'est évident. C'est particulièrement vrai pour les signaux haute fréquence, mais les signaux basse fréquence et même les signaux continus bénéficient aussi des pistes courtes. Les pistes courtes sont bonnes non seulement pour les signaux, mais aussi pour économiser de la place sur la carte. Certains signaux rapides peuvent nécessiter des pistes (ou une paire de pistes) d'une longueur (et impédance) précise ce qui n'est pas toujours la plus courte connexion possible.

N'utilisez pas le stylet le plus fin de votre outil de conception de CI. Les pistes étroites peuvent coûter plus et sont fragiles. Le détail le plus fin de la carte détermine la classe de fabrication de la carte ; plus la classe est haute, plus la carte est chère. Les vibrations peuvent créer des fissures dans les pistes étroites et dégrader, voire couper, des connexions. Si la carte doit être soudée à la main, ou si on prévoit des reprises, les pistes étroites peuvent aisément se délaminer quand on les chauffe trop, trop longtemps. La gravure à domicile n'est généralement pas bien contrôlée et les pistes étroites peuvent être rongées. De plus, une piste étroite peut ne pas supporter l'intensité de courant prévue. Adaptez la largeur de la piste au courant transporté, un peu plus pour la maintenir froide. De nombreux calculateurs en ligne vous aideront à déterminer la meilleure largeur pour votre piste. Une piste étroite transportant trop de courant chauffera et pourra fondre ou se casser ; c'est la chaleur qui limite le courant. Les vias doivent aussi supporter le courant. Recouvrir les pistes avec une épaisse couche de soudure augmentera le courant maximal accepté par la piste. De même, remplir un via avec de la soudure ou du cuivre pourra aussi aider. Ménager des ouvertures dans le masque de soudure sera utile dans ce cas. Plusieurs vias en parallèle sont souvent utilisés pour améliorer la conduction de courant, en limitant le risque de circuit ouvert dû à un via coupé. Bien sûr, un via plus grand peut aussi être utilisé.

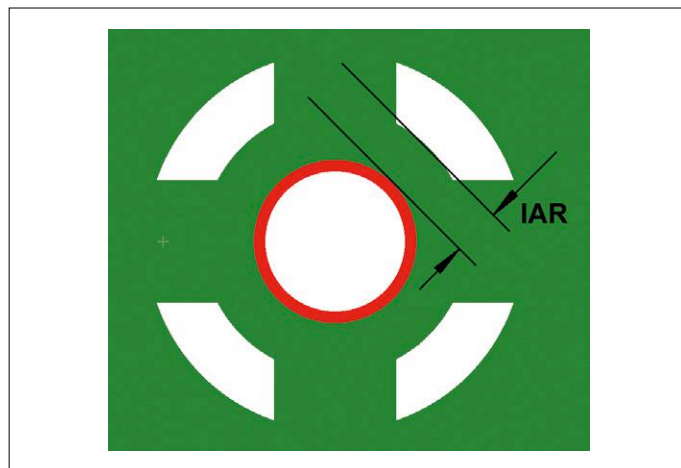


Figure 6. Le frein thermique empêche que trop de chaleur ne s'échappe dans les grands aplats de cuivre pendant le soudage. Ici, l'anneau interne minimal peut être spécifié même si c'est presque le même que l'externe. L'anneau rouge représente la métallisation du trou.

Nommer les réseaux est utile pendant le routage, car cela vous rappelle sur quel signal vous travaillez.

Maintenant, les coins ! Les pistes arrondies sont meilleures parce que, contrairement à celles avec des angles, elles ont une largeur constante. Les changements de largeur de piste provoquent des inadaptations d'impédance et des réflexions, mais seulement à très haute fréquence, donc la plupart des montages ne sont pas affectés par ces problèmes. On considère les angles droits affreux, pourtant vous en trouverez beaucoup en regardant de près les vias sur une carte. Souvent une piste sur une face continue à 90° sur une autre, et même si elle ne le fait pas, le via lui-même introduit deux angles de 90°. Les pistes avec des angles à 90° ont tendance à être plus longues que celles avec des angles de 45°. Les angles aigus peuvent aussi causer des problèmes de délaminage, des poches acides, ou un défaut de gravure. Quelquefois il est impossible d'éviter un angle aigu et dans la plupart des cas ça marche. Personnellement j'essaie de m'en tenir à des angles de 45°.

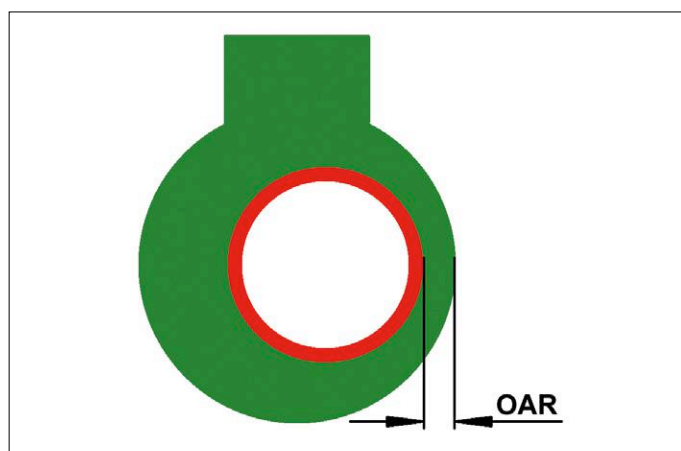


Figure 7. À cause des tolérances de fabrication, un trou peut être décentré. Pour permettre la métallisation (l'anneau rouge), le trou doit être percé plus grand que spécifié. Le trou terminé (avec métallisation) aura le diamètre spécifié. L'anneau qui reste après perçage est l'anneau externe. L'anneau externe minimal acceptable montré ici est spécifié dans les règles de conception.

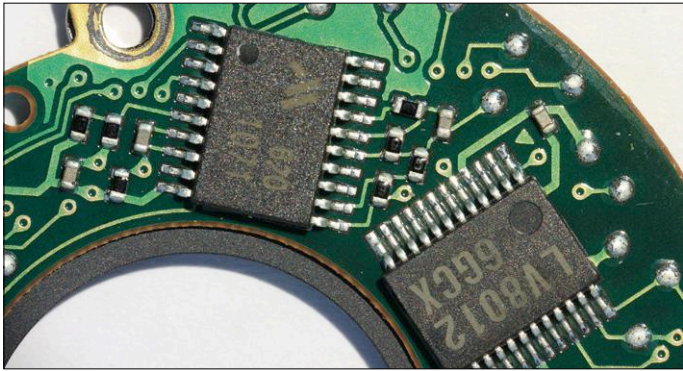


Figure 8. Pas d'annotations de composants, uniquement le marquage de la broche 1, pistes avec toutes sortes d'angles et vias en goutte d'eau. Une fois que vous savez ce que vous faites, c'est une *pratique répandue* d'être tolérant avec la *bonne pratique*.

L'utilisation d'un plan d'alimentation ou d'aplat de cuivre (habituel pour la masse) économisera beaucoup de travail et sera bon du point de vue CEM. Cependant, attention quand vous utilisez plusieurs plans d'alimentation, en particulier proches des bords de la carte et des trous de fixation où des objets métalliques et vis d'assemblage pourraient créer des courts-circuits involontaires entre les plans. Finissez toujours par une vérification des règles électriques (ERC) pour vous assurer que tous les courts-circuits et circuits ouverts ont été corrigés.

Quand vous utilisez un plan ou un aplat pour un réseau, ne routez pas ce réseau, mais laissez l'aplat de cuivre le faire pour vous. Attention aux connexions très fines, comme involontaires, entre des sections de plan, qui peuvent être rongées et couper la continuité du plan. De tels plans sont inadaptés pour la gravure à domicile. Pour empêcher cela, commencez avec un espacement large pour les aplats de cuivre et voyez si le cuivre s'étale partout. Déplacez les pistes et les vias pour améliorer l'écoulement du cuivre. Quand vous êtes satisfait de l'aplat, diminuez l'espacement.

Notez que les plans ont une influence sur la distribution du cuivre de la carte. Quand le cuivre n'est pas réparti régulièrement, la gravure peut devenir irrégulière, et entraîner une répartition irrégulière de l'épaisseur du cuivre pouvant courber le CI aux températures extrêmes (comme dans les applications automobiles).

Appliquez les règles de conception

Utilisez la vérification des règles de conception (DRC) et des règles électriques (ERC) pour vous assurer que tous les éléments du CI les respectent. De bonnes règles de conception aident à éviter que des pistes ne se chevauchent ou passent trop près des bords de la carte, à trouver des diamètres de perçage et espacements incorrects, et bien plus. Utilisez-les ! Lorsque vous pensez que la carte est prête, assurez-vous que tous les réseaux sont bien connectés. Ne laissez pas d'avertissements ni d'erreurs, même s'ils sont acceptables pour vous parce qu'ils pourraient perturber la personne qui héritera du projet – ou vous-même lorsque vous le redécouvrirez six mois plus tard. Si vous devez tolérer un avertissement, documentez pourquoi. Si possible changez la règle de conception qui vous gêne pour le faire disparaître, mais assurez-vous que la nouvelle règle est acceptable pour votre application, de même que pour votre fabricant de carte.

Marquages et impression de composants

Essayez de mettre des annotations pour tous les contours de composants et assurez-vous qu'ils demeurent visibles après le montage. J'aime orienter les annotations dans le même sens que les composants, en lignes et colonnes (voir aussi figure 3). Indiquez la broche 1 sur tous les connecteurs, barrettes, et tous les composants où elle n'est pas clairement identifiable. Marquez aussi la polarité de tous les composants polarisés comme les condensateurs et les diodes (**fig. 8**).

Évitez les marquages sous les composants CMS, en particulier ceux à deux plots, car cela rend leur assise bancale et peut conduire à l'effet « pierre tombale ».

Gardez le texte lisible et utilisez des étiquettes pour guider l'utilisateur quand c'est utile. Notez que les ouvertures dans le masque de soudure sont d'habitude légèrement plus grandes que les pastilles, et que chaque sérigraphie traversant ces ouvertures est coupée, donc assurez-vous que le texte ne les chevauche pas. Mettez un texte lisible sur toutes les couches pour éviter un retournement accidentel d'une couche. Étiquetez ou numérotez les couches pour garantir qu'elles seront empilées dans le bon ordre. Donnez à la carte un nom ou un numéro unique et n'oubliez pas d'indiquer sa version.

Si le dessous de la carte peut aussi être sérigraphié, faites-le. Les cartes FLTM n'ont pas de sérigraphie en général, mais cela ne signifie pas que vous devez oublier le texte et le marquage de polarité ; vous pouvez aussi écrire sur les couches de cuivre. Ainsi, quand c'est possible, placez des marquages et du texte tout en faisant attention à leur taille pour éviter qu'ils ne soient perdus pendant la gravure.

Test

Prévoyez des points de test accessibles. Si possible faites que le montage produise des signaux de test utiles. Les broches des composants ne doivent pas servir de points de test, car une sonde de test appuyée contre une broche peut provisoirement « réparer » un mauvais joint de soudure, feignant une bonne connexion, mauvaise le reste du temps. Les vias peuvent être utiles pour le test manuel, mais seulement quand ils sont laissés ouverts dans le masque de soudure.

C'est tout ?

Après la lecture de ce long article, vous pourriez penser que vous en savez beaucoup sur la conception des CI, alors qu'en réalité nous n'avons qu'effleuré le sujet. Nous n'avons pas parlé, pour ne citer que cela, des chemins de retour de courant, des plans de masse multiples, de la gestion de la chaleur pendant le soudage et en fonctionnement, de la conformité avec la CEM, du traitement des signaux haute fréquence, du dessin des empreintes, etc.

La conception de CI est un vaste thème qui combine la chimie, la physique, l'électronique, la mécanique et l'automatisation. Il est plutôt amusant de découvrir que beaucoup d'aspects de la conception de CI n'ont jamais vraiment été étudiés en détail et sont simplement fondés sur du bon sens et des hypothèses. Vous trouverez sur l'internet beaucoup de discussions sur la conception des CI, donc dans le doute, demandez autour de vous. ◀

Remerciements à Malte Fischer pour ses conseils utiles.

(160397 – version française : Denis Lafourcade)

Lien

[1] www.elektormagazine.fr/160397

la communauté d'Elektor

DÉCOUVRIR CRÉER PARTAGER

82

pays

248153

membres actifs

1040

experts & auteurs

489

publications

235332

visiteurs (par mois)

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 étonne.



Boutique en ligne d'Elektor : la caverne d'Ali Baba des électroniciens ! 10 % de remise permanente pour les membres GREEN et GOLD.
www.elektor.fr



Magazine Elektor : six numéros par an d'électronique inédite (projets, bancs d'essais, cours, trucs et astuces, actualité...)
www.elektormagazine.fr



Elektor PCB Service : gravure à la demande de vos circuits imprimés, en un ou plusieurs exemplaires. www.elektorpcbsevice.com



Elektor GRATUIT, version numérique : lettre d'information à l'affût des nouveautés. C'est gratuit : www.elektor.fr/inscription



Elektor Academy : l'apprentissage par la pratique au travers de webinaires, séminaires, ateliers, DVD... www.elektor.fr



Livres & DVD Elektor : Arduino, Raspberry Pi, microcontrôleurs, Linux et bien d'autres. 10 % de remise pour les membres !
www.elektor.fr



Elektor TV : présentations, comptes-rendus, déballage et évaluation de produits. Pour y voir plus clair.
www.youtube.com/user/ElektorIM



Elektor Labs : l'incubateur où partager ses projets et participer à ceux des autres. Nous faisons grandir vos idées !
www.elektormagazine.fr/labs

connectez-vous à notre communauté

formule

GREEN 92,50 € par an

- ✓ accès à l'archive d'Elektor
- ✓ 10% de remise dans l'e-shoppe
- ✗ 6x magazine imprimé
- ✓ 6x magazine numérique
- ✓ des offres exclusives
- ✓ accès à plus de 1 000 fichiers Gerber
- ✗ le DVD annuel d'Elektor

connectez-vous

www.elektor.fr/formule-greencard

formule

GOLD 127,50 € par an

- ✓ accès à l'archive d'Elektor
- ✓ 10% de remise dans l'e-shoppe
- ✓ 6x magazine imprimé
- ✓ 6x magazine numérique
- ✓ des offres exclusives
- ✓ accès à plus de 1 000 fichiers Gerber
- ✓ le DVD annuel d'Elektor

connectez-vous

www.elektor.fr/formule-goldcard

formule

GRATUITE

- ✗ accès à l'archive d'Elektor
- ✗ 10% de remise dans l'e-shoppe
- ✗ 6x magazine imprimé
- ✗ 6x magazine numérique
- ✓ des offres exclusives
- ✗ accès à plus de 1 000 fichiers Gerber
- ✗ le DVD annuel d'Elektor

connectez-vous

www.elektor.fr/inscription



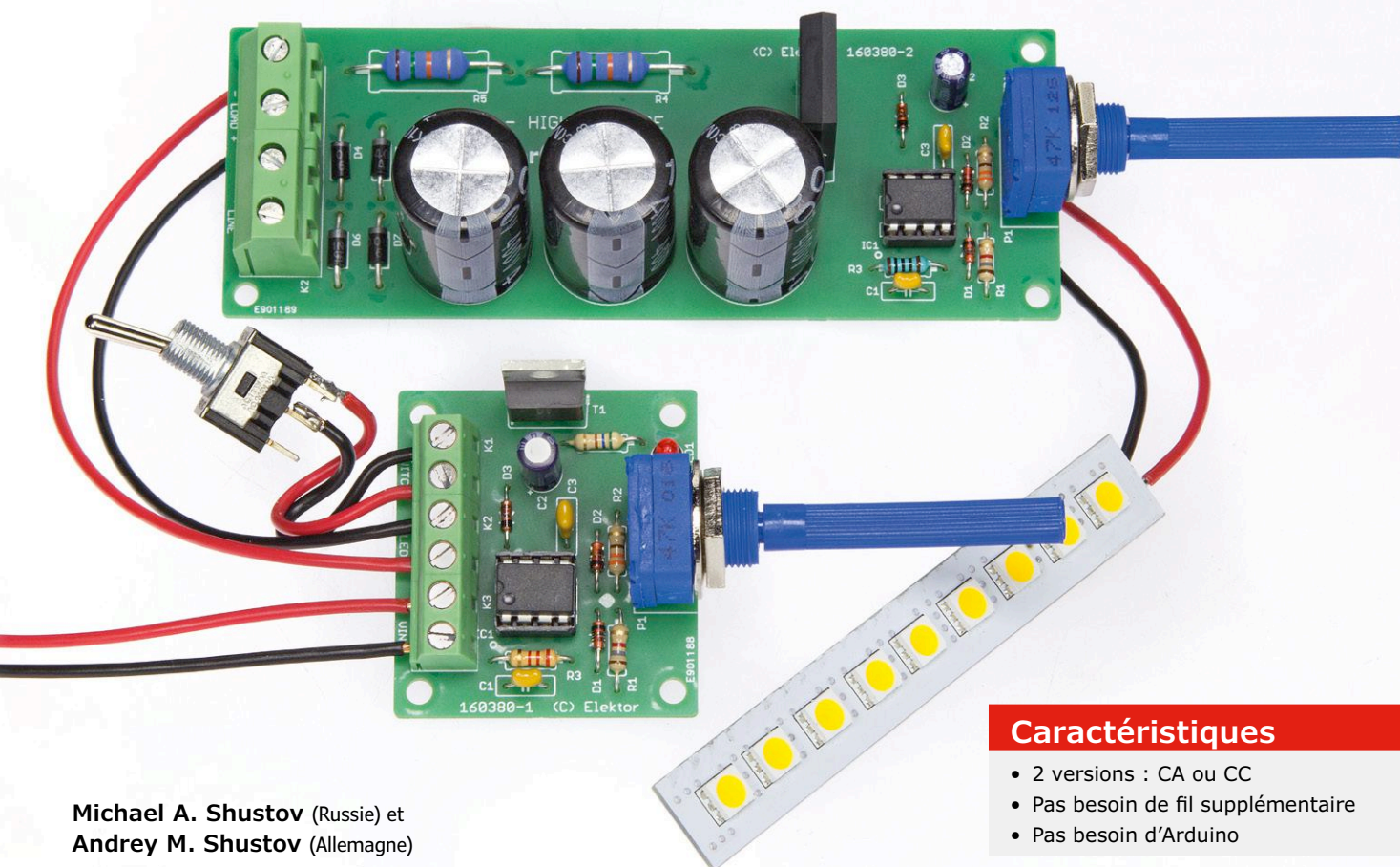
facebook.com/ElektorFR



twitter.com/Elektor

gradateur à deux bornes

commande de LED, lampes et chauffages



Michael A. Shustov (Russie) et
Andrey M. Shustov (Allemagne)

Caractéristiques

- 2 versions : CA ou CC
- Pas besoin de fil supplémentaire
- Pas besoin d'Arduino

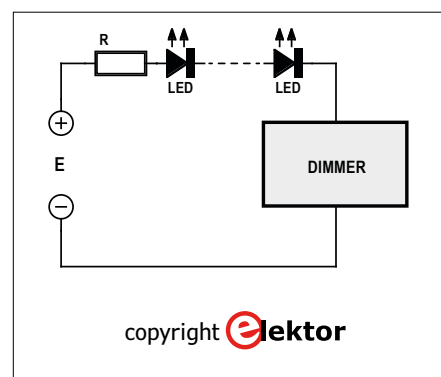
Par le passé, les rhéostats (aujourd'hui aussi appelés potentiomètres) étaient connectés en série avec une charge pour régler l'intensité du courant à travers la charge. Bien que simple, cette technique a des inconvénients.

Le premier de tous est que le courant ne peut être piloté que quand la charge est active. Deuxièmement le rhéostat chauffe, ce qui produit des pertes. De plus, régler l'intensité du flux lumineux de diodes électroluminescentes (LED) avec un rhéostat est quasiment impossible, à cause de la caractéristique courant-tension non linéaire des LED. La modulation de largeur d'impulsion (MLI) est meilleure et plus efficace pour faire varier en douceur l'intensité d'appareils

de chauffage ou d'éclairage (lampes incandescentes et LED), ou commander la vitesse d'un moteur.

Les gradateurs qui pilotent l'intensité des sources lumineuses sont habituellement connectés en parallèle avec l'alimentation, ce sont des appareils à trois bornes. Deux bornes sont connectées à

Figure 1. Le gradateur est en série avec la/les LED et ne nécessite que deux fils.



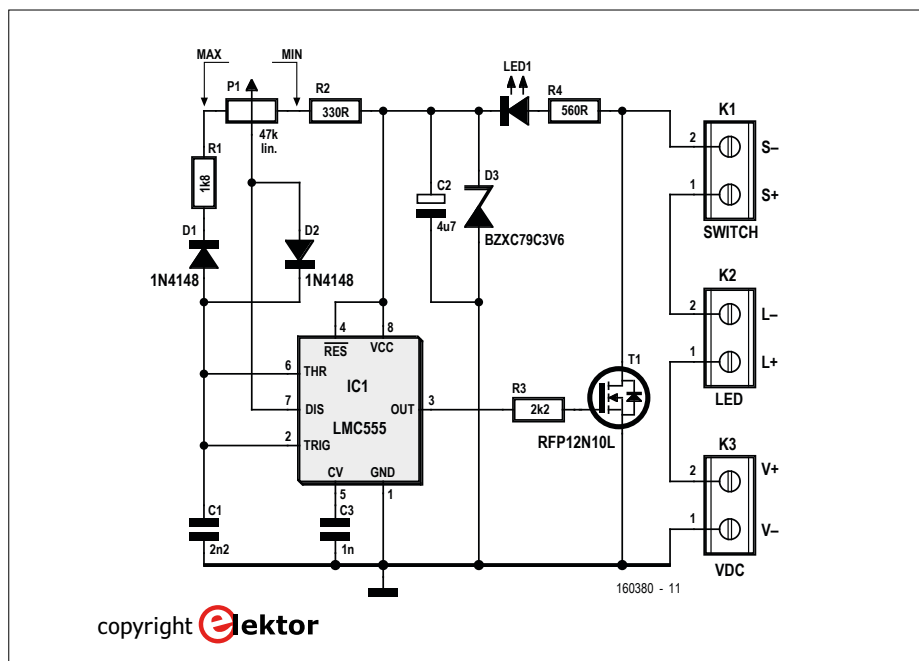


Figure 2. La version CC du gradateur série.

l'alimentation tandis que la troisième est connectée à la charge. Ces gradateurs ne peuvent pas prendre la place d'un interrupteur bipolaire, car ils doivent être raccordés à un 3^e fil d'alimentation. Notre gradateur à deux bornes lui (**fig. 1**) peut remplacer un interrupteur et être connecté en série avec une charge (LED, lampes ou éléments chauffants). Contrairement à un interrupteur, le gradateur permet d'ajuster le courant de charge de presque 0 à env. 97 % de sa valeur maximale. Lorsque le gradateur est réglé sur son minimum, il circule quand même un courant de très faible intensité à travers la charge.

Un inconvénient des gradateurs à deux ou trois bornes est la consommation résiduelle de courant, toutefois elle est insignifiante comparée à celle quand la charge est alimentée.

Les gradateurs décrits ci-dessous peuvent être utilisés avec des appareils d'éclairage domestiques, des thermostats ou encore des éclairages secondaires ou de secours.

Gradateur à deux bornes alimenté en CC

Le cœur du gradateur (**fig. 2**) est un bon vieux temporisateur 555 dans sa version CMOS basse consommation, appelée LMC555CN. Il est câblé en générateur d'impulsions à rapport cyclique variable, un circuit bien connu, rien de nouveau sous le soleil. La sortie d'IC1

pilote le MOSFET de puissance T1 qui « allume » et « éteint » la charge (LED) suivant le signal MLI. Le signal MLI a une fréquence approximative de 6 kHz, trop rapide pour que l'œil humain le perçoive. Le ratio impulsion/repos ou rapport cyclique est ajusté avec le potentiomètre P1. Les résistances R1 et R2 déterminent les valeurs minimale et maximale de la largeur d'impulsion. Jusqu'ici le gradateur est un régulateur *shunt* à trois bornes standard. L'astuce pour se débarrasser d'un bornier, c'est d'alimenter la charge **et** le circuit avec la même source. Quand T1 est bloqué, on charge C2 via R4 et LED1. La diode zener D3 limite la tension à 3,6 V. La LED1 sert de témoin de fonctionnement et permet aussi de trouver plus facilement le gradateur dans le noir. Le transistor T1 peut supporter jusqu'à 100 V et conduire 12 A, mais pas en même temps (le circuit imprimé ne supporte que 2,5 A !). D'après sa feuille de caractéristiques, il peut dissiper jusqu'à 60 W et il doit être monté sur un dissipateur adapté quand il doit supporter plus de 1 W. R4 détermine la tension d'alimentation maximale, ici 20 V. Avec les valeurs données dans le schéma, nous mesurons un courant de repos de 3,5 mA (c'est-à-dire quand le rapport cyclique est de 0%). C'est un courant suffisant pour que certaines LED s'allument de façon notable. Réduire la valeur de R2 de 75-100 Ω devrait améliorer la situation. De plus, augmenter la valeur

de R5 réduira aussi le courant minimal. Un interrupteur peut être connecté à K1 pour éteindre complètement le gradateur.

Gradateur alimenté sur secteur (CA)

Il est bien sûr possible d'adapter le gradateur pour l'utiliser avec une source alternative comme le 230 V CA du secteur disponible dans les maisons de nombreux pays à travers le monde (ça fonctionnera aussi avec du 115 V CA). Il faut franchir deux obstacles pour cela :

- Trouver un transistor de puissance qui résiste à des hautes tensions
- Convertir la tension CA en tension CC.

Le second point nécessite de l'attention. Redresser la tension secteur ne suffit pas, les grosses charges comme les ampoules à incandescence scintilleront, ce qui rend le filtrage obligatoire. De plus, la tension redressée du secteur vaut environ 325 V (163 V pour une tension secteur de 115 VCA), ce qui est trop élevé pour des ampoules domestiques standard (néanmoins, il est possible de monter en série plusieurs ampoules identiques de tension inférieure). À cause de ces limitations pratiques, ce qui suit n'est là que pour illustrer nos propos ; nous vous déconseillons de l'utiliser dans la vraie vie.

La **figure 3** montre la version 230 V CA du gradateur. Le changement le plus notable est l'ajout du redresseur (D4-D7) et du filtre d'alimentation constitué

INFOS SUR LE PROJET

LED
éclairage
Danger !

MLI
gradateur
555

débutant
→ **connaisseur**
expert

Environ 2 h

Fer à souder,
du bon sens

15 €

courant maximal à travers la ou les LED. Consultez la feuille de caractéristiques de la LED pour les valeurs exactes de U_{LED} et I_{LED} . La puissance de la résistance peut être calculée avec

$$P_R = (I_{LED})^2 \times R \quad [W]$$

Exemples

- (Figure 2) Prenons $n = 1$, $U_{LED} = 3 V$, $I_{LED} = 0,05 A$, $U_T = 0,04 V$, $U_{SUP} = 20 V$. Alors $R = 377 \Omega$, ou arrondie à sa valeur standard proche supérieure, 390Ω . La résistance doit être capable d'absorber $(0,05)^2 \times 390 = 0,975 W$, ou arrondie à une valeur commune, $1 W$.
- (Figure 2) Prenons $U_{LED} = 2 V$, $I_{LED} = 0,02 A$, $U_T = 0,04 V$, $U_{SUP} = 16 V$. Si nous limitons $n \times U_{LED}$ à 80% de U_{SUP} , alors $n = 6$. Dans ce cas R devient 220Ω , et sa puissance devrait être supérieure à $88 mW$ ($0,125 W$ serait une valeur standardisée suffisante).
- (Figure 3) Prenons $U_{LED} = 2,5 V$, $I_{LED} = 0,05 A$, $U_T = 3 V$, $U_{SUP} = 325 V$. Si nous limitons $n \times U_{LED}$ à 80% de U_{SUP} , alors $n = 104$ et $R = 1,378 \Omega$. La valeur standardisée la plus proche est $1,5 k\Omega$. Sa puissance devrait être supérieure à $3,75 W$. La résistance

Tableau 1. Tension de seuil en fonction de la couleur émise par une LED.

Couleur	Longueur d'onde [nm]	$U_{LED} [V]$
Infrarouge	≥ 760	$\leq 1,6$
Rouge	610 – 760	1,6 – 2,0
Orange	590 – 610	2,0 – 2,1
Jaune	570 – 590	2,1 – 2,2
Vert	500 – 570	2,2 – 2,5
Bleu	450 – 500	2,5 – 2,7
Violet	400 – 450	2,7 – 3,1
Ultraviolet	≤ 400	$\geq 3,1$
Blanc	–	3 – 3,7

requis peut être obtenue en connectant en parallèle deux résistances de $3 k\Omega / 2 W$. Avec ces valeurs, le courant à travers les LED devient $(U_{SUP} - n \times U_{LED} - U_T) / 1500 = 0,041 A$, ce qui signifie que T1 doit dissiper $U_T \times I_{LED} = 0,125 W$. Il devrait le faire sans dissipateur.

Les auteurs de cet article ainsi qu'Elektor ne peuvent en aucun cas être tenus responsables, pour autant que cela soit juridiquement possible, des dommages dus à l'utilisation des circuits présentés dans cet article. ◀

(160380 – version française : Alexandre Roy)

Lien

[1] www.elektormagazine.fr/160380

Derniers mots

Nous ne vous le répéterons jamais assez : le **circuit de la figure 3 est dangereux**. Il présente des tensions mortelles et peut vous tuer (ou quelqu'un d'autre) quand il est utilisé sans précautions adaptées. **Ne fabriquez pas ce circuit !** Ne l'utilisez pas non plus (même si ce n'est pas vous qui l'avez construit).

DANS L'E-CHOPPE

→ 160380-1

Circuit imprimé
pour la version CC



LISTE DES COMPOSANTS

Résistances :

R1 = 1,8 k Ω
R2 = 330 Ω
R3 = 2,2 k Ω
R4 = 560 Ω
P1 = 47 k Ω , potentiomètre linéaire

Condensateurs :

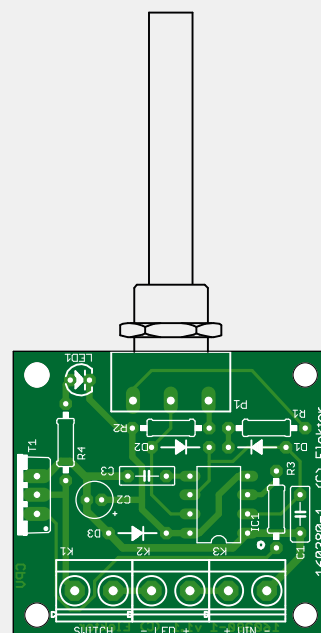
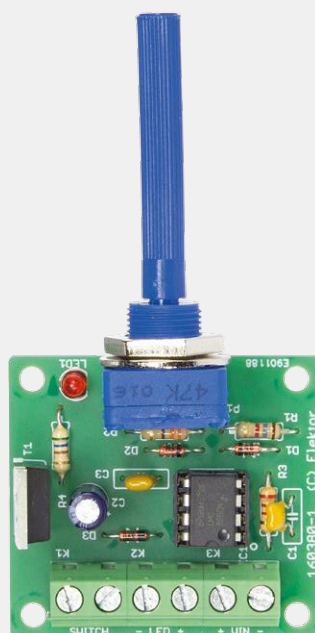
C1 = 2,2 nF, pas de 5,08 mm
C2 = 4,7 μF , pas de 2 mm
C3 = 1 nF, pas de 5,08 mm

Semi-conducteurs :

D1,D2 = 1N4148
D3 = BZX79C3V6
IC1 = LMC555CN
LED1 = rouge, 3 mm
T1 = RFP12N10L

Divers :

K1,K2,K3 = bornier à vis à 2 voies pour circuit imprimé, pas de 5,08 mm
Circuit imprimé, réf. 160380-1





atténuateur HF commutable

niveaux de signal précis
par atténuation réglable

Alfred Rosenkränzer (Allemagne)

Si l'on n'a que très rarement affaire à la haute fréquence, il est utile de pouvoir atténuer facilement le niveau de ces signaux par pas discrets. On peut alors, par exemple, ajuster le niveau de sortie d'un générateur HF à d'autres valeurs que celles fournies par le réglage du générateur. Un atténuateur est très utile pour le réglage ou la réparation des récepteurs ainsi que pour les émetteurs. Les atténuateurs professionnels sont malheureusement très chers, ce qui rend une réalisation personnelle intéressante là où une électronique coûteuse ne s'impose pas.



Caractéristiques

- Atténuateur HF avec 6 inverseurs pour 0 à 31 dB par pas de 1 dB

Il est quelquefois utile et même nécessaire de pouvoir atténuer un signal HF d'une manière précise. Si le signal est fourni par une source HF dont le niveau est difficilement réglable, il n'est guère possible de se passer d'un atténuateur additionnel. Cet atténuateur est d'ordinaire inséré dans une ligne HF. C'est utile, par exemple, si l'on veut évaluer la sensibilité d'un récepteur ou son comportement aux signaux faibles. On insère alors l'atténuateur dans la ligne d'antenne et l'on augmente l'atténuation jusqu'à ce que la réception se dégrade ou que les phénomènes à analyser (comme un souffle ou un décrochage du contrôle automatique de fréquence) se manifestent. Pour l'électronicien HF, les cas d'utilisation d'un bon atténuateur sont évidents.

Les fabricants d'appareils de mesure proposent des atténuateurs de bonne qualité (fig. 1), mais à un prix auquel tout élec-

tronicien n'a pas les moyens ou le désir de consentir, alors que l'appareil n'est composé que d'un boîtier, de bornes, de commutateurs et de quelques résistances. En y mettant un peu de soin, la réalisation maison d'un bon atténuateur pour les domaines de fréquences utiles à un prix raisonnable est donc possible.

Circuit d'atténuation

Pour l'atténuation d'un signal à impédances constantes (les impédances d'entrée et de sortie sont conservées, seul le niveau du signal varie), il existe en principe deux circuits équivalents composés de trois résistances : les circuits en T et en Pi (fig. 2). Pour mon circuit, j'ai choisi



Fig. 1. Atténuateur VHF typique de la société HP. Depuis longtemps plus disponible.

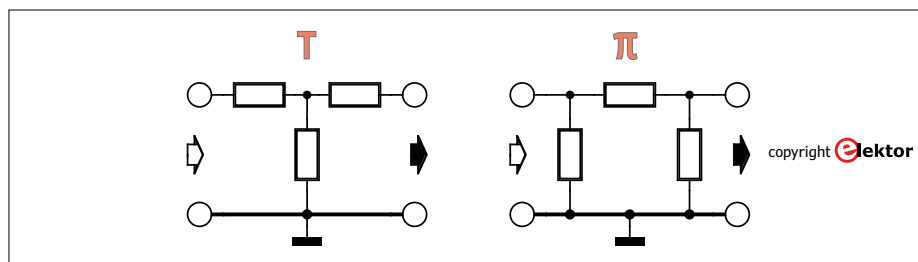


Fig. 2. Configurations de base T et Pi.

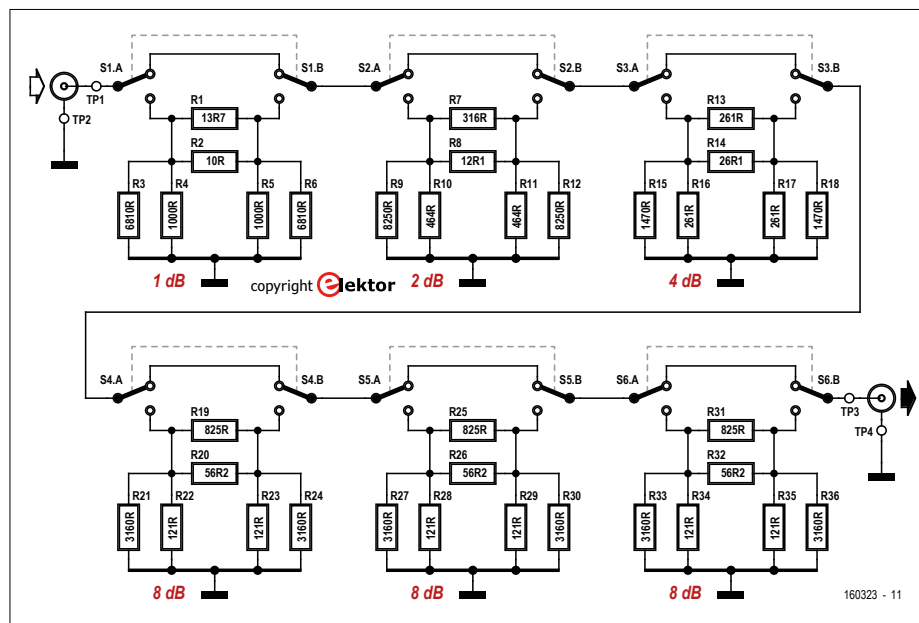


Fig. 3. Schéma complet de l'atténuateur à six étages.

la variante Pi, qui facilite la conception du circuit imprimé. Dans le cas de la HF, le circuit imprimé est préférable à un câblage volant, car a) les paramètres géométriques sont les mêmes pour tous les étages et b) les inductances et les capacités parasites restent faibles. La **figure 3** montre le schéma complet

de l'atténuateur à six étages. Il s'agit tout simplement de six atténuateurs Pi en série, insérés ou supprimés par des inverseurs bipolaires. La seule spécificité est que chaque résistance du circuit de base est remplacée par deux résistances en parallèle. De cette manière, la résistance calculée peut être réalisée

avec une bonne approximation avec des résistances des séries E. Il faut donc en tout six inverseurs et (jusqu'à) $6 \times 3 \times 2 = 36$ résistances.

L'atténuation désirée est obtenue en jouant sur les valeurs des résistances. Théoriquement, on peut réaliser une valeur entre 0 dB et l'infini. Dans la pratique, l'atténuation est limitée à 10 dB. Les valeurs supérieures sont perturbées par les effets parasites des composants, du circuit imprimé et des inverseurs qui ne sont pas spécialement conçus pour les hautes fréquences. La valeur minimale en pratique est de 1 dB, quoique des valeurs d'une fraction de dB soient possibles. Si un inverseur est en position haute, le signal est transmis sans atténuation, en position basse, il est atténué. Les valeurs d'atténuation des différents étages sont indépendantes, car les impédances d'entrée et de sortie sont partout les mêmes. Pour mon circuit, j'ai choisi les atténuations 1 | 2 | 4 | 8 | 8 | 8 dB, ce qui permet de régler l'atténuation totale par pas de 1 dB (environ 10%) dans le domaine 0 à 31 dB. Ce choix convient à mes applications habituelles, mais d'autres combinaisons sont possibles. On peut tout aussi bien envisager 6 fois 10 dB pour une excursion de 60 dB par pas de 10 dB, ou bien une combinaison de 0,5 | 1 | 2 | 4 | 8 | 8 dB pour 0 à 23,5 dB par pas de 0,5 dB.

Calculs

Pour le calcul des résistances, j'utilise le programme AADE, qu'on peut télécharger sur [1]. Normalement, on se sert de ce programme pour des calculs plus compliqués, tels que celui d'un filtre, mais il convient aussi pour les atténuations et épargne ainsi des tas de calculs.

Dans le menu *Design@Attenuator Pad*, on commence par choisir la structure Pi, puis dans le menu suivant l'impédance de ligne. En principe, on peut choisir n'importe quelle valeur, mais aujourd'hui le choix est limité à 50 Ω pour les appareils de mesure et 75 Ω pour les antennes et les signaux vidéo analogiques. Le circuit imprimé est conçu pour 50 Ω , mais il est réalisable pour 75 Ω en jouant sur la largeur des pistes et l'épaisseur de la platine. Pour cela, des logiciels sont disponibles sur l'internet. Si vous en avez l'intention et les compétences, vous pouvez examiner de plus près l'outil proposé sous [2].

L'atténuation souhaitée est saisie dans le menu suivant et l'on obtient un cir-



LISTE DES COMPOSANTS

Résistances :

36x 0603 ou 0805, valeurs : voir le texte

Autres :

S1 à S6 : inverseur bipolaire encartable, TL 46 PO (par ex. Reichelt)
Boîtier en aluminium extrudé, Hammond 1550Z102 (par ex. Reichelt)
2 bornes HF au choix (par ex. BNC pour montage à un trou)
Circuit imprimé (voir texte)

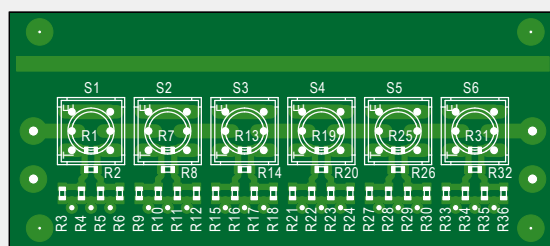


Fig. 4. Dessin du circuit imprimé étudié pour la HF.

cuit avec les valeurs de résistances résultantes. Malheureusement elles ne correspondent généralement pas aux valeurs de la série E. Même la série E96 ne fournit pas toujours une solution satisfaisante. Pour obtenir une valeur aussi proche que possible de celle fournie par AAE par la mise en parallèle de deux résistances de la série E24 standard du commerce, on peut, bien sûr, procéder par calcul mental, martyriser sa calculatrice, créer un tableau Excel ou écrire un petit programme. Mais il est plus simple d'utiliser des outils en ligne qui vous déchargent du travail et vous fournissent, avec les solutions possibles, les pourcentages d'erreur. Un bon outil est disponible sous [3]. Une précision de 1% est suffisante.

Si vous vous demandez pourquoi aucune variante en T n'est proposée ni pour le schéma ni pour le circuit imprimé, la réponse est que cela aurait compliqué le circuit (et dégradé les performances HF).

Sans circuit imprimé, rien à faire

Le circuit imprimé que j'ai conçu est représenté sur la **figure 4**. Les fichiers Eagle correspondants sont disponibles au téléchargement gratuit sur la page associée à cet article sur le site d'Elektor [4]. Les inverseurs sont disposés au milieu, avec, en dessous, les six résistances associées, reliées au plan de masse de la face inférieure par trois vias. La largeur des pistes, associée à l'épaisseur de 1,0 mm de la platine FR4 (important !), donne une impédance de ligne de 50 Ω environ. Les résistances CMS sont de format 0603, mais vu la bonne taille des pastilles, on peut aussi bien utiliser des composants 0805. La puissance de l'atténuateur dépend directement de la puissance des résistances, mais sa conception le destine plutôt aux faibles puissances : il est préférable de rester en dessous des 100 mW. Par rapport au prototype, on a ajouté au circuit une structure de test permettant de mesurer l'impédance de ligne. Les bornes SMA sont facultatives et ne sont pas utilisées en dehors de cette mesure.

Le circuit imprimé trouve place dans un boîtier en aluminium extrudé de la société Hammond (voir la liste des composants). Pour l'entrée et la sortie, on utilise des bornes HF (au labo, en général de type BNC). Pour le montage des composants, rien de particulier, si ce n'est que pour les résistances au for-

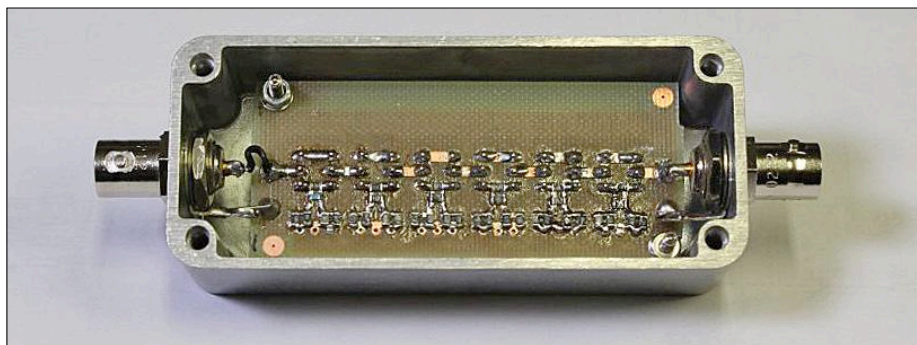


Fig. 5. Le circuit installé dans le boîtier Hammond.



Atténuer des signaux HF pas à pas



Fig. 6. Le prototype terminé.

mat 0603, il faut de bonnes pincettes et une main ferme. Après le montage, il est facile de tester le circuit en n'activant à chaque fois qu'un seul étage d'atténuation (les autres étant sur la position 0 dB) et en mesurant la résistance globale successivement à l'entrée et à la sortie. Il faut, bien sûr, connecter une résistance de 50 Ω (ou deux de 100 Ω en parallèle) de l'autre côté. Si l'on trouve toujours 50 $\Omega \pm <1\%$, on peut être raisonnablement assuré qu'on n'a pas permuté de résistance. Mais on n'en aura la certitude qu'en mesurant l'atténuation de chaque étage, ce qu'on peut faire aussi en courant continu, à condition de ne pas dépasser à l'entrée une tension de 5 V en série avec une résistance de 50 Ω , sans oublier la charge de 50 Ω en sortie. Le rapport des tensions entre l'entrée et la sortie donne l'atténuation réelle.

La **figure 5** montre le circuit installé dans le boîtier Hammond, et la **figure 6** le prototype terminé. Bien solide et pratique, non ? Le prototype n'a pas seulement une bonne mine, il affiche aussi, compte tenu de sa simplicité, des caractéristiques



Fig. 7. Courbes de fréquences pour les atténuations minimale (0 dB) et maximale (31 dB).

très convenables: la **figure 7** montre les courbes de fréquences pour une atténuation de 0 dB et pour le maximum de 31 dB. On voit que l'atténuateur est à peu près stable jusqu'à 200 MHz. ◀

(160323 - version française : Helmut Müller)

Liens

- [1] <http://w1hue.org/filter.html>
- [2] www.eeweb.com/toolbox/microstrip-impedance
- [3] www.qsl.net/in3otd/parallr.html
- [4] www.elektormagazine.fr/160323

L'auteur

Alfred Rosenkränzer est ingénieur de développement depuis plus de 30 ans, au début dans le domaine de la télévision professionnelle. Depuis la fin des années 1990, il conçoit des circuits numériques à haute vitesse et des circuits analogiques pour les appareils de test des circuits intégrés.

CN/A audio pour RPi

lecteur audio sur réseau avec *Volumio*

Ton Giesberts
(labo d'Elektor)

Les mini-ordinateurs, tel le *Raspberry Pi*, sont vraiment prédestinés à être utilisés comme lecteur audio autonome et sur réseau.

Avec une distribution *Linux* spécifique comme *Volumio*, c'est pratiquement un jeu d'enfant. Ton Giesberts le prouve ici, ces lecteurs audio peuvent être de très bonne qualité.

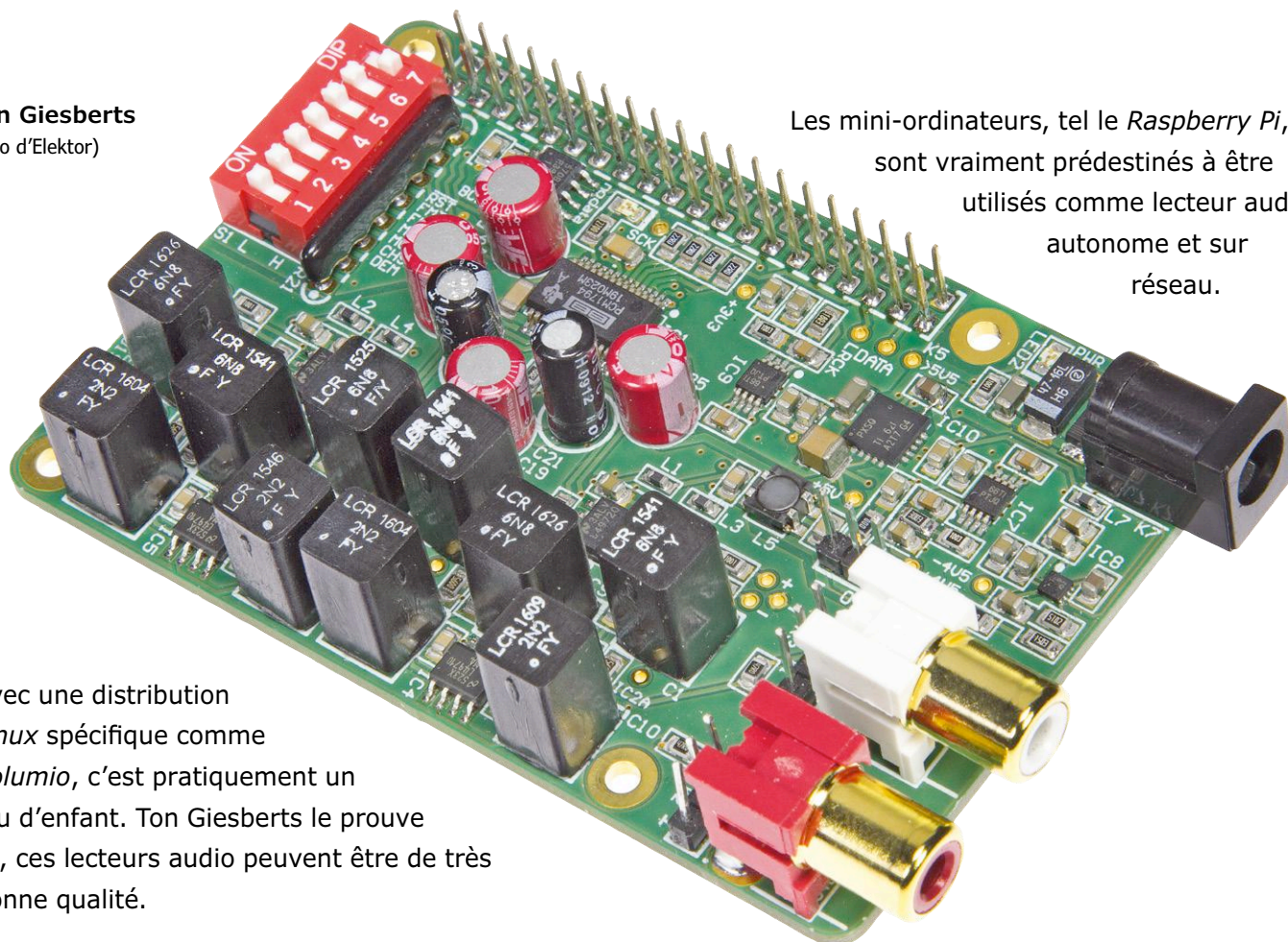
L'offre en lecteurs audio autonomes qui peuvent être mis sur un réseau est limitée, qui plus est si on souhaite aussi un écran tactile. Il y a bien quelques amplificateurs de marque, volumineux et au prix à l'avenant ; en général, ils n'ont pas d'écran tactile pour la commande non plus. En tout cas, si l'on veut un appareil compact et portable, ce n'est pas dans cette voie qu'il faut chercher. On en est vite réduit à se tourner vers une solution faite maison, et une application sur la base du *Raspberry Pi* constitue un bon point de départ. On trouve sur le marché des convertisseurs CN/A de bonne qualité pour ce mini-ordinateur,

mais pour Elektor nous voulions encore mieux. Nous avons donc décidé de concevoir un convertisseur pour le *RPi*, de la meilleure qualité possible, et avec des composants irréprochables.

Vue d'ensemble

Les circuits intégrés CN/A de *Burr-Brown* (désormais *Texas Instruments*) restent sans doute ce qui se fait de mieux, et nous les avons déjà utilisés par le passé. Nous avons choisi pour ce projet le PCM1794A, un des meilleurs du fabricant ; il s'agit d'un CN/A à 24 bits, qui accepte des taux d'échantillonnage de 10 à 200 kHz, et qui comprend un filtre à

suréchantillonnage ($\times 8$). Les caractéristiques de ce circuit intégré, fourni en boîtier *SSOP* (*Shrink Small Out-Line package*) à 28 broches, sont excellentes à tous points de vue : plage dynamique de 132 dB ($9 V_{eff}$ mono) et 127 dB ($2 V_{eff}$ stéréo), distorsion de 0,0004%, et sorties en courant symétriques (7,8 mA crête à crête). Il accepte en outre tous les formats de données connus : standard, justifié à gauche, et – important pour notre projet – *I²S*. L'alimentation de la partie numérique du circuit intégré est de 3,3 V, mais les entrées numériques acceptent les signaux 5 V. La fiche de caractéristiques est disponible sur le site de *TI* [1].



Les sorties en courant symétriques devront bien entendu être transformées en sorties en tension asymétriques à l'aide d'un convertisseur I/V, et filtrées. Forts de ces connaissances, nous avons dessiné le schéma de principe (**figure 1**) et établi notre « liste de courses » : un *Raspberry Pi 3* (la version 2 convient aussi), un CN/A tel que décrit ci-dessus, et un écran tactile compatible avec le *RPi* (nous avons choisi un modèle *Waveshare* de 3,5 pouces). La partie logicielle inclut *Raspbian* pour le *RPi*, *Volumio* (ou un programme équivalent, p. ex. *Mood Player*), et un pilote pour l'écran tactile.

Considérations sur l'alimentation principale

Le circuit intégré CN/A IC1 (voir le schéma en **figure 2**) a besoin de deux alimentations : 3,3 V pour la partie

convertisseurs I/V et les filtres de sortie ; pour éviter cela, nous utilisons des régulateurs, et un inverseur pour obtenir la tension négative à partir du +5 V. En y regardant de plus près, on constate que l'alimentation de la partie analogique est en fait de +5,2 V. Cette valeur quelque peu supérieure – sans danger pour le PCM1794A – est voulue, pour que la valeur maximale de la tension de sortie sans déformation du signal soit de 1 V. On rencontre souvent des spécifications de 2 V en sortie, mais il nous faudrait soit des tensions d'alimentation plus élevées, soit des amplis op avec une capacité de sortie rail à rail. La plupart des amplis op ayant cette capacité n'ont cependant pas les qualités des amplis op conçus spécifiquement pour l'audio haut de gamme.

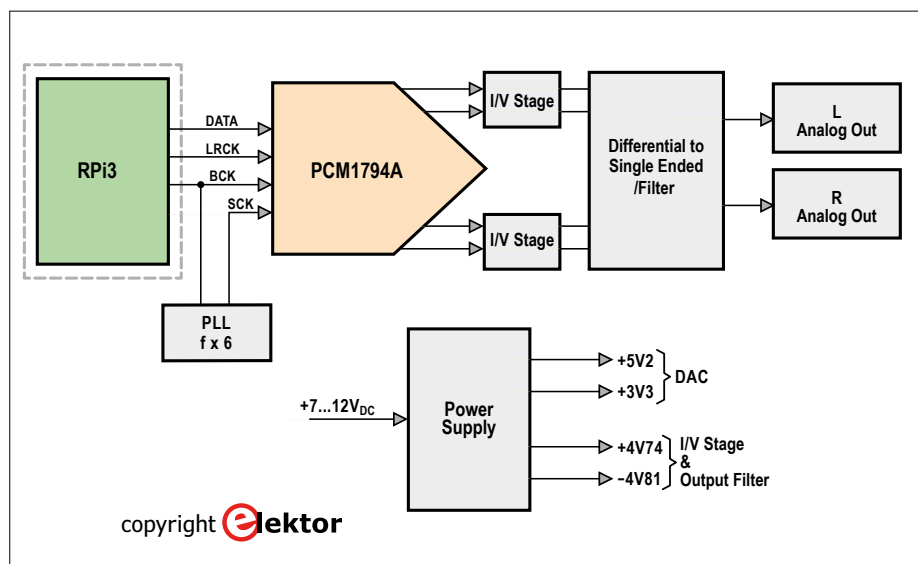


Figure 1. Schéma de principe du CN/A audio.

numérique et 5 V pour la partie analogique. Nous pourrions utiliser les tensions d'alimentation présentes sur le connecteur GPIO du *RPi*, mais elles sont tellement « polluées » qu'il ne resterait rien de notre quête du haut de gamme. Nous avons donc besoin de sources indépendantes pour les tensions de 3,3 V et 5 V ; il est à noter qu'on pourrait alors alimenter le *RPi* à partir de la platine CN/A, mais cela aurait aussi des effets néfastes sur la qualité finale. L'inconvénient est qu'il faudrait deux connecteurs pour l'alimentation positive, et un troisième serait nécessaire pour la tension négative des amplis op utilisés pour les

Pour l'obtention du 5,2 V, nous avons choisi le TPS7A4700, un régulateur de tension à faible chute : 307 mV pour un courant maximal de 1 A. La valeur de la tension de sortie est fixée en reliant certaines broches à la masse, connectées en interne à des résistances ; la tension régulée minimale est de 1,4 V, égale à la valeur de la référence de tension du circuit. Pour obtenir 5,2 V en sortie, ce sont les broches 6, 10 et 11 qui sont reliées à la masse :

$$V_{\text{sortie}} = 1,4V + 3,2V + 0,4V + 0,2V = 5,2V$$

INFOS SUR LE PROJET

audio

lecteur sur réseau

CN/A

Raspberry Pi

écran tactile

débutant

→ connaisseur

expert

Environ 5 h

Raspberry Pi 2 ou 3,
ordinateur sous Putty (Win-
dows) ou Linux,
deux adaptateurs réseau

110 €

Autres tensions

La tension d'alimentation positive des amplis op est fixée à 4,74 V, obtenue à partir du 5,2 V via le régulateur linéaire à très faible bruit IC7, un TPS7A4901. La valeur de la tension de sortie est fixée par le diviseur de tension R28/R29 :

$$V_{\text{sortie}} = \left(\frac{R28}{R29} + 1 \right) \times 1,185V$$

La tension de 3,3 V est obtenue via un régulateur identique, IC9, et le diviseur de tension R32/R33.

Pour la tension d'alimentation négative des amplis op, nous avons choisi un LM27761, un régulateur à découpage inverseur à faible bruit. L'inversion de tension s'effectue par commutation de condensateurs ; la tension obtenue est ensuite stabilisée par un régulateur linéaire à faible bruit. La valeur de la tension est fixée à -4,81 V par le diviseur de tension R30/R31 :

$$V_{\text{sortie}} = - \left(\frac{R30}{R31} + 1 \right) \times 1,22V$$

Rappelons que ces valeurs inhabituelles ($\pm 4,5$ V est plus courant) ont été calculées pour garantir une tension maximale du signal en sortie, sans déformation, de 1 V (0 dB).

L'entrée d'alimentation (via K7) est protégée contre une inversion de polarité par

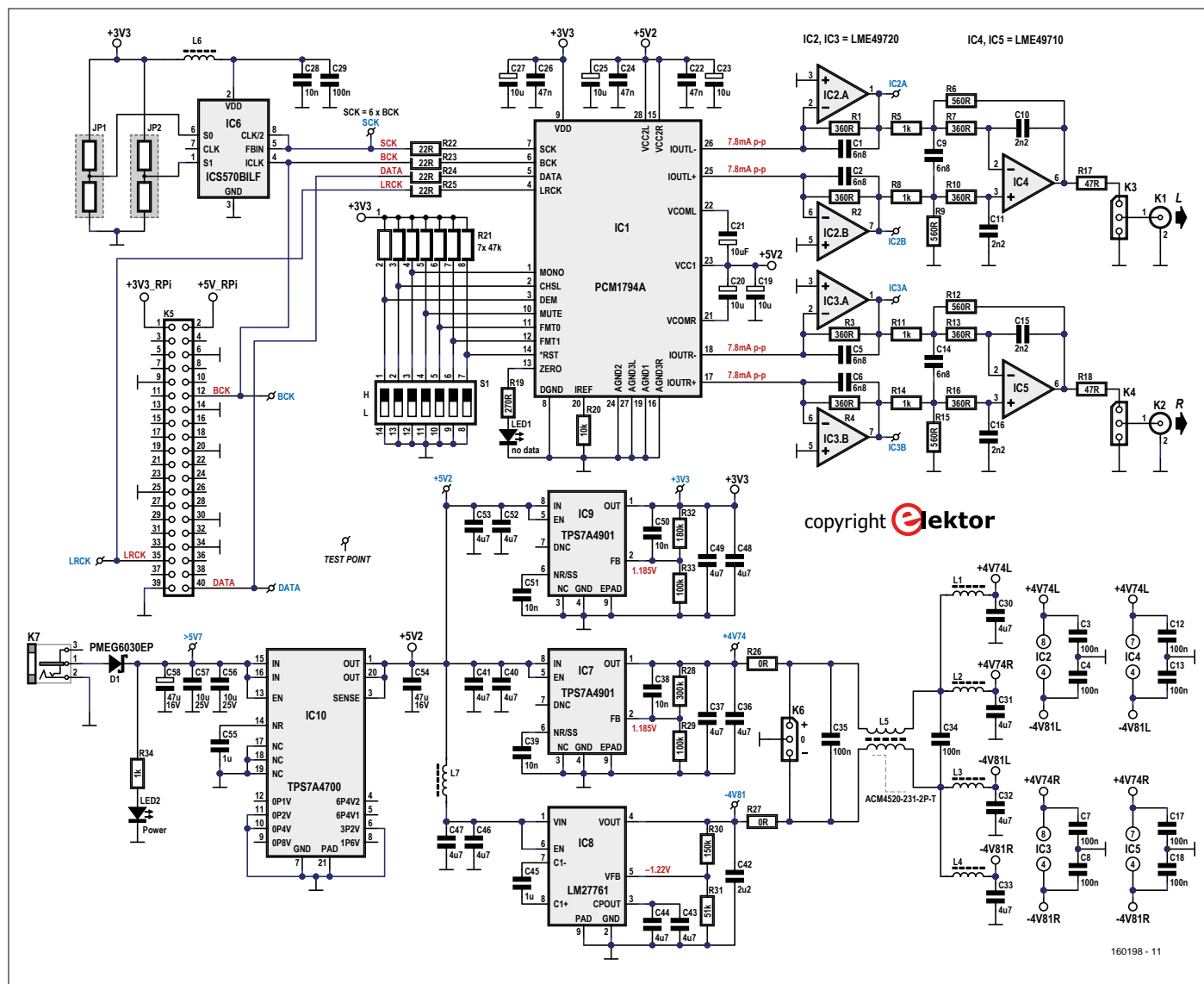


Figure 2. Le schéma détaillé du convertisseur CN/A pour le RPi.

la diode Schottky D1, qui n'occasionne que 0,3 V de chute de tension. Le courant nécessaire dépend de la fréquence d'échantillonnage, sa valeur est reprise dans le **tableau 2**.

Horloge principale

Pour que la carte CN/A audio fonctionne correctement, *Raspbian* (le système d'exploitation du RPi) doit être configuré pour un CN/A *HiFiBerry*. Les signaux I²S (*Inter-IC Sound*) seront alors disponibles sur le connecteur d'extension GPIO. Ces signaux comprennent un signal d'horloge continu (*SCK*) ou horloge bit (*BCLK/BCK*), un signal de sélection de mot (*WS*) ou horloge gauche-droite (*LRCLK/LRCK*), et un signal de données série (*SD/SDATA*). Pour la synchronisation, il faut aussi un signal d'horloge principal (*master clock*), mais cela n'est pas inclus dans les spé-

cifications I²S originales. Ce signal n'est donc pas présent sur le connecteur GPIO, et le CN/A doit le produire lui-même. Nous avons pour ce faire ajouté un multiplicateur à boucle à verrouillage de phase (*PLL*), un ICS570BILF (IC6). Ce circuit intégré dispose d'un tampon sans délai de propagation, les flancs montants des signaux d'entrée et de sortie coïncident donc parfaitement.

Le facteur de multiplication peut être sélectionné entre 0,5 et 32 à l'aide des entrées à trois états S0 et S1. La plage du signal de sortie va de 10 à 170 MHz. Nous avons choisi un facteur de multiplication de 6 (S0 et S1 ouvertes), et nous utilisons la sortie CLK 2, qui est aussi reliée à l'entrée de contre-réaction. Cette configuration offre la plage d'entrée la plus étendue (2,5 à 12,5 MHz) pour une plage de sortie de 15 à 75 MHz. On

pourrait croire qu'un signal dont la fréquence d'échantillonnage est de 32 kHz ne peut être reproduit, mais il n'en est rien : ça marche très bien.

Un petit souci

Lors du test de notre CN/A avec des signaux de 32 ou 24 bits et une fréquence d'échantillonnage entre 32 et 192 kHz, tout fonctionnait parfaitement. Nous avons cependant rencontré un problème avec des signaux de 16 bits : pas de signal de sortie ! Pour résumer notre longue investigation : le PCM1794A supporte bien les signaux audio de 16 et 24 bits, mais apparemment pas en mode I²S ; la fiche de caractéristiques n'est pas très claire à ce sujet... Est-ce que cela signifie que notre lecteur n'accepte pas les signaux de 16 bits ? Oui et non... *Volumio*, du moins la ver-

sion 1.55 que nous utilisons, permet la conversion de la fréquence d'échantillonnage avec trois niveaux de qualité. Cette possibilité sera sans doute aussi offerte par la version 2 du programme.

Le convertisseur N/A

Le PCM1794A intègre un filtre à suréchantillonnage ($\times 8$) et accepte les fréquences d'échantillonnage de 10

à 200 kHz. Les signaux I²S parviennent au CN/A via des résistances de 22 Ω (R22 à R25), pour éviter les parasites RF : pour une fréquence d'échantillonnage de 192 kHz, la fréquence d'horloge principale est de près de 74 MHz ! Les lignes de sélection matérielles (*MONO*, *CHSL*, *DEM*, *MUTE*, *FMT0*, *FMT1*, *RESET*) passent par un interrupteur DIP à 7 pôles, et un réseau de 7 résistances

de rappel de 47 k Ω ; le changement de configuration en est grandement facilité. La sortie *ZERO* (broche 13) commande LED 1, qui indique l'absence de données audio.

Pour le découplage des alimentations, nous utilisons des condensateurs à électrolyte solide (C19, 23, 25 et 27) ; leur résistance série équivalente (*ESR*) n'est que de 40 m Ω à 100 kHz, mais leur cou-



LISTE DES COMPOSANTS

Résistances (1%, 0,125 W, CMS 0805, sauf indication contraire)

R1, R2, R3, R4, R7, R10, R13, R16 = 360 Ω
 R5, R8, R11, R14, R34 = 1 k Ω
 R6, R9, R12, R15 = 560 Ω
 R17, R18 = 47 Ω
 R19 = 270 Ω
 R20 = 10 k Ω
 R21 = 47 k Ω , réseau de 7 résistances en boîtier SIP, 125 mW, 2%
 R22, R23, R24, R25 = 22 Ω
 R26, R27 = 0 Ω
 R28 = 300 k Ω
 R29, R33 = 100 k Ω
 R30 = 150 k Ω
 R31 = 51 k Ω
 R32 = 180 k Ω

Condensateurs

C1, C2, C5, C6, C9, C14 = 6,8 nF, 63 V, 1%, polystyrène, EXFS/HR 6 800 pF +/- 1%, LCR Components (pas de 5 mm ou CMS 0805)
 C3, C4, C7, C8, C12, C13, C17, C18, C29, C34, C35 = 100 nF, 50 V, 10%, X7R, CMS 0805
 C10, C11, C15, C16 = 2,2 nF, 63 V, 1%, polystyrène, EXFS/HR 2 200 pF +/- 1%, LCR Components (pas de 5 mm ou CMS 0805)
 C19, C23, C25, C27 = 10 μ F, 35 V, 0,04 Ω , diamètre max. de 6,3 mm, pas de 2 ou 2,5 mm, 870055673001 (WCAP-PTHR Series), Würth Elektronik
 C20, C21 = 10 μ F, 63 V, 1,06 Ω , diamètre max. de 6,3 mm, pas de 2 ou 2,5 mm, UPM1J100MDD, Nichicon
 C22, C24, C26 = 47 nF, 50 V, 10%, X7R, CMS 1206
 C28 = 10 nF, 50 V, 10%, X7R, CMS 0603
 C30, C31, C32, C33, C36, C37, C40, C41, C43, C44, C46, C47, C48, C49, C52, C53 = 4,7 μ F, 25 V, 10%, X5R, CMS 0805
 C38, C39, C50, C51 = 10 nF, 50 V, 10%, X7R, CMS 0805
 C42 = 2,2 μ F, 10 V, 10%, X7R, CMS 0805
 C45, C55 = 1 μ F, 16 V, 10%, X7R, CMS 0805
 C54 = 47 μ F, 16 V, 20%, X5R, CMS 1210
 C56, C57 = 10 μ F, 25 V, 10%, X5R, CMS 1206

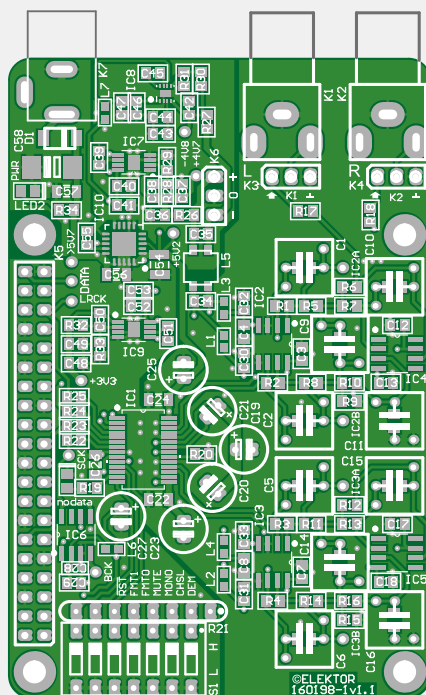


Figure 3. Le circuit imprimé, côté composants.

C58 = 47 μ F, 16 V, 10%, tantale, CMS-C (2312), TR3C476K016C0350, Vishay

Bobines

L1, L2, L3, L4, L6, L7 = 600 Ω @ 100 MHz, 0,15 Ω CC, 1,3 A, CMS 0603, BLM18KG601SN1D, Murata
 L5 = 230 Ω @ 100 MHz, 2 x 0,05 Ω CC, 2,6 A, CMS, ACM4520-231-2P-T, TDK

Semi-conducteurs

D1 = PMEG6030EP, 60 V, 3 A, CMS SOD-128
 LED1, LED2 = LED verte, faible consommation, CMS 0805
 IC1 = PCM1794ADB, CMS SSOP-28
 IC2, IC3 = LME49720MA/NOBP, CMS SOIC-8
 IC4, IC5 = LME49710MA, CMS SOIC-8
 IC6 = ICS570BILF, CMS SOIC-8
 IC7, IC9 = TPS7A4901DGNT, CMS MSOP 8
 IC8 = LM27761DSGT, CMS WSON-8
 IC10 = TPS7A4700RGWT, CMS VQFN-20

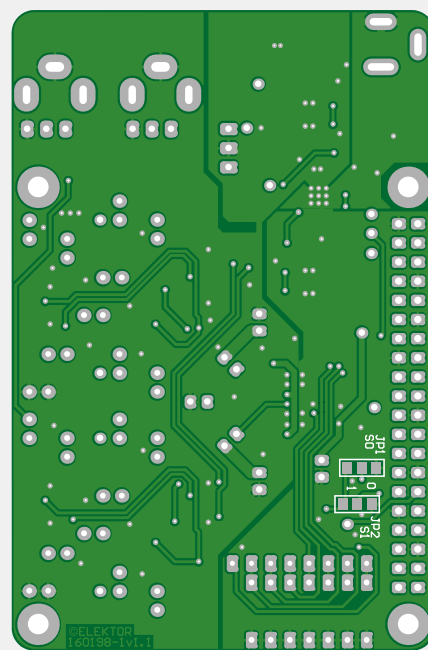


Figure 4. Le circuit imprimé, côté soudure. En principe, il ne faut pas monter JP1 et JP2.

Divers

K1 = connecteur audio RCA blanc pour circuit imprimé, doré, PJRAN1X1U02AUX, Switchcraft
 K2 = connecteur audio RCA rouge pour circuit imprimé, doré, PJRAN1X1U03AUX, Switchcraft
 K3, K4, K6 = 3 picots à souder, pas de 2,54 mm
 K3, K4 = cavalier, pas de 2,54 mm
 K5 = connecteur GPIO à 40 broches, 2x20, femelle, extra haut
 K7 = connecteur alimentation CC, 3 A, 1,95 mm, Lumberg NEB 21 R
 S1 = interrupteur DIP à 7 pôles
 4 entretoises M2,5 de 17 mm avec bout fileté
 4 entretoises M2,5 de 14 mm avec bout fileté
 8 écrous et rondelles M2,5

Circuit imprimé 160198-1

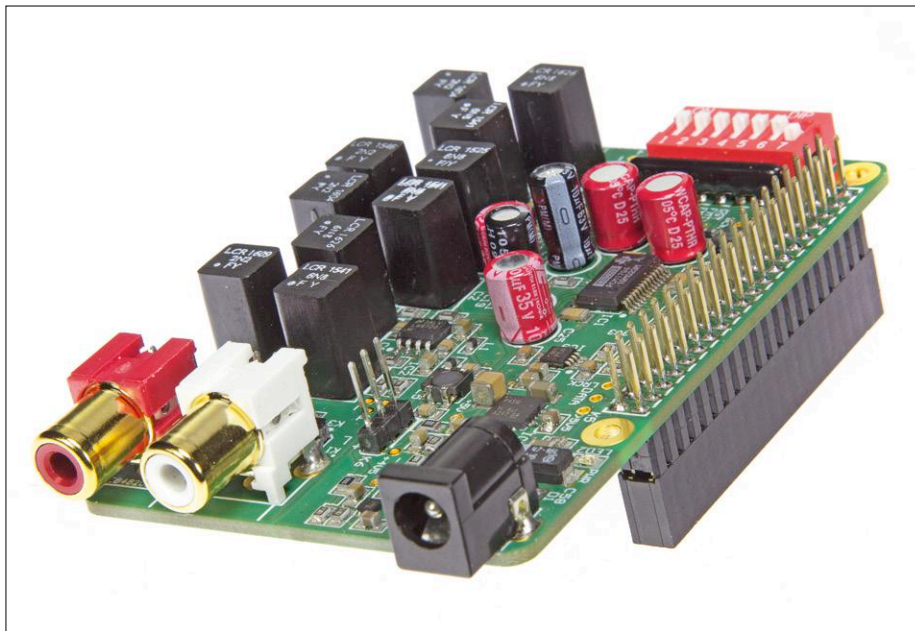


Figure 5. Le connecteur GPIO est inséré côté soudure.

Tableau 1 : positions de S1

S1-1	L	Désaccentuation 44,1 kHz désactivée
S1-2	L	Filtre numérique à flanc raide
S1-3	L	Sortie mono
S1-4	L	Sortie coupée (<i>mute</i>)
S1-5, S1-6	L	Mode I ² S
S1-7	H	Remise à zéro (<i>reset</i>) désactivée

Tableau 2 : courant en fonction de la fréquence d'échantillonnage

Fréquence d'échantillonnage	Courant (K7)
Pas de données	120 mA
32 kHz	127,5 mA
44,1 kHz	131,7 mA
48 kHz	133 mA
96 kHz	149,5 mA
192 kHz	182,1 mA
(mesuré avec une alimentation de 8 V, les deux sorties terminées sur 10 kΩ)	

Tableau 3 : distorsion harmonique

DHT + bruit				
Fréquence d'échantillonnage		48 kHz	96 kHz	192 kHz
1 kHz	B = 22 kHz	0,0008 %	0,0009 %	0,0013 %
	B = 80 kHz	0,0028 %	0,0012 %	0,0014 %
7 kHz	B = 22 kHz	0,00095 %	0,0011 %	0,0013 %
	B = 80 kHz	0,003 %	0,0014 %	0,0016 %
DIM				
50 Hz : 7 kHz = 4:1		0,0014 %	0,002 %	0,0036 %

rant de fuite maximal est relativement élevé, 100 μ A. Nous ignorons si ce courant de fuite peut influencer les réglages de courant internes du circuit intégré, nous utilisons donc aussi des condensateurs électrolytiques « ordinaires » : ESR de 1,06 Ω à 100 kHz, courant de fuite de 4 μ A. On pourra expérimenter avec divers types de condensateurs, il y a suffisamment de place sur le circuit intégré pour cela.

Le filtre de sortie

Les produits de la fréquence d'échantillonnage sont limités par le filtre à suréchantillonnage, mais il y a tout de même des composantes HF présentes en sortie du CN/A. Pour atténuer ces fréquences, nous utilisons un filtre de Butterworth passe-bas du 3^e ordre. Les filtres et les convertisseurs I/V sont imbriqués et bâtis autour des amplis op IC2 et IC3. La fréquence de coupure est de 64,5 kHz. Une fréquence de 352,8 kHz, correspondant à 8 fois une fréquence d'échantillonnage de 44,1 kHz, est atténuée de plus de 40 dB ; pour une fréquence d'échantillonnage de 192 kHz, l'atténuation de l'harmonique 8 est supérieure à 80 dB. Le choix de la fréquence de coupure du filtre est toujours un compromis entre une atténuation suffisante avec des fréquences d'échantillonnage basses, et la bande passante du signal audio ; ici, cette bande passante est tout de même plus de trois fois supérieure au spectre audible.

Les sorties du CN/A sont en courant et symétriques, ce qui permettrait l'utilisation d'un filtre externe. Pour notre carte audio, les sorties doivent être converties en tension, et désymétrisées. Les convertisseurs I/V sont bâtis autour d'IC2 et IC3 ; les sorties symétriques d'IC2A/IC2B et IC3A/IC3B sont ensuite désymétrisées par IC4 et IC5, qui sont aussi configurés en filtres passe-bas du 2^e ordre. Grâce à la combinaison avec les convertisseurs I/V, nous avons bien finalement des filtres du 3^e ordre (avec une fonction de transfert de type Butterworth).

Les signaux de sortie sont disponibles sur K1 et K2, via K3 et K4. Les signaux peuvent transiter directement de K3/K4 vers K1/K2, ou on peut insérer un potentiomètre double comme réglage de volume entre les connecteurs.

Le circuit intégré CN/A ne permet pas le réglage du volume. On pourrait utiliser *Volumio*, mais ce serait au détriment de la résolution, et donc de la qualité, ce qui n'est bien entendu pas acceptable.

Nous avons conçu un circuit séparé pour la commande du volume et de la balance par télécommande. Ce circuit sera décrit dans le prochain numéro d'Elektor, il est aussi sur le site du labo [3]. Un dernier détail : l'alimentation symétrique des amplis op est filtrée par une inductance en mode commun et des filtres LC (un par canal).

Le circuit imprimé

Nous n'avons besoin que d'une seule alimentation de 7 à 8 V pour la platine, le *RPi* ayant sa propre alimentation. Nous ne recommandons pas d'utiliser une alimentation de 9 à 12 V, bien que ce soit en théorie possible : la dissipation d'IC10 serait trop importante.

Les **figures 3 et 4** montrent les deux faces du circuit imprimé, dont les dimensions sont identiques à celles du *RPi*. Les connecteurs d'alimentation et de sortie sont d'un même côté du circuit, l'interrupteur DIP S1 de l'autre (là où se trouve sur le *RPi* l'antenne *Wi-Fi*). Le connecteur GPIO à 40 broches K5 est inséré du côté soudure (voir photos).

Les pattes de K1 à K4, K6 et K7 côté soudure doivent être coupées le plus possible à ras du circuit imprimé, pour éviter tout contact avec les connecteurs réseau et *USB* du *RPi* ; un morceau d'isolant entre les deux circuits serait même conseillé. Nous avons utilisé des entretoises M2,5 de 17 mm avec bout fileté pour fixer le CN/A au *RPi*. Les îlots de soudure pour les condensateurs des filtres de sortie (C1, C2, C6, C9-C11, C14-C16) sont tels que divers modèles peuvent être utilisés, p. ex. CMS 0805 ou conventionnels au pas de 5 mm. Nous recommandons l'utilisation de condensateurs au polystyrène avec une tolérance de 1%.

La broche 1 des circuits intégrés est identifiable grâce à une petite bande blanche ; ce n'est pas facile à voir pour IC7, à cause de la présence de C38. Le numéro des résistances et condensateurs CMS 0805 et 1206 est à l'emplacement même du composant, il n'y avait pas assez de place à côté ; les numéros ne sont donc plus visibles une fois les composants montés. Pour le montage à la main, avec un pistolet à air chaud ou au four, il est préférable d'utiliser une copie agrandie de la photo du circuit imprimé côté composants, pour faciliter la mise en (bonne) place des CMS. Une fois les composants traversants soudés, ce qui se fait en dernier, il sera très difficile de dessouder la plupart des CMS 0805 !

JP1 et JP2 sont montés côté soudure ; ils peuvent en principe rester ouverts ($SCK = 6 \times BCK$). Les connecteurs audio *RCA* sont de marque *Switchcraft* ; leurs pattes de connexion sont plus rapprochées que sur des modèles standard, où l'espacement est de 10 mm. Si vous utilisez des connecteurs d'une autre marque, il est probable qu'il ne soit pas possible de les insérer sur le circuit imprimé.

L'écran tactile de 3,5 pouces de chez *Waveshare* doit être monté 16 mm au-dessus du CN/A. Nous avons utilisé pour ce faire des entretoises M2,5 de 14 mm avec bout fileté, et des écrous M2,5 pour la distance additionnelle. Ne poussez pas le connecteur de l'écran au maximum vers le bas ! En cas de doute, il vaut mieux utiliser un connecteur supplémentaire et monter l'écran un peu

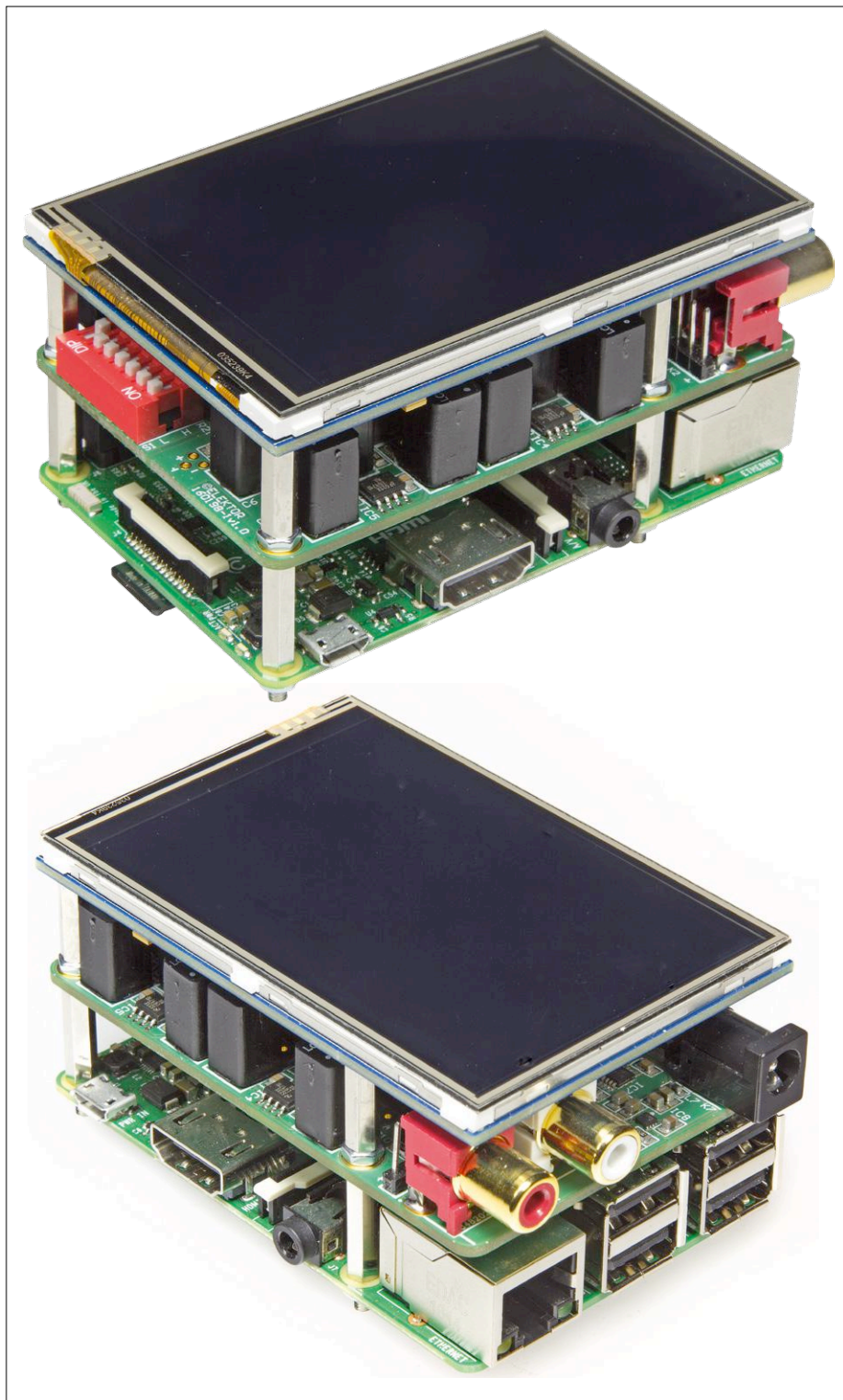


Figure 6. Le lecteur audio en réseau complet.

Volumio

Volumio est une distribution *Linux* à code ouvert (*open source*) gratuite, spécifiquement adaptée à la reproduction musicale. Le logiciel tourne sur de nombreux appareils, dont les mini-ordinateurs comme le Raspberry Pi. Après installation, l'appareil cible devient un lecteur audio « sans tête » (*headless audiophile music player*) ; il faut en effet un autre ordinateur, téléphone ou tablette pour le commander.

Ceci est possible grâce à l'interface utilisateur de *Volumio*, une application *web* qui tourne sur tout appareil disposant d'un navigateur. L'interface est simple et intuitive ; elle permet la reproduction de fichiers musicaux. Les communications entre *Volumio* et l'application web se font via le réseau local.

Les instructions pour l'installation de *Volumio* se trouvent sur la page du projet au labo [2]. Nous avons utilisé la version 1.55 ; la version 2, qui date de décembre 2016, ne supporte pas (encore) l'écran tactile utilisé dans ce projet.

La **figure 7** vous montre la commande de *Volumio* via l'écran tactile (à gauche) et un ordinateur (à droite).

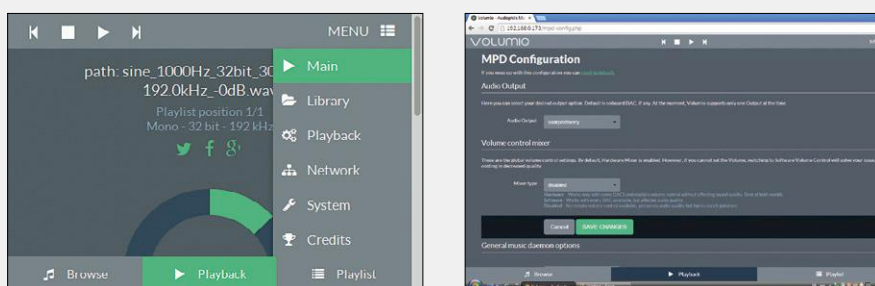
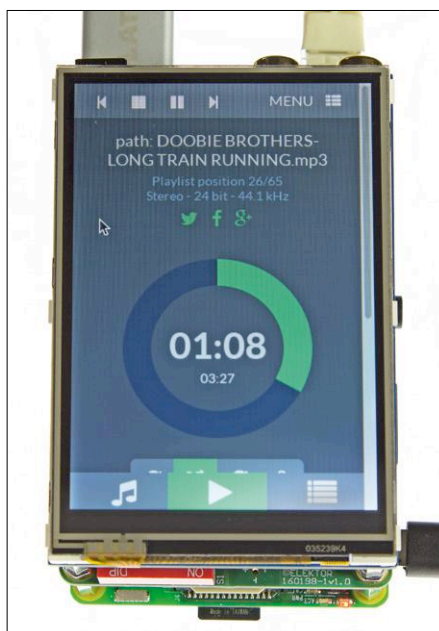
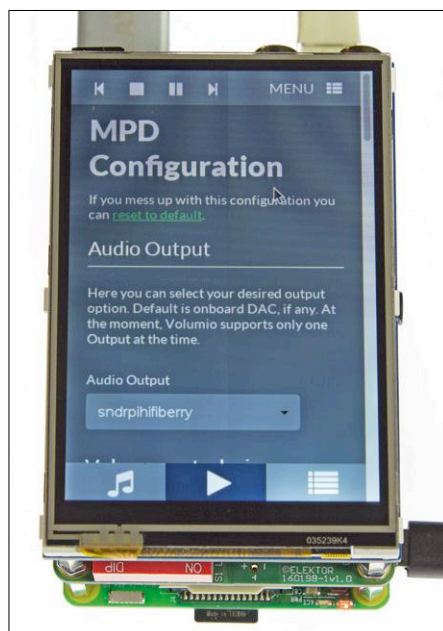


Figure 7. *Volumio* peut être commandé via l'écran tactile (à gauche), ou un ordinateur (à droite).



Liens

- [1] www.ti.com/lit/ds/symlink/pcm1794a.pdf
- [2] www.elektormagazine.fr/labs/audio-dac-for-rpi-networked-audio-player-using-volumio
- [3] www.elektormagazine.fr/labs/volume-control-160321-for-raspberry-pi-dac-160198



DANS L'E-CHOPPE

→ réf. 160198-1 :

Circuit imprimé vierge

→ réf. 160198-91 :

Module monté avec écran tactile ; les connecteurs doivent être soudés par l'utilisateur

→ réf. 17631 :

Raspberry Pi 3, modèle B

plus haut. La distance entre le connecteur à 26 broches de l'écran et le circuit imprimé du CN/A est d'environ 2 mm. Le résultat final est visible en **figures 5** et **6**.

Mesures

Nous avons bien entendu soumis notre CN/A audio à toute une batterie de tests. Les principaux résultats sont repris aux **tableaux 2** et **3**. La position des interrupteurs S1, utilisée pour les mesures, est résumée dans le **tableau 1**.

Le courant est proportionnel à la fréquence d'échantillonnage, voir **tableau 2**.

Le **tableau 3** résume les mesures de distorsion harmonique totale (DHT), augmentée du bruit, et de distorsion par intermodulation (DIM).

Sur la page du projet au labo [2], quelques graphiques obtenus avec un analyseur *Audio Precision* sont aussi disponibles. ◀

(160198 – version française : Jean-Louis Mehren)



trucs et astuces

les lecteurs écrivent aux lecteurs

Encore des solutions futées qui facilitent la vie des électroniciens.

Un appareil-photo numérique en guise de générateur de mire

Alfred Rosenkränzer (alfred_rosenkraenzer@gmx.de)

Pour mesurer les performances d'une installation de surveillance vidéo, j'avais envie depuis quelque temps d'utiliser un petit générateur de mire mobile et autonome. Je possède divers générateurs (hérités de mon ancienne activité de développeur de techniques vidéo), mais ils ne sont ni petits ni autonomes. J'ai donc commencé à réfléchir à ce que je pourrais faire avec des CPLD, de la RAM et des CN/A.

Et tout à coup, j'ai réalisé que j'avais déjà quelque chose d'équivalent sous la main (vous aussi, sans doute) : tous ces composants sont déjà rassemblés dans un appareil-photo numérique. Non seulement il peut afficher les images enregistrées sur son écran, mais il peut les envoyer vers un moniteur externe par un connecteur de sortie. Les anciens APN n'ont qu'une sortie analogique (souvent un jack 3,5 mm), les plus récents ont une sortie HDMI.

Des images toutes prêtes existent sur l'internet par ex. chez Buroschi [1]. Mais aucun de mes APN ne voulait afficher une image qu'il (lui ou un autre APN similaire) n'aurait pas prise lui-même. Au bout d'un moment, j'ai abandonné l'idée de trouver et modifier les métadonnées EXIF du fichier d'une mire. Si un lecteur s'y connaît, j'apprécierais beaucoup qu'il me donne un coup de main.

Pour contourner ce problème, j'ai tout simplement fait une photo avec l'APN et j'ai échangé le contenu de la photo à l'aide d'un éditeur photo spécialisé. Vu le peu de choses à afficher, une résolution de 640 × 480 ou 800 × 600 pixels est suffisante pour la sortie analogique. La norme vidéo en PAL est de 720 × 576 et en NTSC de 720 × 480 pixels.

La figure 1 montre la mire que j'ai ainsi enregistrée dans mes trois APN. Un signal vidéo de test dit multiburst peut être utilisé pour mesurer la réponse en fréquence d'une ligne de transmission (dans mon cas, long de 80 m). Comme son nom le laisse entendre, il est constitué de plusieurs salves qui commencent à basse fréquence puis atteignent la fréquence de coupure haute du système (env. 5 MHz en PAL et 4,2 MHz en NTSC). Un signal multiburst est également intégré dans la mire normalisée, autrefois diffusée pendant qu'il n'y avait pas d'émissions TV. À l'autre bout de la ligne de transmission, on observe soit la mire à l'aide d'un moniteur, soit



Fig. 1. La mire sur mes trois APN.

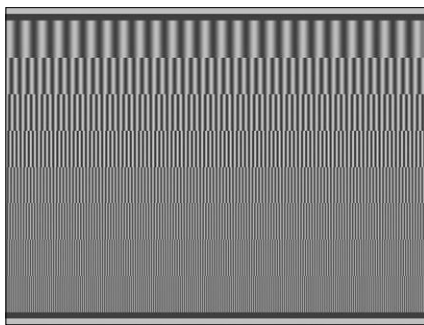


Fig. 2. La mire multiburst concoctée maison.

le signal à l'aide d'un oscilloscope. Si toutes les salves de fréquence ont la même amplitude, cela veut dire que la réponse en fréquence est plate et la bande passante élevée. Les salves peuvent être affichées sur une ligne ou bien être « saucissonnées » pour apparaître sur une verticale dans l'image. On parle alors de multiburst horizontal ou vertical.

Comme je ne trouvais pas une telle mire, je l'ai fabriquée moi-même (les lecteurs intéressés peuvent

m'envoyer un courriel, je leur donnerai volontiers des détails). Sur les bords supérieur et inférieur, j'ai ajouté les niveaux max. (clair) et min. (sombre) : ils forment des bandes de référence. Le début et la fin de chaque ligne virent au gris pour éviter l'apparition de suroscillations. Vous pouvez admirer le résultat sur la figure 2.

Le convertisseur N/A et le filtre de sortie d'un APN n'ont malheureusement pas la qualité d'un générateur de mire professionnel. La fréquence d'échantillonnage n'est pas toujours suffisante pour couvrir la totalité du spectre de 5 MHz. Il faut donc tenir compte des limitations de la fréquence maximale du signal et de la qualité de l'image. Les fichiers des images que j'ai utilisées sont disponibles au téléchargement sur le site d'Elektor [2]. Si votre APN se montre aussi rebelle que les miens, vous pouvez comme exposé ci-dessus échanger le contenu d'une image prise avec l'APN avec celui d'une mire avec Photoshop ou un autre logiciel similaire. ◀

(160426 – version française : Yves Georges)

[1] <https://goo.gl/WHE0H9>

[2] www.elektormagazine.fr/160426

Vous avez une solution futée pour arranger une bricole... Une façon bien à vous d'utiliser 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 € !



BBC micro:bit pour électroniciens (2)

enregistreur de données et oscilloscope

Burkhard Kainka

Tout microcontrôleur muni d'un convertisseur A/N et d'une interface pour PC peut servir de sonde pour un système d'acquisition de données. Avec la carte BBC micro:bit s'y rajoutent un petit afficheur à LED et une interface sans fil. Idéal pour des applications spéciales dans un labo d'électronique !

C'est justement en raison de ses dimensions réduites que la carte micro:bit est prédestinée aux applications de métrologie. Qu'elle soit alimentée par câble USB ou pile, qu'elle communique par Bluetooth ou un protocole sans fil simplifié, il sera toujours aisé de la placer à proximité de l'objet à mesurer.

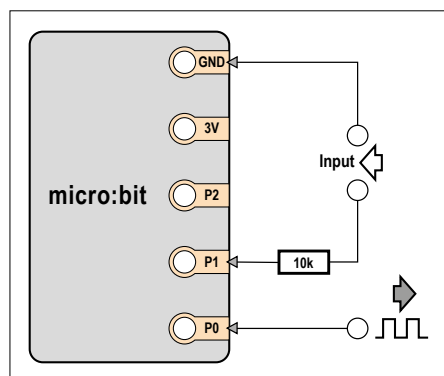
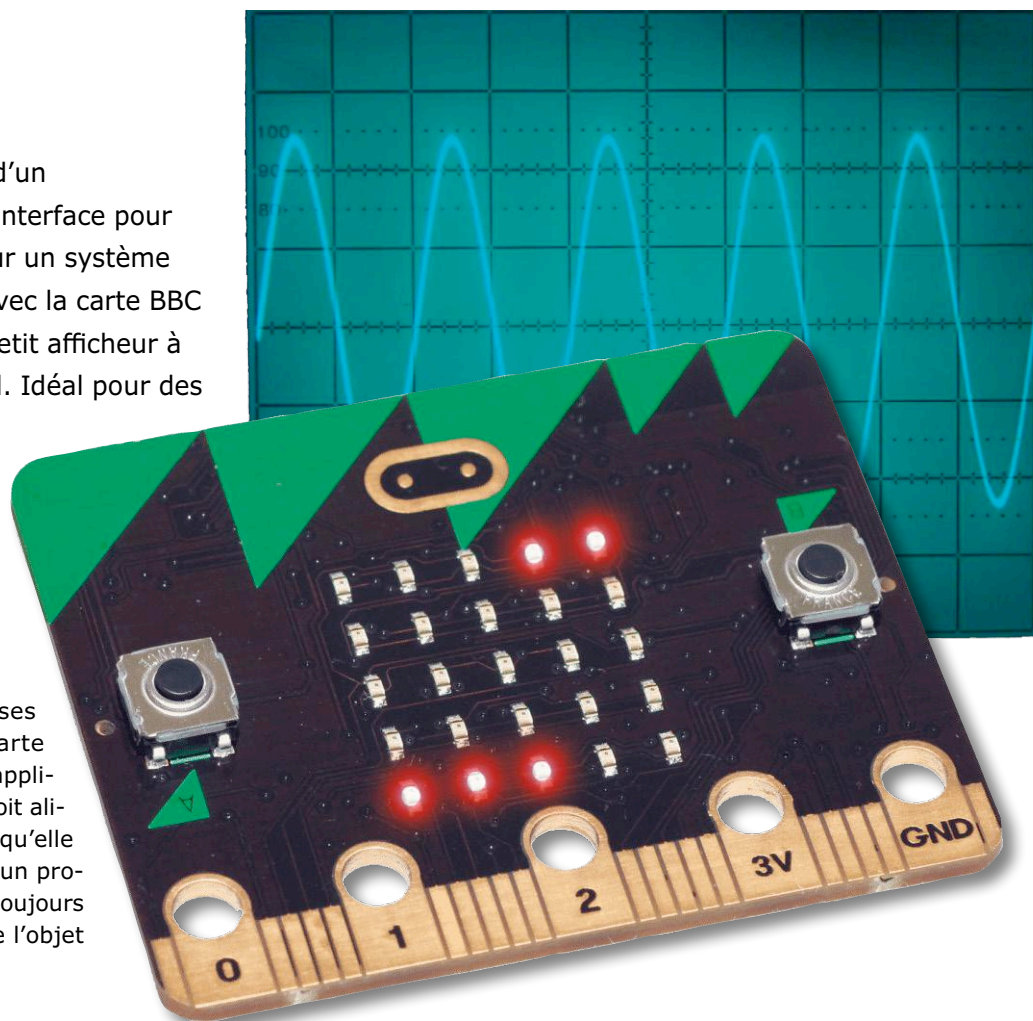


Fig. 1. Entrée de mesure et sortie du signal.

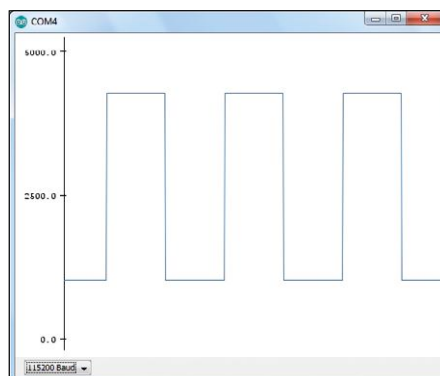


Fig. 2. Un signal carré à 10 Hz.

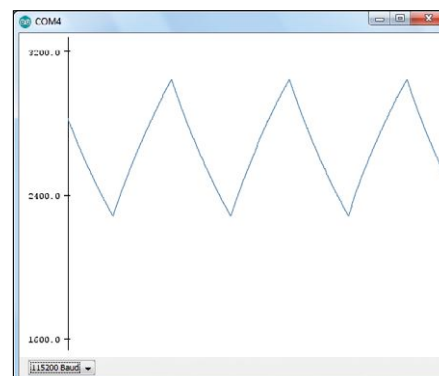


Fig. 3. Le signal filtré.

Un oscilloscope USB

La plateforme mbed a fait ses preuves pour la programmation de la micro:bit. Les premiers pas ont été expliqués en [1]. Tous les programmes de cet article sont disponibles en téléchargement au format texte sur la page [2] du site d'Elektor et doivent être copiés dans un projet mbed existant.

Lors de toute mise en œuvre de la micro:bit pour des tâches générales de mesure, on veillera scrupuleusement à ne pas dépasser l'étendue de mesure entre GND et VCC. Dans tous les cas, une résistance de protection de 10 kΩ en série avec l'entrée s'avèrera utile afin de limiter le courant en cas de dépassement involontaire (**fig. 1**). Outre l'entrée analogique, on dispose aussi d'une sortie de signal carré, utile comme signal de test. Le programme du **listage 1** capture les mesures à vitesse maximale sans sauvegarde intermédiaire, et les retransmet immédiatement. Une part significative du temps de cycle provient de la transmission série à 115 200 bauds. Puis le déroulement de la mesure sera différent si celle-ci s'effectue sur un seul chiffre (3 mV) ou quatre chiffres (3000 mV). C'est pourquoi on préférera augmenter la tension de mesure de 1000 mV, ce qui donnera des valeurs de 1000 mV à 4300 mV qui requièrent toutes le même temps de traitement. Le programme inclut aussi une source de signal, on dispose ainsi directement d'un signal de test. P1 devient alors une sortie à MLI de fréquence 10 Hz et de période 100 ms. Il existe de nombreuses possibilités d'exploitation des mesures. On peut recevoir les données à l'aide d'un programme de terminal puis les afficher dans une feuille de calcul. Une alternative confortable est le traceur série de l'EDI d'Arduino à partir de la version 1.6.8. Le logiciel fournit un affichage continu et ajuste automatiquement la zone d'affichage, de sorte qu'un changement de gamme est superflu. Nous avons mesuré un signal carré symétrique de 10 Hz avec les paramètres donnés (**fig. 2**). On dispose ainsi d'un axe du temps défini. L'oscillogramme complet couvre bien entendu un temps de mesure de 300 ms.

Le traceur de l'EDI d'Arduino trace toujours d'abord de gauche à droite, jusqu'à ce que l'écran soit rempli. Après quoi l'image se décale à gauche, de sorte que les données plus anciennes disparaissent. Le problème de l'affichage continu ou stationnaire est résolu par la

Listage 1. Mesure rapide avec retransmission directe des données.

```
//Voltage Logger/Scope
#include "MicroBit.h"
MicroBit uBit;

int main()
{
    uBit.init();
    MicroBitSerial serial(USBTX, USBRX);
    uBit.io.P1.setAnalogValue(512);
    uBit.io.P1.setAnalogPeriodUs(100000);
    while (1) {
        if(uBit.buttonA.isPressed()){
            int u = 1000+3300 * uBit.io.P0.getAnalogValue()/ 1023;
            uBit.serial.printf("%d\r\n", u);
            // uBit.sleep(100);
        }
    }
}
```

touche A (`if(uBit.buttonA.isPressed())` dans le listage 1). En appuyant sur la touche A durant la mesure, on obtient un affichage continu. Dès que la touche est relâchée, l'affichage se fige pour en permettre une meilleure observation ou une sauvegarde.

Le signal de sortie supplémentaire peut

s'avérer utile si l'on souhaite tester des circuits ou des composants. Grâce au filtre passe-bas composé de 4,7 kΩ et 22 μF, le signal carré est transformé en dent de scie comme prévu (**fig. 3**). La plage de mesure du traceur de l'Arduino s'adapte automatiquement aux tensions mesurées les plus faibles.

Listage 2. Sauvegarde rapide suivie de la transmission.

```
//Fast Scope
#include "MicroBit.h"
MicroBit uBit;

int main(){
    char d[400];
    uBit.init();
    MicroBitSerial serial(USBTX, USBRX);
    uBit.io.P1.setAnalogValue(512);
    uBit.io.P1.setAnalogPeriodUs(2000);
    while (1) {
        if(uBit.buttonA.isPressed()){
            for(int i = 0; i < 400; i++){
                d[i] = uBit.io.P0.getAnalogValue()/ 4;
            }
            for(int i = 0; i < 50; i++) uBit.serial.printf("%d\r\n", 0);
            for(int i = 0; i < 400; i++){
                uBit.serial.printf("%d\r\n", d[i]);
            }
            for(int i = 0; i < 50; i++) uBit.serial.printf("%d\r\n", 255);
        }
        uBit.sleep(500);
    }
}
```

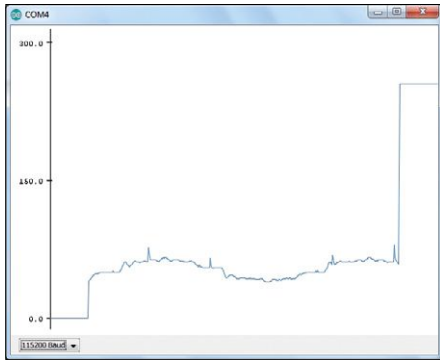


Fig. 4. Mesure à fréquence d'échantillonnage plus élevée.

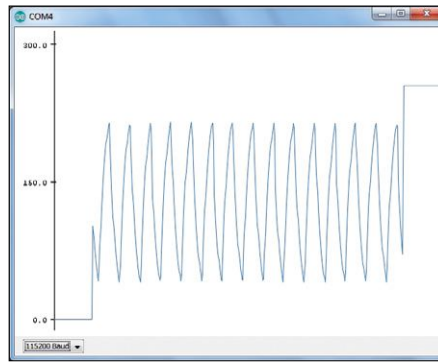


Fig. 5. Une dent de scie à 500 Hz.

Augmentation de la fréquence d'échantillonnage par mise en cache

Si on peut s'affranchir de la durée de la transmission série, alors seule la durée de la mesure du convertisseur A/N détermine la fréquence d'échantillonnage maximale. On crée donc un tableau de données, on le remplit avec les mesures, et ensuite on le transmet au PC. Mais c'est là que commencent les problèmes auxquels on n'avait peut-être pas pensé. Bien que le contrôleur dispose généreusement de RAM, on atteint déjà la limite avec un tableau de type `int d[100]`, car l'environnement d'exécution (*runtime*) de la micro :bit ne laisse plus grand-chose.

Mais si l'on souhaite utiliser le traceur, 500 mesures sont souhaitables. Cependant il faut savoir que le type `int` occupe quatre octets dans un système à 32 bits. On dispose donc de 400 octets. C'est pourquoi nous déclarons `char d[400]` qui accapara 400 octets. Grâce à une division par quatre, les données sur dix bits du convertisseur A/N sont converties en huit bits, plus économes en place dans la mémoire. Désormais seuls 400 octets sont donc sauvegardés et transmis (**listage 2**). Ne manquent donc en fait que cent octets de nécessité vertu, et envoyer un en-tête de zéro octet suivi de 255 octets : le tra-

ceur représentera alors toujours la pleine échelle et donnera à l'utilisateur une vision claire des limites de l'affichage. La mesure à la **figure 4** montre un signal à 50 Hz avec les parasites typiques que l'on recueille avec un câble de mesure non connecté. La mesure dure 30 ms, car environ 1,5 période sont affichées, ce qui signifie une fréquence d'échantillonnage d'environ 13 kHz avec 400 points de mesure. Comme la fréquence du signal MLI a été ajustée à 500 Hz dans ce programme, on peut facilement vérifier cela avec son propre signal. La **figure 5** montre le signal MLI à la sortie d'un passe-bas de 4,7 kΩ et 100 nF.

Transmission sans fil des mesures

La carte BBC micro:bit est équipée du Bluetooth Low Energy (BLE). On cherchera en vain une puce spéciale sur la carte, les sections HF sont intégrées au microcontrôleur. Le circuit nRF51822 de Nordic Semiconductor a été développé initialement pour des applications telles que claviers et souris sans fil qui ne nécessitent pas de longue portée, mais usent avec parcimonie l'énergie d'une pile. La micro:bit en tire profit également. On peut alimenter la carte avec une pile de 3 V, et ainsi renoncer au câble USB. Le système est alors compatible aussi avec des applications de longue durée avec alimentation par pile. Les caractéristiques principales du nRF51822 parlent d'elles-mêmes :

- émetteur-récepteur à 2,4 GHz
- sensibilité de -93 dBm en mode Bluetooth® Low Energy
- débits de données supportés : 250 kbps, 1 Mbps, 2 Mbps
- puissance d'émission : -20 à +4 dBm par pas de 4 dB
- puissance d'émission en mode *Whisper* : -30 dBm
- réception 13 mA crête, émission 10,5 mA crête (0 dBm)
- réception 9,7 mA crête ; émission 8 mA crête (0 dBm) avec DC/DC
- RSSI (résolution de 1 dB)
- processeur à 32 bits ARM® Cortex™-M0
- fonctionnement de la mémoire flash : 275 µA/MHz
- fonctionnement de la RAM : 150 µA/MHz
- *Serial Wire Debug* (SWD)

La programmation avec mbed permet l'utilisation de Bluetooth, on peut ainsi

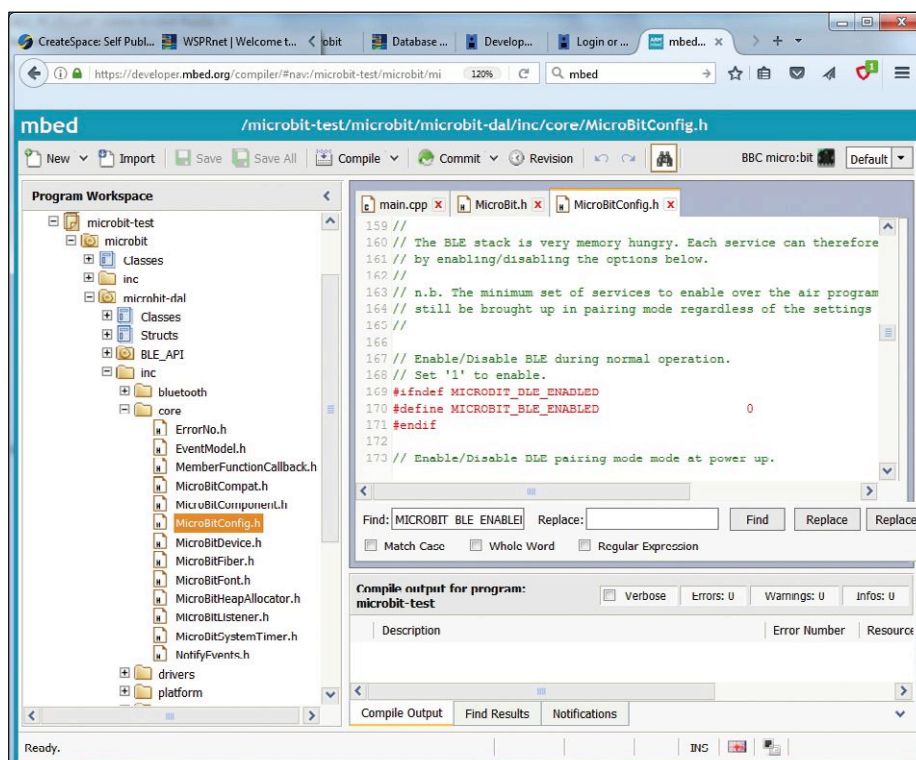


Fig. 6. Désactivation de BLE.

envoyer des données directement à sa tablette ou son ordiphone. Cela requiert néanmoins une pile logicielle assez grande et le développement d'une appli appropriée pour l'appareil correspondant. Mais il est possible de faire plus simple. On peut en effet communiquer avec l'émetteur-récepteur à 2,4 GHz à un niveau inférieur et renoncer ainsi au protocole Bluetooth compliqué. Pour cela, on dispose de l'aide de MicroBitRadio, avec des datagrammes simplifiés, c'est-à-dire des messages courts qui utilisent directement l'émetteur-récepteur. Une carte micro:bit envoie un message court, celui-ci est reçu par toutes les autres à sa portée. Pour cela un canal et une puissance d'émission par défaut ont été définis, en fait on ne se soucie donc de rien. Voilà une des méthodes des plus simples pour transmettre des données sans fil, et effectuer des mesures flottantes. Une application typique pourrait être l'ECG, une mesure flottante affranchie des ronflements perturbants. On a simplement besoin de deux cartes micro:bit, l'une utilisée en instrument de mesure qui émet les données, l'autre

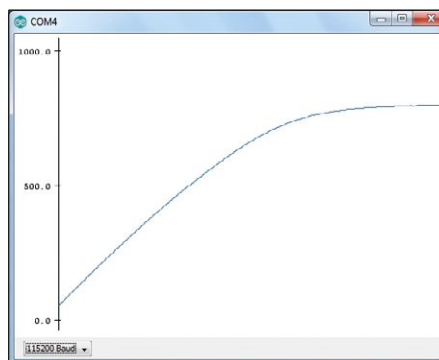


Fig. 7. Mesures envoyées sans fil.

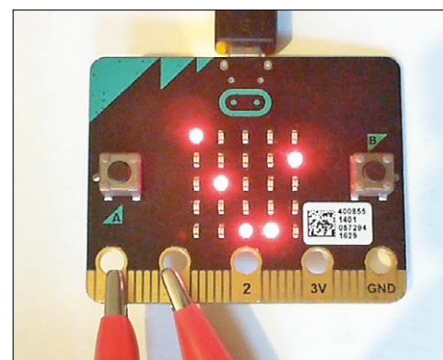


Fig. 8. Mesure du signal de test.

programmée en récepteur qui affiche les données ou les retransmet à un PC par USB. Pour utiliser MicroBitRadio dans mbed, il faudra en revanche désactiver Bluetooth Low Energy. La documentation de la BBC micro:bit fournit un avertissement à ce sujet :

« *It is not currently possible to run the MicroBitRadio component and Bluetooth Low Energy (BLE) at the same time. If you want to use the MicroBitRadio functionality, you need to disable the BLE*

stack on your micro:bit by compiling the runtime with #define MICROBIT_BLE_ENABLED 0 in your inc/MicroBitConfig.h file. »

Il n'est pas facile de trouver le bon endroit parmi les nombreux fichiers de l'environnement d'exécution. L'emplacement exact est le suivant : `micro-bit\microbit-dal\inc\core\MicroBitConfig.h` (voir **fig. 6**). On trouve dans ce fichier une entrée `MICROBIT_BLE_`

Code source et fichiers Gerber : où sont-ils ?



Vous êtes nombreux à chercher les fichiers associés (code, circuit imprimé...) aux articles publiés dans le magazine, dans la série de livres « 3XX circuits » ou encore dans les compilations. Malheureusement, en dépit de notre vigilance, vous tombez parfois sur des liens morts. Ceux-ci sont un dommage collatéral de l'évolution constante des sites d'Elektor et nous vous remercions de nous les signaler. Nous les corrigerons aussitôt.

Le plus sûr chemin pour trouver un fichier lié à un article publié est de commencer par repérer les six premiers chiffres de la référence mentionnée en fin d'article dans le magazine. Prenons par ex. le « Chargeur de batteries à panneaux solaires » publié en juillet 2008. Sa référence est **080225-I**. Il suffit maintenant de saisir dans votre navigateur l'URL suivant : www.elektormagazine.fr/080225

Vous accéderez automatiquement à la page associée à cet article avec tous les téléchargements disponibles.

La rédaction à votre service.



juillet 2017

- ♦ **Innovative City**
05 au 06/07 – Nice
www.innovative-city.com
- ♦ **Japan Expo Paris - festival des loisirs japonais**
06 au 09/07 – Paris
www.japan-expo-paris.com/fr/
- ♦ **Maker Faire Nantes**
07 au 09/07 – Nantes
<https://fr-fr.facebook.com/MakerFaireNantes/>
- ♦ **20^e congrès international de l'IFAC**
International Federation of Automatic Control
09 au 14/07 – Toulouse
www.ifac2017.org
- ♦ **Salon du véhicule électrique et hybride**
13 au 16/07 - Val d'Isère
www.salon-vehicule-electrique.com


Un événement oublié ?

Vous organisez une conférence, un salon... ou bien vous participez à un séminaire ou tout autre événement qui aurait sa place ici, partagez cette information avec tous les lecteurs. Envoyez-nous tous les détails à redaction@elektor.fr.

ENABLED 1 où il faut remplacer le 1 par un 0. BLE sera alors désactivé à la prochaine compilation et vous pourrez utiliser la version simplifiée MicroBitRadio. Le programme du **listage 3** envoie et reçoit des datagrammes avec des mesures. Émetteur et récepteur peuvent donc utiliser le même programme pour s'échanger des données. L'émetteur exécute une mesure sur P1 et envoie le résultat en mV. Le récepteur retransmet par USB les données reçues. Pas de changement côté PC, le traceur série peut être utilisé. Un résultat de mesure est donné à la **figure 7**. L'émetteur était alimenté par pile et se trouvait à 3 m du récepteur. La liaison fonctionne jusqu'à 10 m. Un condensateur électrolytique de 10 μ F était connecté à l'entrée analogique P1. Celui-ci s'est lentement chargé par la résistance de rappel de 10 M Ω de la carte. On observe des différences par rapport aux courbes de charge usuelles, car ce condensateur n'a plus été utilisé depuis longtemps. Dans un tel cas, un faible courant de fuite circule d'abord, qui diminue progressivement. Dans la partie droite du diagramme, la tension mesurée reste encore nettement inférieure à 1 V et ne croît que lentement.

Mini-oscilloscope avec afficheur à LED

Un oscilloscope très simple, c'est toujours mieux que pas d'oscilloscope du tout, et parfois il est plus important que l'appareil soit petit, autonome et pratique. Les mesures sont affichées sous forme graphique sur l'afficheur à 5 \times 5 LED (**listage 4**). Il est parfaitement exploitable même si on est l'heureux possesseur d'un appareil plus puissant. C'est incroyable tout ce qu'on peut observer avec un tel oscilloscope.

Le mini-oscilloscope utilise à nouveau le port 1 comme entrée analogique et en plus le port 0 comme sortie MLI. Avec une période de 500 μ s, on obtient un signal de sortie de fréquence 2 kHz. Une connexion directe à l'entrée de mesure montre les limites du convertisseur A/N (**fig. 8**). La période d'échantillonnage est clairement trop grande pour afficher les flancs raides du signal MLI. La fréquence de coupure de notre simple oscilloscope est ainsi de l'ordre de 10 kHz. Cela ne suffit pas pour un labo HF, mais ça le sera pour bon nombre de mesures simples ou d'expériences. 

(160392 – version française : Xavier Pfaff)

Listage 3. Envoi et réception de datagrammes.

```
//Radio Data
#include "MicroBit.h"
MicroBit uBit;

void onData(MicroBitEvent e)
{
    ManagedString s = uBit.radio.datagram.recv();
    uBit.serial.send (s);
    uBit.serial.send (" \r\n");
}

int main()
{
    uBit.init();
    uBit.messageBus.listen(MICROBIT_ID_RADIO, MICROBIT_RADIO_EVT_DATAGRAM, onData);
    uBit.radio.enable();
    char output[16];
    while (1) {
        int u = 3300 * uBit.io.P1.getAnalogValue() / 1023;
        itoa (u, output);
        uBit.radio.datagram.send(output);
        uBit.sleep(100);
    }
}
```

Liens

- [1] www.elektormagazine.fr/160273
- [2] www.elektormagazine.fr/160392
- [3] <https://developer.mbed.org/>

- [4] <https://lancaster-university.github.io/microbit-docs/ubit>
- [5] B. Kainka,
BBC micro:bit Tests Tricks Secrets Code, CreateSpace 2016

Listage 4. Utilisation de l'affichage à LED.

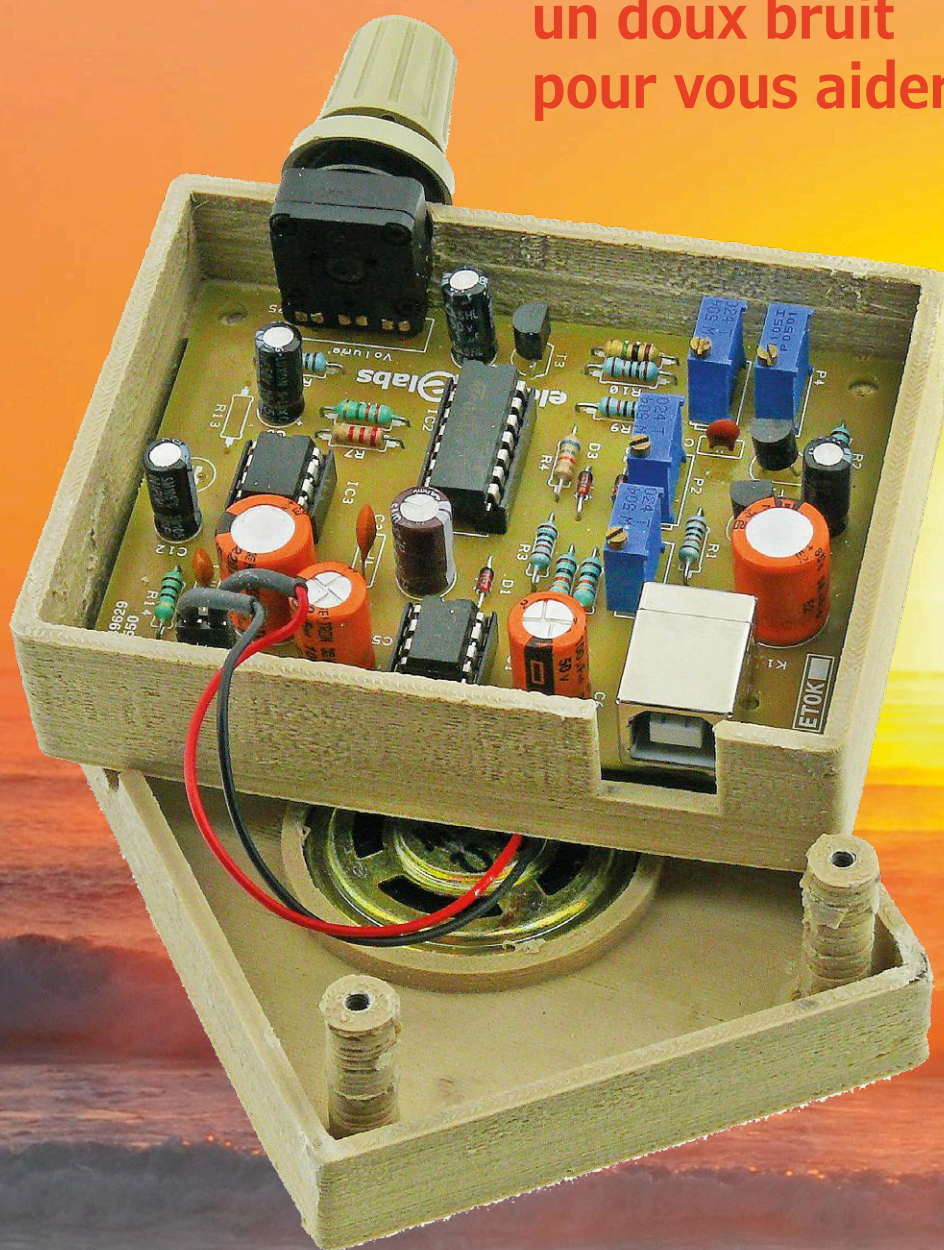
```
//LED-Scope
#include "MicroBit.h"
MicroBit uBit;

int main()
{
    int y;
    uBit.init();
    uBit.io.P0.setAnalogValue(512);
    uBit.io.P0.setAnalogPeriodUs(500);
    uBit.display.enable();
    MicroBitImage image(5,5);
    while (1) {
        for(int x = 0; x < 5; x++){
            y = 4 - (uBit.io.P1.getAnalogValue() / 205);
            image.setPixelValue(x,y,255);
        }
        uBit.display.print(image);
        uBit.sleep(500);
        image.clear();
    }
}
```


simulateur de murmure de la mer

un doux bruit
pour vous aider à vous endormir

Sunil Malekar et
Clemens Valens (labo d'Elektor)



Lorsque les bras de Morphée vous repoussent, écouter le bruit des vagues s'échouant sur la plage peut vous aider à vous détendre. Avec ce montage posé sur votre chevet, le marchand de sable s'arrêtera chez vous.

Vous pourriez penser que pour produire un magazine tel qu'Elektor, il suffit de quelques rédacteurs pour écrire des articles et d'un ou deux graphistes pour créer de belles illustrations. Ce n'est qu'en partie vrai. L'une des principales raisons – peut-être la seule – pour laquelle vous pouvez lire votre magazine

favori toutes les huit semaines est notre super-planificatrice multifonctions, une héroïne méconnue : Hedwig. Bien qu'elle ne soit pas une spécialiste de l'électronique (« Comment s'appelle ce truc bleu ? Un bidule capacitif ? »), Hedwig a un circuit fétiche : le générateur de bruit de la mer. Donc, Hedwig, celui-ci

INFO SUR LE PROJET



bruit
simulateur
relaxation



➔ débutant
connaissseur
expert



1 h env.



Fer à souder,
petit tournevis,
lit



15 € env.

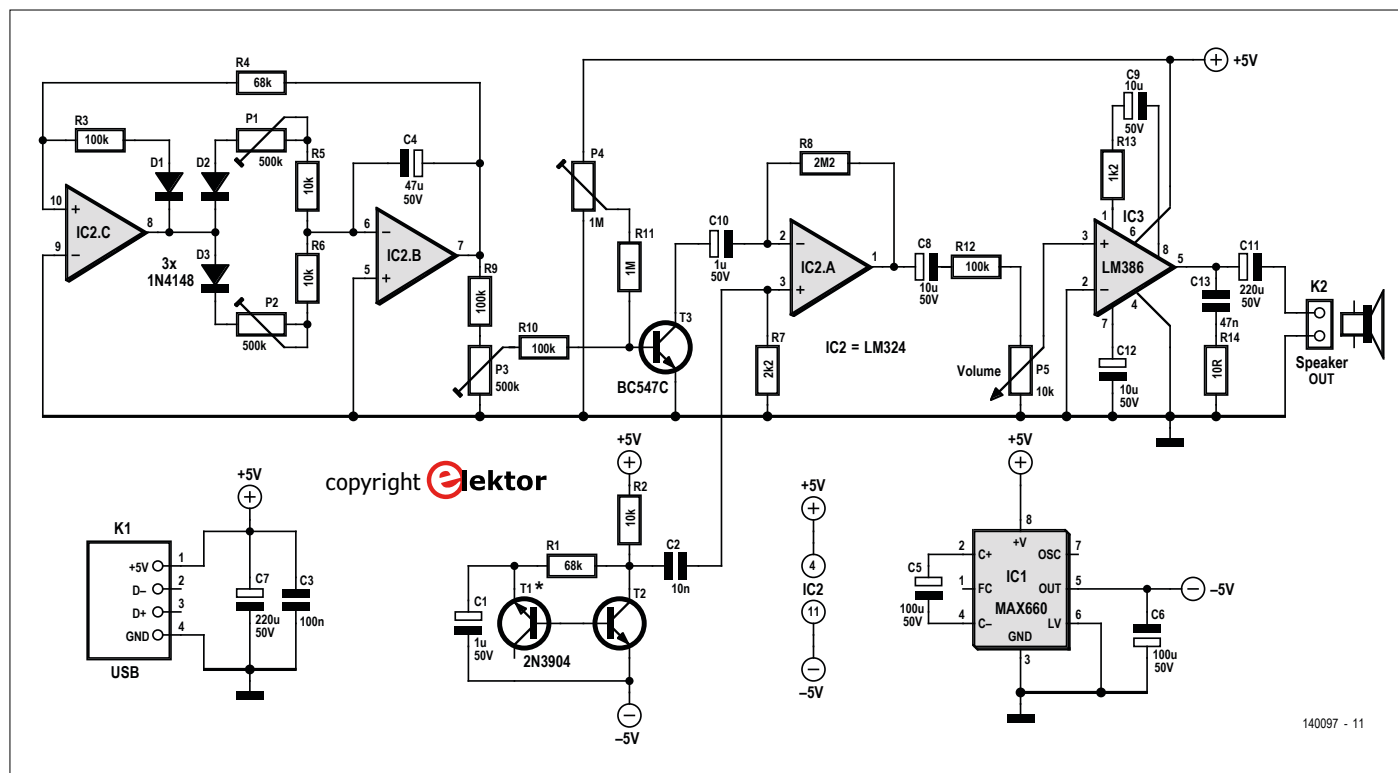


Figure 1. Les entrailles du simulateur de murmure de la mer.

Caractéristiques

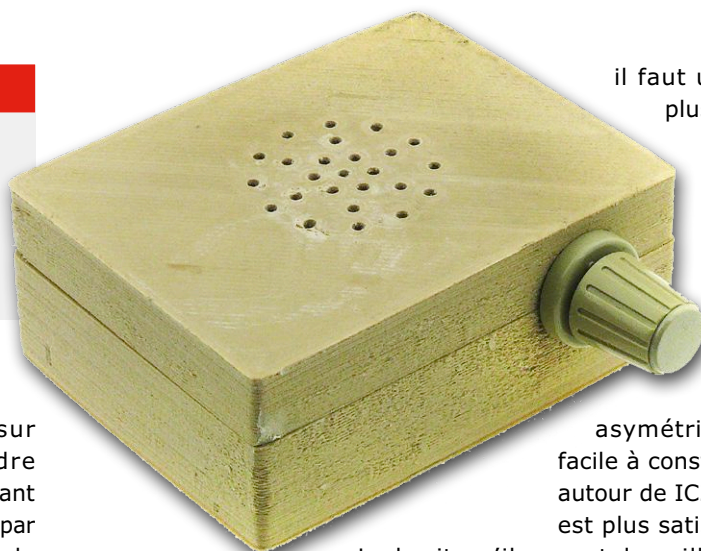
- Contrôle indépendant de l'attaque et de la chute d'une enveloppe
- Alimenté par un chargeur de téléphone
- Montage sans (0) Arduino

est pour toi.

Le son des vagues se brisant sur la plage est connu pour détendre beaucoup de personnes. Aussi relaxant et romantique soit-il, une fois analysé par un scientifique, il s'avère n'être que du bruit modulé. Peut-être un peu de bruit rose, comme une vague réminiscence du soleil couchant, mais du bruit tout de même. En réalité, en électronique, il est facile de produire du bruit, et beaucoup plus difficile de ne pas en créer intempestivement. Une diode polarisée en inverse peut suffire.

Le circuit

S'il est facile de créer du bruit, créer du bruit utilisable demande quelques efforts. La **figure 1** montre le schéma du bidule auquel nous sommes arrivés. La source de bruit est T1, monté en diode polarisée en inverse, son émetteur comme cathode, et sa base en anode.



Le bruit qu'il produit est amplifié par T2. Vous pourrez remarquer que ce transistor est connecté entre +5 V et -5 V, soit 10 V en tout. Cette « haute tension » est nécessaire pour produire un bruit adéquat. T1 et T2 sont des 2N3904, car nous avons remarqué qu'ils produisent plus de bruit que les BC547 chers à Elektor, autrefois appelés TUN. Nous les avons aussi essayés, cela a fonctionné, mais les résultats sont meilleurs avec les 2N3904. Si vous voulez essayer des BC547, retenez qu'ils n'ont pas le même brochage. Maintenant que nous avons du bruit, il faut le moduler pour simuler le son des vagues et du ressac. Pour moduler,

il faut un oscillateur. Une onde plus ou moins sinusoïdale et asymétrique peut donner un effet réaliste, mais c'est un peu compliqué, surtout à cause de l'aspect asymétrique, la montée du son n'ayant pas la même durée que sa diminution. Un générateur de triangles

asymétriques est beaucoup plus facile à construire, voyez la circuiterie autour de IC2.C et IC2.B, et le résultat est plus satisfaisant. Habituellement, un tel oscillateur produit une onde symétrique par rapport à 0, mais ce n'est pas ce que nous voulons ici, car les moitiés négatives bloqueraient notre modulateur T3, créant des coupures. D1 permet d'éviter cela. Ainsi, la sortie de IC2.B balaie bien une plage de 0 à 3 V environ.

Les potentiomètres P1 et P2 contrôlent la pente du signal modulant ; P1 règle la pente ascendante, P2 la pente descendante. Ils déterminent à eux deux la fréquence du signal, autrement dit la vitesse des vagues. Plus leur durée totale est faible, plus la fréquence sera haute. Leur rapport définit la symétrie du signal. Le transistor T3 agit en résistance commandée en courant, contrôlant le

gain de IC.2A ; plus la résistance est faible, plus le signal de sortie sera fort. Les potentiomètres P3 et P4 permettent de régler finement l'amplification. La sortie de IC.2A arrive par P5, le contrôle de volume global, à l'ampli de puissance construit autour de IC3. C'est un classique LM386 qui pilote un haut-parleur ou des écouteurs. Son gain peut être ajusté de 20 à 200 grâce à la combinaison R13/C9. Sans eux, le gain est de 20 ; avec C9 seul entre les broches 1 et 8 (soit R13 = 0 Ω), il est de 200. Les valeurs du schéma donnent un gain de 50 environ.

Bien qu'une grande partie du circuit soit alimentée en symétrique (± 5 V), un chargeur de téléphone USB 5 V standard suffit, grâce à IC1, un MAX660 monté en inverseur de tension à capacités commutées, et au connecteur USB K1.

Réglage

Comme tout instrument de haute précision, notre simulateur de murmure de la mer possède quelques potentiomètres qu'il convient d'ajuster, afin d'arriver au meilleur résultat. Heureusement, la procédure est simple. On commence par mettre P3 au minimum (curseur à 0 V). P4 contrôle le bruit de fond de la mer virtuelle quand il n'y a pas de vagues. Ajustez P4 pour obtenir un beau et doux murmure dans le haut-parleur.

Réglez ensuite P3 pour ajouter les vagues. Il contrôle efficacement l'amplitude des vagues – réglez-le simplement à un niveau réaliste. Si le son semble commencer à écrêter, baissez un peu P4. Rappelez-vous que P3 et P4 sont interdépendants, cela peut nécessiter une légère retouche des réglages.



Finalement, la « forme » des vagues doit être ajustée avec P1 et P2. Des études et simulations approfondies au labo ont montré que la montée de la vague (P1) est habituellement 5 à 10 fois plus courte que sa chute (P2) ; la forme d'onde requise est donc plus une dent de scie qu'un triangle. La période de la vague devrait être de quelques secondes – ajustez selon vos goûts. ◀

(140097 – version française : Guy Matt)

Lien

[1] www.elektormagazine.fr/140097



LISTE DES COMPOSANTS

Résistances

R1, R4 = 68 k Ω
 R2, R5, R6 = 10 k Ω
 R3, R9, R10, R12 = 100 k Ω
 R7 = 2,2 k Ω
 R8 = 2,2 M Ω
 R11 = 1 M Ω
 R13 = 1,2 k Ω
 R14 = 10 Ω
 P1, P2, P3 = 500 k Ω ajustables multitours
 P4 = 1 M Ω ajustable multitours
 P5 = 10 k Ω potentiomètre logarithmique

Condensateurs

C1, C10 = 1 μ F 50 V
 C2 = 10 nF
 C3 = 100 nF
 C4 = 47 μ F 50 V
 C5, C6 = 100 μ F 50 V
 C7, C11 = 220 μ F 50 V
 C8, C9, C12 = 10 μ F 50 V
 C13 = 47 nF

Semi-conducteurs

D1, D2, D3 = 1N4148

IC1 = MAX660

IC2 = LM324

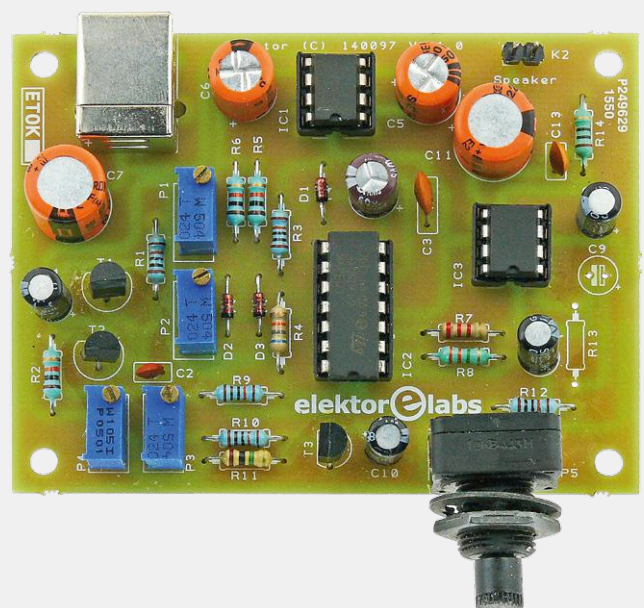
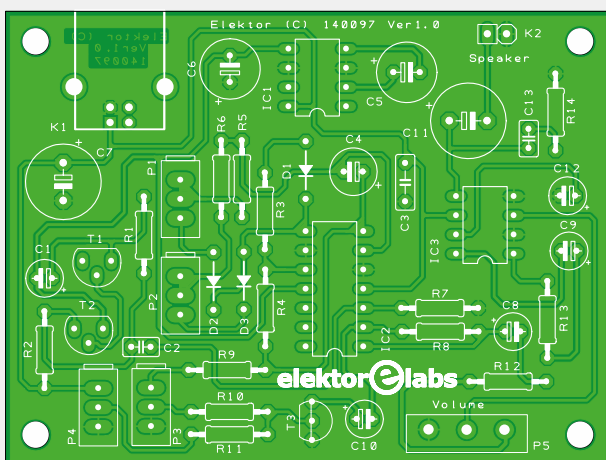
IC3 = LM386

T1, T2 = 2N3904

T3 = BC547C

Divers

K1 = connecteur USB-B
 K2 = connecteur mâle à 2 broches,
 pas de 2,54 mm
 Circuit imprimé 140097-1

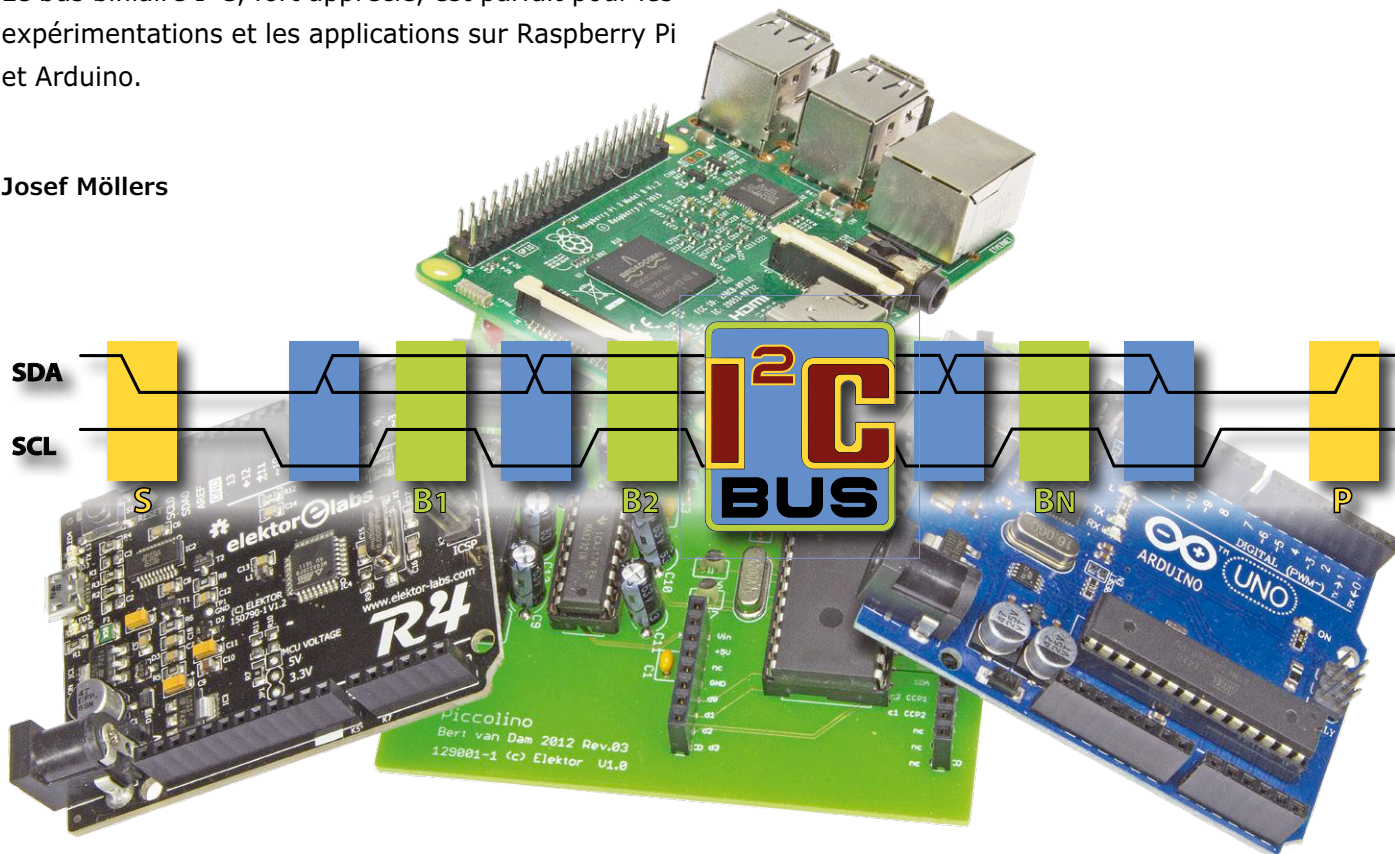


le bus I²C

2^e partie : le bus en action

Le bus bifilaire I²C, fort apprécié, est parfait pour les expérimentations et les applications sur Raspberry Pi et Arduino.

Josef Möllers



Raspberry Pi, BeagleBone, Arduino, Genuino, ATmega, PIC, presque tous les PC : à peu près tout ce qu'on trouve aujourd'hui comme outil de calcul sur la paillasse des labos ou des faiseurs est équipé d'un ou plusieurs ports I²C. À l'aide d'un capteur de température LM75, nous décrivons ici l'utilisation des interfaces I²C du Raspberry Pi, de l'ATmega et de l'Arduino en mode maître et, quand c'est possible, en mode esclave.

Raspberry Pi

Le Raspberry Pi dispose de deux bus I²C physiques, mais habituellement un seul est utilisable directement. Les résistances de rappel vers 3,3 V sont déjà présentes et actives en permanence sur le RPi. Les signaux SDA et SCL sont présents sur les broches 3 (SDA) et 5 (SCL) du connecteur GPIO, opportunément tout près des 3,3 V (broche 1), 5 V (broche 2) et masse (broche 6). Ces broches appartiennent au bus I²C 1. On peut donc installer une mini-carte d'extension pour un capteur de température ou de position, ou bien une horloge en temps réel, en occupant un minimum de place.

Attention : il est impératif que la tension sur une broche du RPi **ne dépasse pas 3,3 V**, lui appliquer 5 V peut conduire à la destruction du RPi ! C'est pourquoi on doit contrôler soigneusement tout circuit avant de le connecter aux broches

du RPi. En particulier, il faut s'assurer qu'un esclave I²C ne possède pas de résistances de rappel vers le 5 V. Au besoin, il faut les supprimer, elles n'ont de toute façon aucune raison d'être.

Grâce à un environnement confortable, le RPi se prête bien à la prise en main de nouveaux esclaves, jusqu'alors inconnus. La **figure 1** montre le raccordement au RPi d'un LM75 installé sur une plaque d'essai.

Mode maître

Pour pouvoir utiliser le bus I²C du RPi sous Raspbian, il faut commencer par charger deux pilotes additionnels. Pour cela, on lance `raspi-config` et on sélectionne dans la rubrique **Advanced Options** la sous-rubrique **I2C** pour activer l'interface et charger les modules du noyau. Autre méthode, on ajoute avec l'éditeur de texte les lignes :

```
i2c-dev
i2c-bcm2708
```

au fichier `/etc/modules`.

Après un réamorçage, les deux pilotes (ainsi que d'autres pilotes éventuellement nécessaires) sont chargés et les liens vers les

périphériques `/dev/i2c-<n>` existent, ce qu'on peut vérifier avec les commandes (`$...`) suivantes :

```
$ lsmod | fgrep i2c_
i2c_dev      XXXX  0
i2c_bcm2708  YYYY  0
$ ls /dev/i2c-*
dev/i2c-1
```

Les nombres XXXX et YYYY indiquent la taille des modules, les deux zéros le fait qu'ils ne sont encore utilisés par personne. Ces commandes fonctionnent aussi en mode non-privilégié. On installe ensuite le paquet `i2c-tools` :

```
sudo apt-get install i2c-tools
```

qui contient entre autres un détecteur de bus et de modules I²C. Raspbian contient déjà les pilotes de certains périphériques I²C, comme l'horloge en temps réel (RTC) RV-8523. Par défaut, le RPi gère le bus I²C dans le mode standard, à la vitesse de 100 kbits par seconde.

Pour commencer, on peut, en utilisant l'outil `i2cdetect` à la console, obtenir une vue d'ensemble des esclaves présents sur le bus I²C numéro 1 et reconnus par le RPi. Le résultat est celui de la **figure 2** si le LM75 répond présent à l'adresse `0x48`. Avec les commandes `i2cget`, `i2cset` et `i2cdump`, on peut communiquer avec le LM75 sans avoir à programmer soi-même (`0x00` est le numéro du registre et il doit toujours être indiqué) :

```
pi@raspberrypi ~ $ i2cget -y 1 0x48 0x00 w
0xa010
```

Il faut permuter les deux octets de la réponse (`0x10a0`). Seuls les neuf bits de poids fort de la réponse sont significatifs (`0x021`). Le LM75 fournit la température par incréments de 0,5 K, on a donc mesuré une température de 16,5 °C.

Théoriquement, le matériel du RPi peut aussi fonctionner en mode esclave, mais ce mode n'est pas supporté par le pilote Linux.

Programmation en C et Python

Pour la programmation **en C**, on a besoin de cinq fonctions :

- `open()` pour l'accès au périphérique (ou port) I²C,
- `ioctl()` pour la configuration des paramètres de l'esclave I²C,
- `read()` et `write()` pour l'échange de données avec l'esclave,
- `close()` pour terminer la session (fermer l'accès au périphérique I²C).

Pour définir l'argument `I2C_SLAVE` utilisé ci-dessous, il faut ajouter le fichier `linux/i2c-dev.h` à la liste des fichiers inclus :

```
# include <linux/i2c-dev.h>
```

On ouvre l'accès au port I²C avec `open()` :

```
fd = open("/dev/i2c-1", O_RDWR);
```

Ensuite on sélectionne l'adresse de l'esclave par un appel à

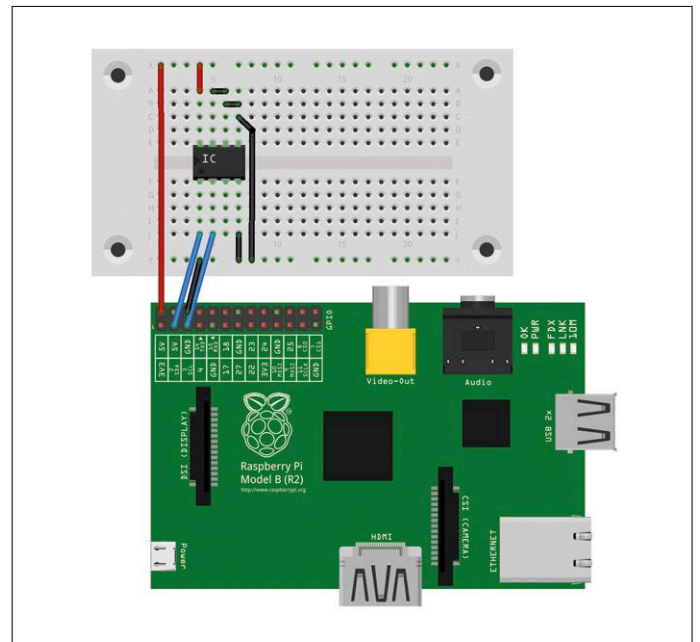


Figure 1. Connexion d'un LM75 sur carte d'essai avec un Raspberry Pi.

```
ioctl() :
```

```
ioctl(fd, I2C_SLAVE, 0x48);
```

Maintenant, nous pouvons lire et écrire :

```
unsigned char buf[2];
float T;
buf[0] = 0;
write(fd, buf, 1); /* write register number 0 */
read(fd, buf, 2); /* read temperature register */
T = ((buf[0]<< 8) | buf[1]) / 256.0;
```

Enfin, on termine la session :

```
close(fd);
```

Bien entendu, pour détecter les erreurs, il faut améliorer le traitement des valeurs retournées par les fonctions appelées. Par exemple pour savoir si l'on dispose du droit d'ouvrir `/dev/`

```
pi@raspberrypi ~ $ i2cdetect -y 1
    0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  48  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
```

Figure 2. `i2cdetect` découvre le LM75 à l'adresse 48.

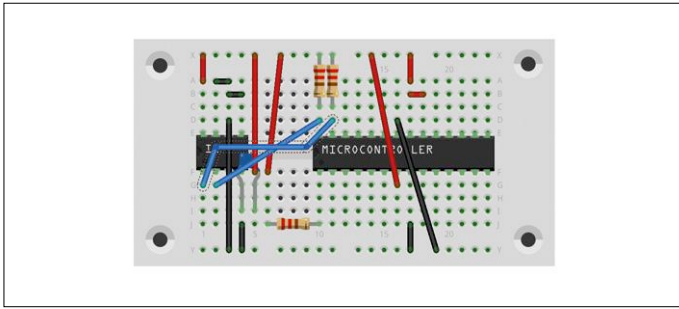


Figure 3. L'ATmega88 en compagnie du capteur sur une carte d'essai.

`i2c-1`, s'il existe un esclave à l'adresse `0x48` ou si l'échange de données a réussi.

Pour programmer le bus I²C **en Python**, il faut avoir installé le paquet Raspbian `python-smbus` :

```
sudo apt-get install python-smbus
```

On peut alors lire la température mesurée par le LM75 :

```
#!/usr/bin/python
import smbus
import time
```

```
bus = smbus.SMBus(1)
address = 0x48
```

Listage 1. Boucle principale de lecture du capteur LM75.

```
# include <i2cmaster.h>

# define LM75 (0x48 << 1)           // see datasheet

int
main(void)
{
    unsigned char val[2];

    i2c_init();

                                // initialisation I2C

    i2c_start(LM75 | I2C_WRITE);

                                // addressing to write

    i2c_write(0x00);

                                // temperature register

    i2c_rep_start(LM75 | I2C_READ);

                                // addressing to read

    val[0] = i2c_read();

                                // degrees Celsius

    val[1] = i2c_read();

                                // tenth part bit

    i2c_stop();

                                // ready

    for(;;);
}
```

```
w = bus.read_word_data(address, 0)
print format(w, '04x')
```

Le « 0 » dans l'appel de la méthode `read_word_data` indique le numéro du registre interne de l'esclave, ici le registre de température (le protocole SMBus qui encapsule l'I²C exige l'envoi préalable d'un numéro de registre). Cela peut poser problème, par ex. pour la puce d'expansion d'entrées-sorties PCF8574 qui ne possède pas de registre d'adresses !

La valeur lue pose le même problème que `i2cget` : il faut permuter les deux octets du mot `w`.

ATmega

Les microcontrôleurs ATmega d'Atmel disposent d'un contrôleur I²C intégré qui gère le mode standard (100 kHz) ainsi que le mode rapide (400 kHz) et qui peut être utilisé en mode maître, esclave ou combiné. Les ATmega324PB et ATmega328PB possèdent même deux bus I²C. Dans les exemples suivants, j'utilise un ATmega88 enfiché sur la platine d'essai à côté du capteur de température (**fig. 3**).

Ici, le mieux est d'utiliser la bibliothèque `i2cmaster` de Peter Fleury [1], mais en notant que cette bibliothèque n'active pas les résistances internes de rappel vers le haut. Il faut donc fournir ces résistances soi-même. Elles apparaissent en haut de la figure 3, au niveau de la ligne d'alimentation 5 V. La bibliothèque configure le port I²C dans le mode standard (100 kHz). L'accès au capteur LM75 prend alors la forme du **listage 1**.

Avec la bibliothèque `lcdlibrary` du même auteur, on peut réaliser un thermomètre numérique rien qu'en ajoutant un afficheur à cristaux liquides. On peut aussi utiliser une carte d'interface I²C (qui embarque le plus souvent un PCF8574) ou bien directement un afficheur avec une interface I²C intégrée. Le contrôleur I²C intégré de la série ATmega n'est pas seulement capable de fonctionner en mode maître pour gérer des esclaves, il peut aussi travailler en mode esclave, sous le contrôle d'un maître. Il peut être maître pour communiquer avec un LM75, puis esclave pour être lu par un RPi. Utiliser l'ATmega comme esclave peut être intéressant dans le cas où celui-ci acquiert des données sur ses broches d'entrées-sorties et les soumet à un prétraitement avant de les faire suivre à un RPi.

Mais ce mode de fonctionnement est plus complexe et en interaction plus étroite avec le reste du logiciel. La bibliothèque disponible en [2] est un bon point de départ puisqu'elle permet d'utiliser le contrôleur I²C des ATmega en esclave.

Le déroulement décrit ci-dessous suit les états décrits dans les feuilles de caractéristiques d'Atmel. Les tableaux correspondants se trouvent dans chaque feuille de caractéristiques d'Atmel, ainsi qu'au chapitre 22.7 du manuel de l'ATmega48/88/168. Pour une meilleure efficacité, l'exécution du code est pilotée par interruptions. L'unité de contrôle I²C provoque une interruption dans les conditions suivantes :

- après l'émission d'une *START condition* ou d'une *Repeated Start condition*,
- après l'émission de l'adresse et du bit de lecture/écriture,
- après l'envoi d'un octet de données,
- quand l'ATmega a perdu l'arbitrage (lorsqu'il se produit une collision lors de l'envoi d'une *START condition* ou de l'octet d'adresse ou du bit de lecture/écriture),
- quand l'ATmega a reconnu une *STOP condition* alors qu'il a

- été adressé comme esclave
- quand l'ATmega a reçu un octet de données
- quand l'ATmega a reconnu une *START condition* ou une *Repeated Start condition* alors qu'il a été adressé (de nouveau) comme esclave,
- quand une transaction invalide a été détectée sur le bus

Comme les quatre premières conditions concernent le mode maître, nous ne nous intéresserons ici qu'aux quatre dernières.

L'unité de contrôle I²C doit être initialisée avec l'adresse d'esclave désirée. On peut ensuite l'activer et commencer le travail :

```
TWAR = (I2C_Slave_Addr << 1);
TWCR = _BM(TWEA) | _BM(TWEN) |
        _BM(TWIE);
```

Il n'y a pas lieu de régler la vitesse en mode esclave, étant donné qu'elle est imposée par le maître.

Si une interruption se présente, il faut en déterminer la raison. Pour cela, on lit le registre d'état *TWSR* et on exploite les cinq bits de poids fort.

Le **listage 2** présente une partie de la routine de service de l'interruption (ISR) pour l'adressage comme récepteur, comme émetteur, ainsi que pour la *STOP condition*.

Les descriptions et les tableaux des feuilles de caractéristiques ATmega sont très complets, aussi bien pour l'explication des codes d'état que pour les réponses requises du programme et les actions matérielles qu'elles provoquent.

Arduino

Comme beaucoup de cartes Arduino sont basées sur des microcontrôleurs de la famille ATmega, toutes les explications ci-dessus restent en principe valables. Toutefois l'Arduino bénéficie d'une bibliothèque souvent utilisée et très pratique du nom de *Wire*.

Pour l'Arduino, le raccordement est aussi simple que pour le RPi (**fig. 4**). Selon le type (ou le clone) d'Arduino, il faut prendre garde à ce que les broches d'un Arduino alimenté en 3,3 V ne soient jamais soumises à une tension de 5 V. Donc, mesurez d'abord, connectez ensuite et, le cas échéant, supprimez les résistances de rappel vers le haut de l'esclave !

La bibliothèque *Wire* est déjà contenue dans l'Environnement de Développement Intégré (EDI) ; de même que pour la bibliothèque *i2cmaster* de Peter Fleury, il faut commencer

Listage 2. Routine de traitement d'interruption de la *STOP condition*. Le code complet et abondamment commenté est à télécharger sur la page du projet [5].

```
ISR(TWI_vect)
{
    /*
     * These variables need to be preserved across interrupts
     */
    static unsigned char i2c_idx,      /* Index into twi_msg[] */
                       i2c_tosend;    /* Number of bytes to send */

    switch (TWSR & 0xf8)
    {
        ...
        /*
         * RECEIVE Code
         * See Table 19-4. Status Codes for Slave Receiver Mode
         * [Page 229]
         */
        case 0x60:
            /*
             * Own SLA+W has been received; ACK has been returned
             * TWDR: No TWDR action
             * STA=X STO=0 TWINT=1 TWEA=1
             * Data byte will be received and ACK will be returned
             */
            TWCR = (TWCR & ~_BM(TWSTO)) | (_BM(TWINT) | _BM(TWEA));
            i2c_idx = 0;
            break;

            ...

        case 0xA0:
            /*
             * A STOP condition or repeated START condition has been
             * received while still addressed as slave
             * TWDR: No action
             * TWA=0 STO=0 TWINT=1 TWEA=1
             * Switched to the not addressed Slave mode;
             * own SLA will be recognized;
             * GCA will be recognized if TWGCE = "1"
             */
            TWCR = (TWCR & ~(_BM(TWSTA) | _BM(TWSTO))) | (_BM(TWINT) | _BM(TWEA));
            break;

            ...

        case 0xA8:
            /* Own SLA+R has been received; ACK has been returned
             * TWDR: Load data byte
             */
            /*
             * The address (register number) has been received,
             * Start sending payload
             */
            TWDR = 0x42;
            break;
    }
}
```

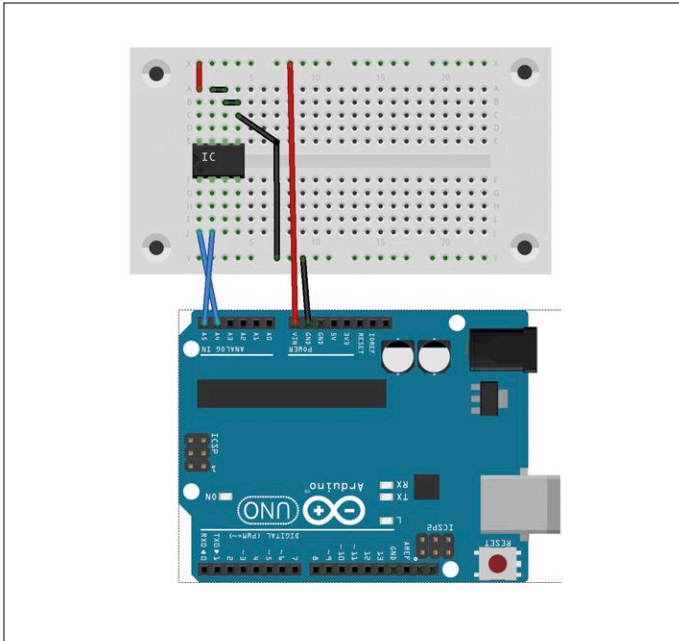


Figure 4. La connexion à l'Arduino est pratiquement identique à celle au RPi.

par initialiser la bibliothèque, c'est-à-dire le port I²C :

```
#include <Wire.h>
void setup() {
  Wire.begin();
}
```

L'appel de `Wire.begin()` active les résistances internes de rappel vers le haut. Si cela pose problème du fait des valeurs assez élevées de ces résistances, on peut leur connecter en parallèle deux résistances de 10 à 20 kΩ. Ou bien, on les désactive immédiatement et on n'utilise que des résistances externes d'environ 4,7 kΩ. Pour cela, on ajoute les deux lignes suivantes immédiatement après l'appel à `Wire.begin()` :

```
digitalWrite(SDA, 0);
digitalWrite(SCL, 0);
```

Remarque : ceci n'est pas officiel et pourrait totalement changer dans l'avenir. Le mieux est de faire une mesure au multimètre après l'initialisation.

Attention : même si l'on désactive aussitôt les résistances internes de rappel, elles ont quand même eu une courte période

Le bus I²C sur PC

Chaque PC possède une interface I²C et, le plus souvent, plusieurs. On rencontre des esclaves I²C dans les afficheurs (DDC [3]) et les barrettes de mémoires (SPD [4]). Même les capteurs de température peuvent être connectés par I²C. Malheureusement, pratiquement aucun constructeur ne fournit d'information sur les types de puces utilisées, ni comment accéder aux bus I²C de l'extérieur (à supposer que ce soit possible et qu'ils ne soient pas enfermés à l'intérieur de quelques modules).

Si l'on est sous Linux, on peut essayer de charger le module `i2c-dev` :

```
$ sudo modprobe i2c-dev
```

et voir ensuite s'il existe des périphériques `i2c` dans `/dev` :

```
$ ls /dev/i2c*
/dev/i2c-0 /dev/i2c-1 /dev/i2c-2 /
dev/i2c-3 /dev/i2c-4 /dev/i2c-5
```

Cela marche aussi en mode non-privilegié. Sous Debian et ses avatars (Raspbian et Ubuntu), il y a les outils `i2c-tools` :

```
apt-get install i2c-tools
```

Il arrive qu'un seul des bus soit doté d'esclaves :

```
root@bounty:~# i2cdetect -y 5
 0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
```

```
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  37  --  --  3a  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50: 50  --  --  --  --  --  --  --  --  58  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
```

Il n'est toutefois pas toujours très facile de savoir de quel genre d'esclaves il s'agit. Ici, on a à l'adresse 0x50 la PROM EDID de l'afficheur :

```
root@bounty:~# i2cdump -y 5 0x50
No size specified (using byte-data access)
  0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f    0123456789abcdef
00: 00 ff ff ff ff ff ff 00 1a b3 d4 07 ec 22 02 00    .....?????"?.
10: 0a 16 01 03 80 34 20 78 2a ef 95 a3 54 4c 9b 26    ??????4 x*???TL?&
20: 0f 50 54 a5 4b 00 81 80 81 00 81 0f 95 00 95 0f    ?PT?K.????.???
30: a9 40 b3 00 01 01 28 3c 80 a0 70 b0 23 40 30 20    ?@?.??(<??p?#@0
40: 36 00 06 44 21 00 00 1a 00 00 00 fd 00 38 4c 1e    6.?D!...?.8L?
50: 52 10 00 0a 20 20 20 20 20 20 00 00 00 fc 00 42    R??.? ...?.B
60: 32 34 57 2d 35 20 45 43 4f 0a 20 20 00 00 ff    24W-5 ECO? ....
70: 00 59 56 32 45 31 34 30 30 31 32 0a 20 20 00 8e    .YV2E140012? .?
```

Si vous disposez encore d'un port VGA, essayez de connecter un esclave I²C à la broche 12 (SDA) et à la broche 15 (SCL). Les broches 6 (SCL) et 7 (SDA) d'un connecteur DVI, ainsi que les broches 15 (SCL) et 16 (SDA) d'un connecteur HDMI sont également de bons candidats. Mais ces bus sont probablement sous le contrôle total de la carte graphique et de son micrologiciel.

d'activité. Cela peut avoir des conséquences graves pour un esclave 3,3 V avec des résistances externes de rappel vers 3,3 V, connecté directement à un Arduino 5 V. Dans ce cas, il faut impérativement intercaler un convertisseur de niveau ! Après l'initialisation, on peut programmer un envoi de données à un esclave I²C dans la fonction `loop()`, par ex. l'adresse du registre de température du capteur LM75 :

```
void loop() {
  Wire.beginTransmission(0x48);
  Wire.write(byte(0x00));
  Wire.endTransmission();
}
```

ou bien une lecture de données :

```
Wire.requestFrom(0x48, 2);
c1 = Wire.read();
c2 = Wire.read();
}
```

Comme on le voit, il faut indiquer à la bibliothèque `Wire` le nombre d'octets à recevoir de l'esclave (ici: 2). Avec cet appel, on en lit déjà deux. Ensuite, on peut les acquérir un par un. La bibliothèque `Wire` peut aussi être utilisée pour le mode esclave. Là aussi, il faut commencer par l'initialiser. Pour cela, on appelle `Wire.begin()` avec un paramètre représentant l'adresse choisie pour l'esclave, ce qui active le mode esclave. De plus, il faut spécifier une fonction de traitement des événements exécutée lorsque l'Arduino est adressé comme esclave, par ex. dans le cas où il doit recevoir des données :

```
#include <Wire.h>
void setup() {
  Wire.begin(0x48);
  Wire.onReceive(receive);
}
```

Ici aussi, la méthode `Wire.begin()` active les résistances de rappel vers le haut, chose plutôt inattendue pour un esclave. Il faut donc les désactiver immédiatement, comme ci-dessus.

Avertissement : cela n'empêche pas un Arduino 5 V d'être utilisable comme esclave d'un maître 3,3 V, mais une impulsion à 5 V sur des lignes 3,3 V peut détruire le maître. Si l'on veut connecter un Arduino à un RPi, on doit obligatoirement faire en sorte que la connexion électrique ne s'établisse qu'après l'initialisation de l'Arduino. Dès l'arrivée de données, le gestionnaire d'événements `receive` spécifié ci-dessus est appelé avec le nombre d'octets à lire en paramètre :

```
void receive(int n) {
  while (n-- > 0) {
    uint8_t c = Wire.read();
    // traitement de c
  }
}
```

Pour l'envoi de données au maître, on appelle, au lieu de `onReceive()`, la méthode `onRequest()` et on lui passe un gestionnaire d'événements qui met à disposition les données.

Comme la quantité de données à transmettre ou à recevoir est inconnue, ce gestionnaire n'a pas de paramètre. Les données doivent toutefois être transmises par **un** appel à `Wire.write()` :

```
#include <Wire.h>
void setup() {
  Wire.begin(0x48);
  Wire.onRequest(transmit);
}
void loop() {
  while (1) delay(1000);
}
void transmit() {
  uint8_t msg[N];
  // production du contenu de msg[]
  Wire.write(msg, N);
}
```

Ces deux méthodes peuvent être combinées dans un croquis :

```
#include <Wire.h>
void setup() {
  Wire.begin(0x48);
  Wire.onReceive(receive);
  Wire.onRequest(transmit);
}
```

On peut ainsi reproduire un LM75 précis, utiliser un capteur 1-Wire (DS18B20) à la place d'un LM75 ou émuler une horloge en temps réel qui reçoit l'heure par DCF77 ou GPS.

Dans la troisième partie

La partie suivante de la série présentera quelques composants I²C très appréciés : outre le capteur de température LM75, la puce d'expansion d'entrées-sorties PCF8574 et l'horloge en temps réel RV8523. Cette petite série d'articles se conclura sur le thème de la recherche et de l'analyse d'erreurs avec des outils appropriés et à la portée d'un labo d'amateur. ◀

(160418 – version française : Helmut Müller)

Liens

- [1] <http://homepage.hispeed.ch/peterfleury/avr-software.html>
- [2] www.mikrocontroller.net/topic/182638
- [3] https://en.wikipedia.org/wiki/Display_Data_Channel
- [4] https://en.wikipedia.org/wiki/Serial_presence_detect
- [5] www.elektormagazine.fr/160148

des circuits vraiment imprimés

imprimante de carte/four à refusion Voltera V-One



Karl-Ludwig Butte (Allemagne)

Figure 1. Vue latérale de la Voltera V-One.

Avant, quand on voulait graver soi-même son circuit imprimé, c'était une débauche de produits chimiques. Il fallait d'abord transférer le dessin du circuit sur un film transparent puis sur la carte par une technique photographique : insolation-développement-gravure. Et la gravure du cuivre, toujours chimique, était synonyme de taches et de mauvaises odeurs. Ça pourrait bien changer : la Voltera V-One imprime les pistes, applique la pâte à souder et peut même souder les composants selon un procédé par refusion.

Tous ces produits chimiques ont fini par détourner de nombreux concepteurs de la réalisation des cartes de leurs prototypes. Les manipulations fastidieuses commençaient avec le typon : il fallait déterminer les durées d'exposition et de développement de façon empirique. L'idée était de trouver le bon compromis entre l'opacité du noir (pistes de cuivre) et la transparence du reste de la surface. Plus le noir était opaque, plus les surfaces censées rester transparentes viraient au gris et le cuivre n'était pas complètement attaqué là où il aurait dû disparaître. À l'inverse, si les zones transparentes restaient bien claires, le noir n'était pas assez profond et les pistes fines se trouvaient en partie attaquées.

Une fois le film réalisé de façon satisfaisante, on pouvait enfin graver la platine. Celle-ci était à son tour insolée, développée puis gravée. Normalement, après élimination de la couche de résine photosensible et résistante à la gravure, des pistes bien brillantes apparaissaient. Au lieu de cartes à circuit imprimé

(*Printed Circuit Boards* ou PCB), on aurait mieux fait de parler de cartes à circuit gravé (*Etched Circuit Boards* ou ECB). Avec son imprimante V-One, il se pourrait bien que la *startup* canadienne Voltera révolutionne la fabrication des circuits imprimés et conduise bien plus d'électroniciens à réaliser eux-mêmes leurs cartes. Une technique en phase avec le 21^e siècle, car la V-One ne se contente pas d'imprimer les pistes, elle applique aussi la pâte à souder et peut souder par refusion les composants placés manuellement.

Le matériel

La V-One, c'est une imprimante à plat avec une commande à trois axes. Elle trouve sa place sans difficulté sur un bureau (**fig. 1**). La **figure 2** présente la réalisation mécanique : un socle accueillant un pont mobile (axe des Y). Le chariot mobile (*carriage*) du pont comporte un logement pour recevoir trois outils différents. Ce chariot parcourt l'axe des X. En outre, les

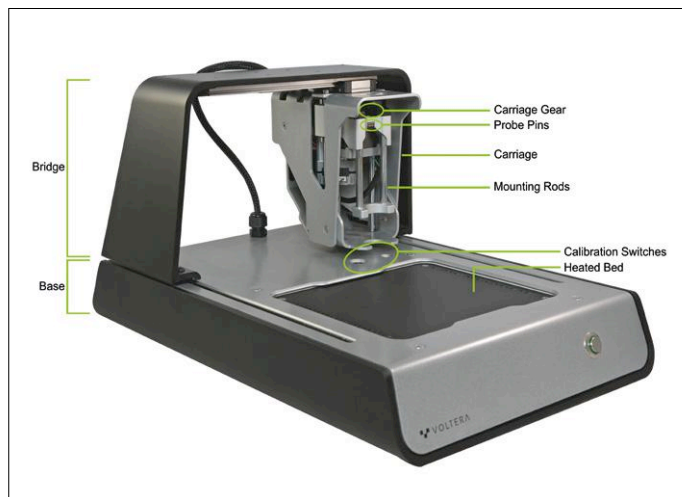


Figure 2. Architecture de la V-One.



Figure 3. Outils de la Voltera V-One.

outils peuvent se déplacer verticalement (axe des Z).

Les trois outils (**fig. 3**) sont un palpeur, une tête pour déposer l'encre conductrice et une tête pour déposer la pâte à souder. Les trois outils sont très faciles à échanger, car ils ne sont fixés au porte-outil qu'au moyen de quatre puissants petits aimants. La carte à imprimer (FR4) est maintenue sur le plateau à l'aide d'un système de plaques de serrage. Un système de chauffage est monté au-dessous du plateau. Il permet non seulement de fixer l'encre des pistes imprimées, mais également de souder les CMS par refusion. La V-One ne se contente pas d'imprimer vos cartes, elle fait également office de four à refusion ! Comme la V-One dépose aussi la pâte à souder avec précision sur les pastilles, vous n'avez plus besoin de pochoir (*stencil*). Avec une V-One, tout électronicien peut réaliser facilement des montages CMS de qualité professionnelle.

Les fentes le long desquelles le pont se déplace sur l'axe des Y sont éclairées du dessous par deux rangées de LED RVB (**fig. 4**). Il s'agit non seulement de produire un effet spécial, mais également de signaler la température du plateau : rouge veut dire « ne pas toucher ». La plaque peut atteindre 250 °C : brûlures garanties. Le bleu indique qu'une phase thermique (séchage ou refusion) est terminée et que l'on peut retirer la carte.

L'imprimante a une interface USB. Pour se servir de l'imprimante sous Windows de Microsoft, il faut à installer le pilote que l'on peut télécharger depuis le site de Voltera.

Le logiciel

Le logiciel est une solution complète : *calibrage - commande - impression - instructions vidéo*. La dernière partie mérite des louanges : chaque étape de travail est expliquée au moyen d'une ou plusieurs courtes vidéos et chaque manipulation est présentée de manière pratique (**fig. 5**). Ces vidéos s'enchaînent et vous aident à reproduire chaque manipulation sur votre machine. Difficile de faire plus simple et plus clair !

Le logiciel est disponible pour Windows 7 (64 bits) de Microsoft (ou plus récent) et pour OSX (à partir de la version 10.11). Une version Linux se fait encore attendre. À l'heure actuelle, les logiciels sont uniquement téléchargeables : le fabricant a abandonné les antiques médias optiques. À chaque démarrage, le logiciel vérifie la disponibilité éventuelle de mises à jour. Selon le cas, le logiciel ou le microprogramme de l'imprimante sont mis à jour.

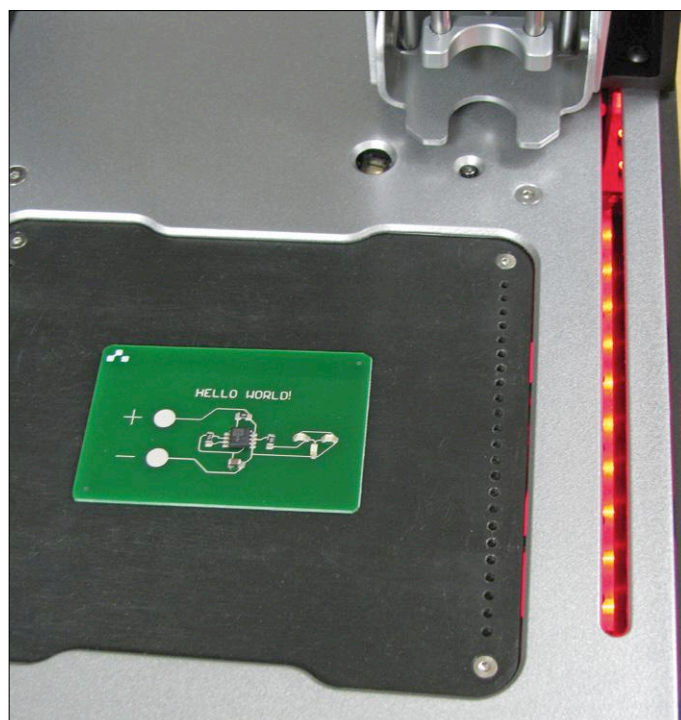


Figure 4. Couleur des LED : quand les fentes de l'axe des Y diffusent du rouge, le plateau est à haute température.

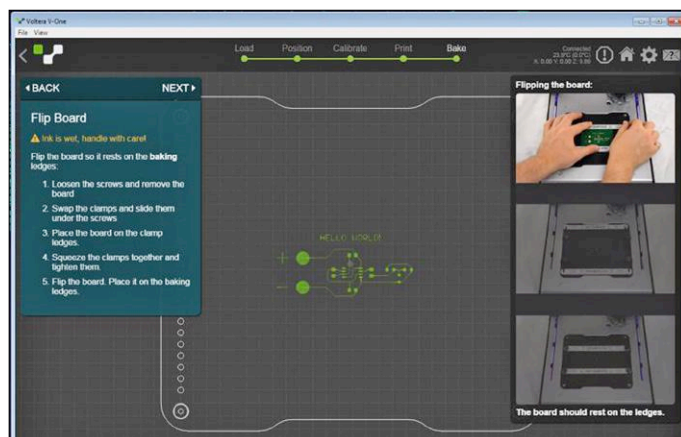


Figure 5. Le logiciel V-One comporte de très bons tutoriels en vidéo.

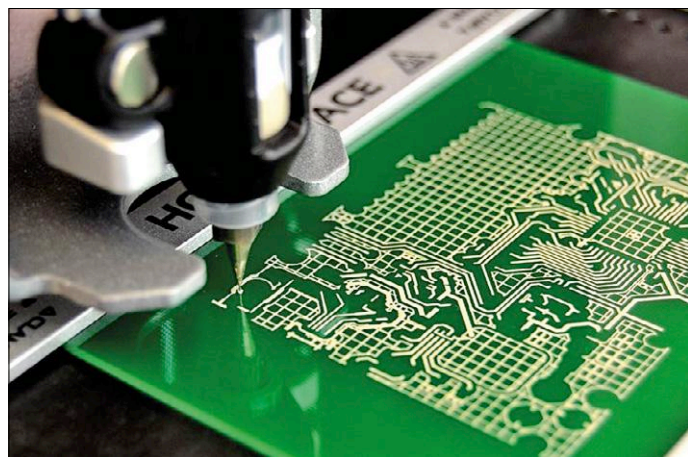


Figure 6. La V-One en action : pistes conductrices imprimées.

Prise en main de la V-One

La Voltera V-One accepte uniquement les fichiers Gerber, mais la plupart des programmes de dessin de circuits imprimés actuels peuvent exporter dans ce format. Voltera a testé les versions Gerber produites par Eagle, Altium, Upverter et KiCad. Les projets d'Elektor sont très souvent assortis de fichiers de ce type à télécharger. Pour pouvoir essayer l'imprimante immédiatement, Voltera livre toutes les pièces nécessaires à l'impression d'une carte « Hello World », c'est une délicate attention. Il s'agit de faire clignoter une LED avec l'insubmersible *timer* NE555 en version CMS. Le dessin du circuit est intégré au logiciel et accessible en sélectionnant un lien particulier dans la fenêtre de dialogue « ouvrir fichier ».

Première impression

Les cartes de résine époxyde renforcée de fibre de verre de type FR4 conviennent parfaitement comme matériau pour les circuits imprimés. Ma V-One a été livrée avec un jeu de dix plaques de 2 x 3" (50 x 76 mm) et 3 x 4" (76 x 102 mm) ainsi qu'une carte d'extension pour Arduino Uno ou Mega. La surface maximale d'impression est de 5,5 x 4" (138 x 102 mm). Comme les traces de graisse cutanée peuvent être préjudiciables, je nettoie toujours les cartes vierges à l'alcool.

Pour commencer, il faut fixer la carte vierge au milieu du plateau à l'aide de plaques de serrage métalliques. Ensuite, le palpeur reconnaît le rectangle à l'intérieur duquel le circuit doit être imprimé. Si le rectangle n'est pas centré sur la platine ou en dehors des limites de celle-ci, on a tout le loisir de corriger le positionnement sur l'écran. Cela permet également d'imprimer plusieurs petites platines sur une même plaque d'époxy pour les découper ultérieurement.

Dès que le positionnement est correct, le palpeur analyse la hauteur. Les buses d'application de l'encre conductrice et de la pâte à souder sont très fines et très fragiles. Elles ne doivent en aucun cas heurter la surface de la carte, mais elles ne doivent pas non plus la survoler de trop haut, sinon la qualité de l'impression en souffrirait.

Une fois l'analyse de hauteur terminée, il faut remplacer le palpeur par la tête d'impression contenant l'encre conductrice. Ensuite, un dessin de test est imprimé : deux lignes sinusoïdales sont imprimées ainsi que des traits parallèles, séparés du minimum de 0,8 mm. Ces pistes ont une épaisseur mini-

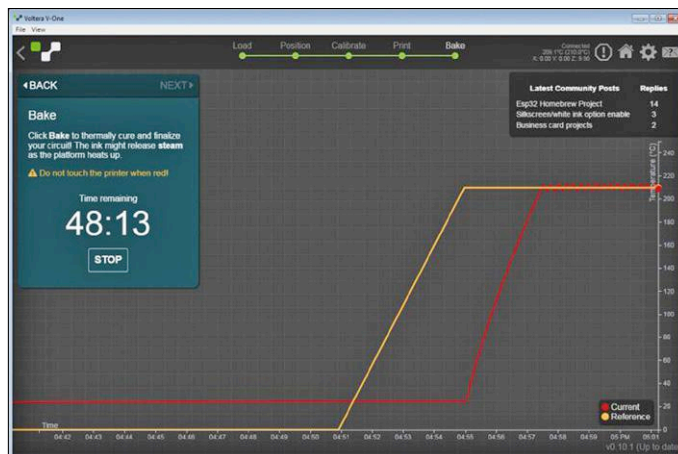


Figure 7. Évolution de la température (consigne et mesure) lors du séchage de l'encre.

male de 8 mil. (~0,2 mm). Si ce dessin n'a pas été imprimé correctement, il est possible de régler le débit d'encre par logiciel et après l'avoir effacé, de l'imprimer à nouveau. D'après le constructeur, l'encre conductrice et la pâte à souder doivent être conservées au réfrigérateur et placées à température ambiante 30 min avant utilisation. J'ai cependant observé qu'en portant la durée de mise en température à 2 h, j'obtenais des pistes plus uniformes et surtout sans interruption. Si le dessin de test est correct, vous n'avez plus qu'à l'essuyer à l'aide de papier absorbant et à nettoyer la platine à l'alcool.

Le spectacle de l'imprimante au travail a toujours quelque chose de fascinant (fig. 6). La tête d'impression ne travaille pas ligne par ligne dans un mouvement de va-et-vient comme celle d'une imprimante à jet d'encre. Elle travaille comme celle d'un traceur XY en imprimant les pistes entières les unes après les autres et termine avec les pastilles.

Si à certains endroits l'impression ne paraît pas bonne (il peut y avoir de nombreuses raisons à cela), il est toujours possible de choisir des parties à répéter depuis l'écran, le cas échéant en retouchant l'apport d'encre.

► Impression et soudage par refusion automatiques

Séchage de l'encre

Une fois l'impression terminée, le processus de séchage (ou durcissement) de l'encre encore humide peut commencer. Il faut rester très attentif pour ne pas effacer des pistes par inadvertance tant que l'encre n'est pas fixée. Pour le séchage, les plaques de serrage sont permutées et la carte retournée (pistes vers le bas) prend place dans les évidements des plaques de serrage. La surface inférieure de la carte est alors maintenue à une faible distance au-dessus du plateau.

Le séchage est lancé par le logiciel et dure de 40 à 50 min. Pendant ce processus, les valeurs et les courbes de la température de consigne et de la température instantanée du système de chauffage s'affichent sur l'écran (fig. 7). Les fentes de guidage du pont virent graduellement à un rouge intense.

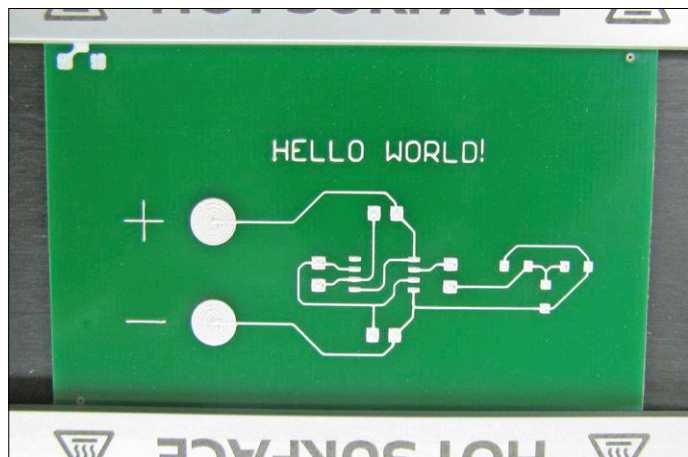


Figure 8. Après le séchage des pistes, le circuit imprimé est poli.

À 250 °C, il est formellement déconseillé de toucher l'imprimante, les brûlures seraient sévères. Par précaution, j'ai d'ailleurs installé ma V-One sur une double épaisseur de carreaux en céramique. Quand les LED repassent au bleu, le système de chauffage a refroidi et le séchage est terminé. La carte peut être retirée sans risque.

La V-One est livrée avec une éponge synthétique spéciale permettant de polir les pistes pour garantir des soudures de qualité. La **figure 8** montre un circuit imprimé poli.

Dépôt de la pâte à souder

Avant de commencer à déposer la pâte à souder, il faut fixer à nouveau la carte sur le plateau ; ensuite on doit indiquer à l'imprimante la position exacte des deux points de référence. Cela ne pose pas de problème grâce au palpeur et aux fonctions logicielles adéquates. Pour finir, il suffit de remplacer le palpeur par la tête d'application de la pâte à souder et de lancer la procédure.

Mise en place des composants et soudage

À ce jour, le placement des composants reste manuel. De bonnes brucelles sont indispensables. Il faut être très attentif à ne pas étaler ou essuyer la pâte à souder et à respecter le sens des composants polarisés (condensateurs chimiques, diodes, circuits intégrés, etc.). Une fois tous les composants positionnés, il faut placer la carte au centre du plateau et démarrer le processus de refusion (fig. 4). Lorsque les LED sont redevenues bleues, la carte est prête et peut être testée (**fig. 9**). Les fils de raccordement au connecteur de la pile de 9 V ont été soudés manuellement : c'est l'objet du paragraphe suivant.

Traitement des composants classiques

Avec la V-One, peut-on utiliser d'autres composants que les CMS ? Oui : les composants classiques à pattes/fils peuvent être soudés à la main. Pour cela on utilise les pastilles habituelles avec trou central. Au lieu de déposer automatiquement la pâte à souder, après avoir séché et poli la carte, il faut effectuer à la main le perçage de la carte et y placer les composants. En ce qui concerne le soudage, l'encre conductrice se comporte différemment du cuivre. Il faut s'y habituer. Il faut impérativement une station de soudage régulée pour souder à une température de 190 à 210 °C. La soudure recommandée par Voltera

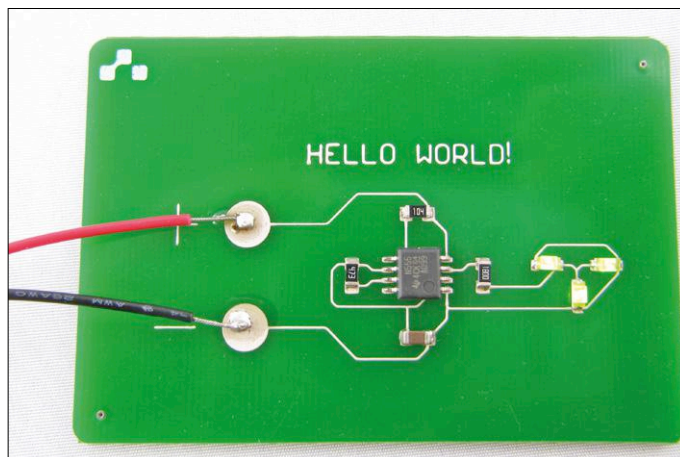


Figure 9. La carte soudée par la V-One est prête, il n'y a plus qu'à la tester.

(Sn62) fonctionne correctement et la SnBiAg1 encore mieux. Le fil de soudure est si fin qu'il n'est pas fourré. Le flux de soudure (par ex. ChipQuick SMD 291) doit être déposé avant. J'ai pu me rendre compte que cela demande un peu de pratique.

Résumé

Avec la Voltera V-One, la gravure des circuits imprimés va être reléguée au rang d'antiquité. Une fois que vous l'aurez essayée, l'électronicien que vous êtes ne voudra plus s'en passer. En particulier si vous concevez des cartes, c'est un gros avantage de ne plus avoir à attendre des jours, voire des semaines que les circuits imprimés des prototypes soient livrés. La capacité de soudage par refusion est un plus décisif et le soudage des CMS devient une routine. Selon le tarif donné par Voltera, l'imprimante V-One coûte 2 000 € ce qui est assez cher, les consommables le sont aussi, mais à mon avis cela vaut vraiment le coup. ◀

(160384 – version française : Yves Georges)

Lien

[1] www.voltera.io

À propos de l'auteur

Karl-Ludwig Butte a étudié l'informatique (branche économie) à l'école supérieure de Fulda et travaille actuellement dans l'industrie automobile sur les processus et les systèmes informatiques. Pendant ses loisirs, il conçoit des projets électroniques, et se passionne pour les ordinateurs et l'histoire de l'informatique.

nWatch,

un système de développement au poignet
ou comment fabriquer votre montre intelligente



Piotrek Wasilewski (Pologne)

À l'origine conçue pour tricher à l'école, cette montre est devenue de plus en plus intelligente. En la faisant évoluer, l'auteur a accidentellement appris pas mal de choses en chemin. Gadget à la mode ou plateforme éducative ? Construisez-la et faites-vous une opinion.

L'objet baptisé nWatch présenté ici est une montre intelligente que vous pourrez utiliser comme carte de développement pour microcontrôleur. Au début, ce projet n'était pas censé être une carte de développement, mais simplement une montre pour m'aider à tricher aux contrôles de connaissances à l'école dans les matières qui, à mon humble avis, n'étaient pas pertinentes. Le premier prototype n'était pas très avancé, mais remplissait sa fonction. J'ai ensuite commencé à penser à la possibilité de développer un système qui serait utile et me permettrait d'apprendre quelque chose, c'est ainsi que j'ai eu l'idée de fabriquer une montre intelligente utilisable comme carte de développement pour microcontrôleur. L'exercice s'est cependant avéré plus difficile que prévu. Quand j'ai commencé, je n'avais que des compétences limitées en électronique et en programmation ; je pensais acquérir les connaissances nécessaires en chemin. Le but était de fabriquer la montre à partir de composants accessibles aux amateurs comme moi et trouvables sur l'internet. J'explique ici mon approche du projet et j'espère prouver que n'importe qui peut fabriquer un objet similaire à la maison.

Commencez toujours au commencement

Un beau matin, je me réveille avec l'idée de fabriquer une montre intelligente tactile avec un grand LCD et d'autres fonctions comme la lecture de MP3, carte SD, etc. Que faire d'une telle idée ? En général, mieux vaut commencer par écrire ce qu'on veut faire, les avantages et inconvénients des solutions envisagées, les fonctions de l'appareil... En résumé : un cahier des charges. Au début, je me dis : « je peux sauter cette étape et simplement dessiner un schéma et concevoir une carte ». Mais, cette première version comporte tellement d'erreurs que j'en

fais une deuxième puis une troisième. Je suis mal parti : je veux créer un appareil complexe en très peu de temps. Très mauvaise idée ! Je passe du temps à repenser le schéma et refais la carte pour réduire le nombre de bogues. Après environ deux semaines de correction des problèmes, je produis la version finale. Morale : il faut réfléchir à un projet avant de se précipiter sur le dessin du schéma ou de la carte. Ce projet m'a aussi appris à être patient.

Il est doté d'un cœur ARM Cortex-M4 cadencé à 168 MHz et possède un DSP et une FPU, 1 Mo de mémoire flash, et une myriade de périphériques. Ce µC a été choisi pour plusieurs raisons. Tout d'abord, il possède suffisamment de ports d'E/S et bus pour y relier toutes les autres puces. Une unité matérielle de gestion de la mémoire est présente ; elle est utilisée pour commander le LCD et la mémoire SRAM externe (IC10). La MMU rend leur commande beaucoup plus

Caractéristiques

- ARM Cortex-M4
- Bluetooth Low Energy (BLE)
- Gyroscope, accéléromètre et magnétomètre
- Lecture MP3
- Boîtier imprimé en 3D
- Fantastique projet didactique

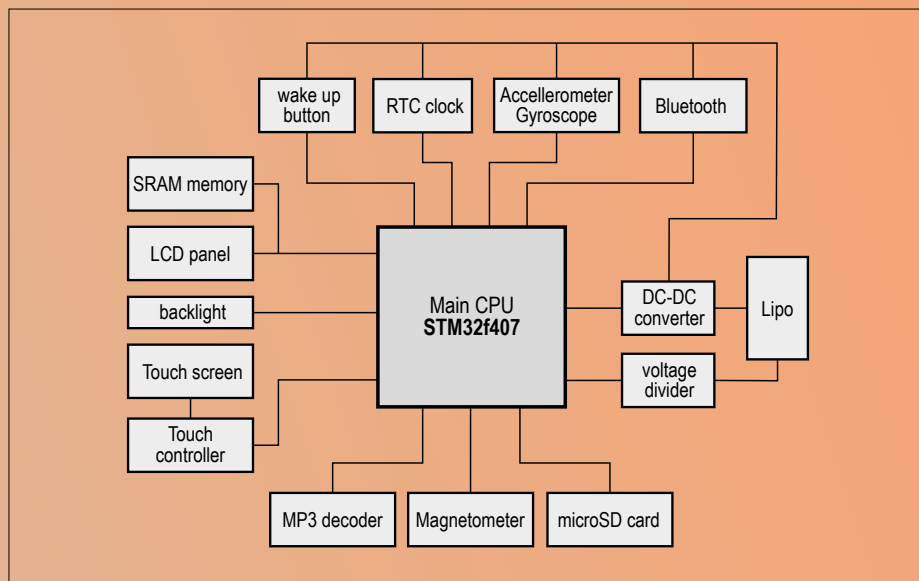


Figure 1. Aperçu de la structure de la nWatch. Pas facile de tout faire tenir au poignet !

Le matériel de la nWatch

Un diagramme de la nWatch est visible en **figure 1**, le schéma correspondant en **figure 2**. Comme vous le voyez, elle est composée de plusieurs blocs détaillés ci-après.

Le micro

L'âme de la nWatch est IC1, un µC STM32f407ZGT6 de la famille « haute performance » de STMicroelectronics.

facile qu'avec des broches d'E/S indépendantes. Le µC contient une RAM rapide et assez de flash pour mon projet. Toutefois, il y a un inconvénient majeur : son boîtier est énorme (tout est relatif). Malheureusement il n'y avait pas d'autre solution sans utiliser un boîtier BGA.

SRAM externe et carte micro-SD

Il y a deux mémoires externes à la nWatch : une puce SRAM et une carte SD.

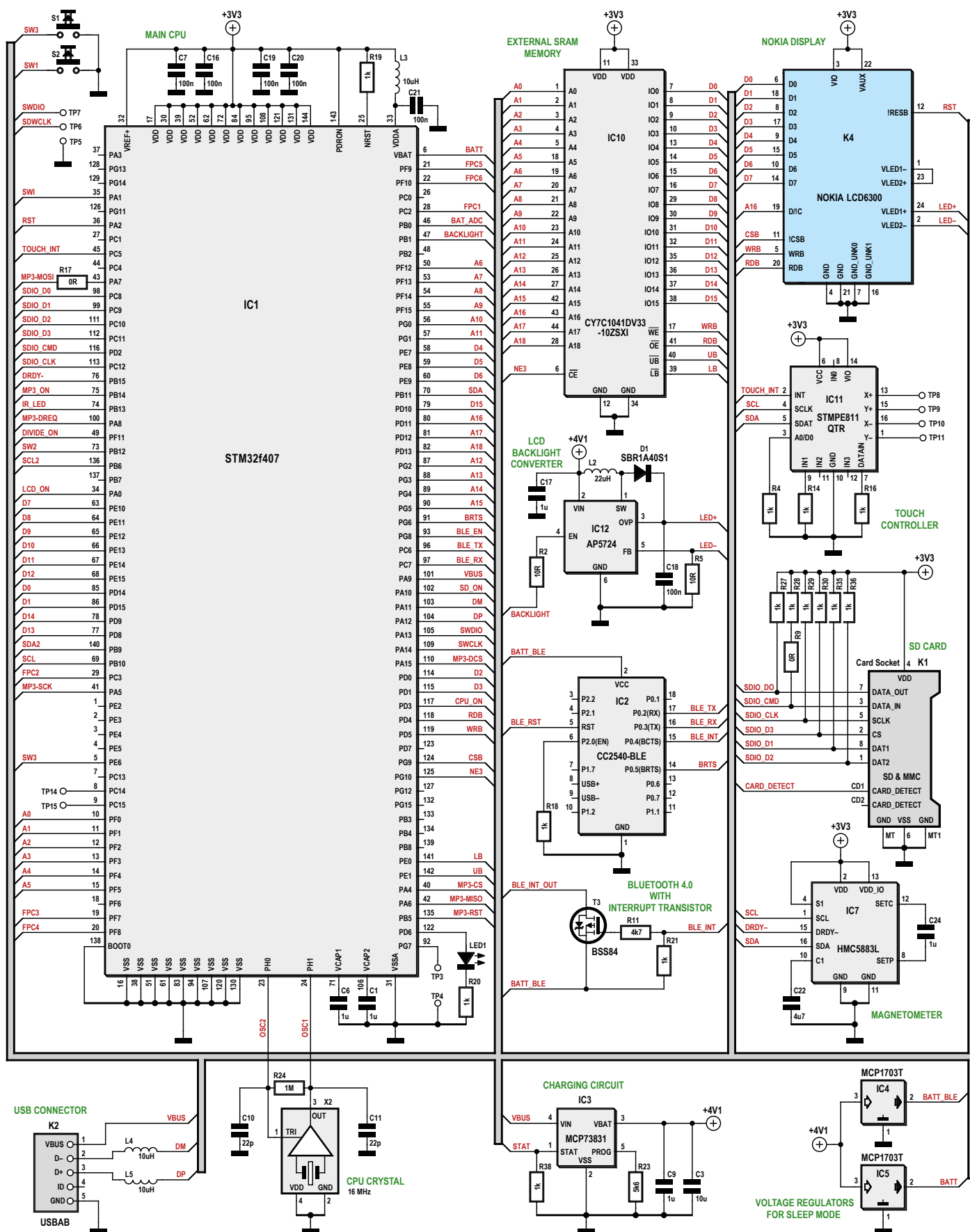
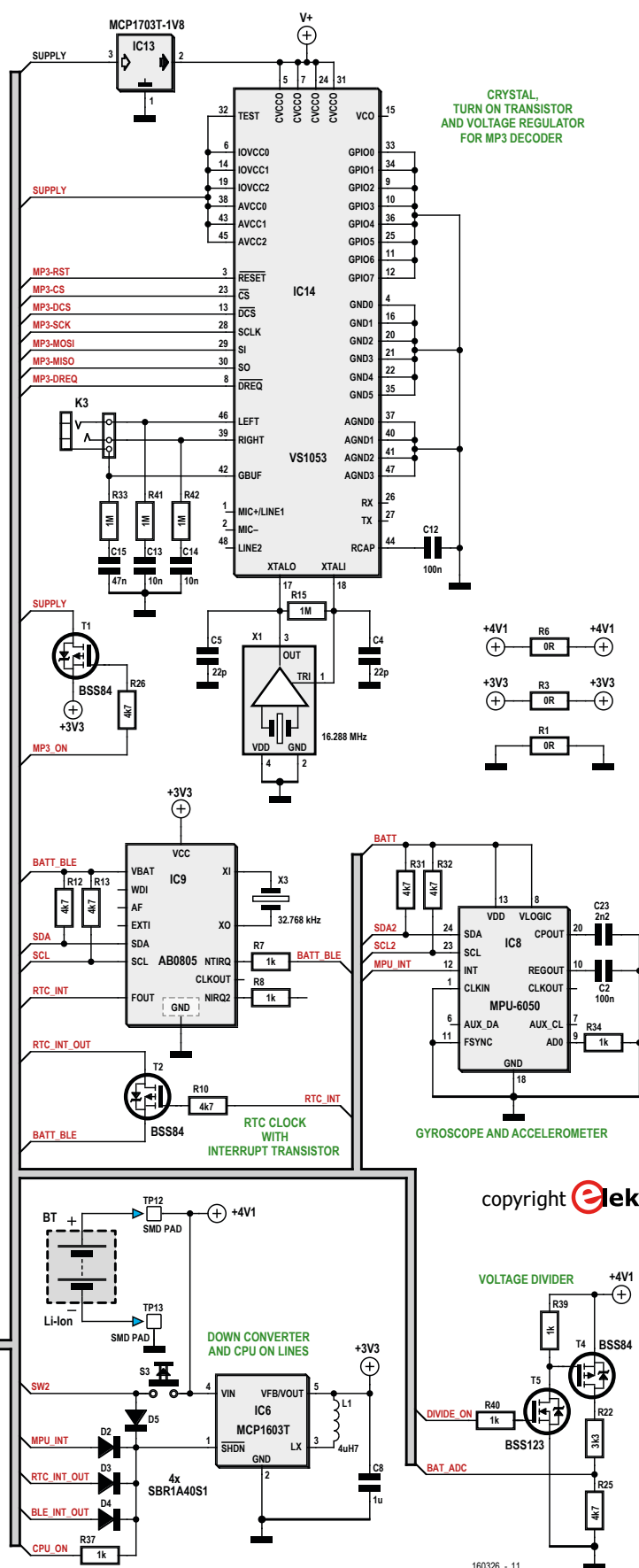


Figure 2. À part quelques résistances et condensateurs, le schéma n'ajoute guère d'information au schéma fonctionnel.



La SRAM IC10 n'était pas prévue, mais elle est nécessaire pour utiliser l'afficheur de manière confortable ; la RAM interne du processeur était trop petite pour répondre aux besoins en mémoire de la bibliothèque graphique STemWin (voir plus loin). IC10 est utilisée comme tampon d'écran pour le LCD. Toutes les modifications de l'écran sont appliquées au tampon stocké dans cette mémoire avant qu'il ne soit transféré dans le tampon de l'afficheur. Nota : cette mémoire n'est pas aussi rapide que la mémoire interne du μC ; il faut garder les piles (mémoire FreeRTOS, etc.) dans la mémoire interne pour que le processeur tourne le plus rapidement possible. La SRAM externe est reliée au même bus que le LCD et est commandée par le *flexible static memory controller* (FSMC) du μC .

La seconde mémoire externe est la carte SD non volatile K1. La carte SD est facile à retirer et à relier à un ordinateur ou un ordiphone pour y charger un fichier image ou texte ; une conséquence des débuts de la montre comme outil de triche. À l'époque, je ne voyais pas comment charger des antisèches autrement qu'en utilisant une carte SD. Cette dernière est reliée par un bus SDIO, bien plus rapide que le SPI.

L'afficheur

Le 'n' de nWatch est dû à l'afficheur, récupéré d'un vieux téléphone Nokia 6300 (**fig. 3**). Je ne l'ai pas choisi par hasard. Il est pas cher (moins de 5 \$), plutôt facile à commander via son bus parallèle à 8 bits, sur un connecteur qui n'est pas ingérable pour un amateur. De plus, il fait la bonne taille : suffisamment petit pour une montre et suffisamment grand pour la commande tactile. La rétro-ingénierie de l'afficheur n'avait pas l'air trop difficile ; j'ai trouvé plein d'articles décrivant le protocole de communication sur l'internet, et j'ai même fini par dénicher la feuille de caractéristiques du circuit de commande. Cela m'a certes per-

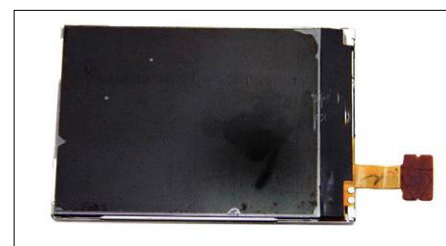


Figure 3. On peut encore trouver l'afficheur du Nokia 6300 sur l'internet.

suadé d'utiliser l'afficheur Nokia pour ma montre, mais ça ne m'a pas empêché de réaliser un prototype avec un de ces LCD TFT populaires à base d'ILI9320. À ce moment, je me dis que l'afficheur serait plus rapide si j'utilisais un bus à 16 bits au lieu d'un bus à 8 bits ; en outre la puce de commande ILI9320 intègre plus de fonctions. Cependant, cette expérience, bien qu'intéressante, ne m'a pas satisfait : le LCD possède des angles de vue pas terribles et ses couleurs sont plutôt pâles. Mon TFT n'est peut-être pas le meilleur, mais cette expérience m'a décidé à continuer avec l'afficheur Nokia. D'accord, ce n'est pas le plus rapide, mais il l'est suffisamment pour des interfaces utilisateur simples. N'oubliez pas que la manière dont l'image de l'afficheur est dessinée influence également beaucoup la vitesse. Dessiner pixel par pixel sera beaucoup plus lent que de copier de vastes zones à l'aide du *Direct Memory Access* (DMA). Le LCD est relié au connecteur K4.

Ajouter un panneau tactile

Le panneau tactile est un des objectifs principaux du projet. Toutefois, j'ai eu du mal à trouver un modèle qui remplisse mon cahier des charges et, même si j'ai réussi, avec le recul, je vois ça comme un problème majeur pour la reproductibilité de la nWatch.

Le premier panneau testé vient d'un téléphone Nokia C3 de 2010. Il convient parfaitement pour un LCD de 2,2" N82, le grand frère presque totalement compatible du Nokia 6300. Malgré sa technologie résistive, la sensibilité est plutôt bonne. Malheureusement, il est trop grand pour une montre et ne convient pas à l'afficheur 2" Nokia 6300. Après des heures de recherche sur l'internet, je trouve un panneau tactile convenable sur Aliexpress. Sa taille est parfaite, mais sa sensibilité pas vraiment impressionnante ; il faut presser assez fort. C'est le panneau actuel de la nWatch mais je n'en suis pas satisfait ; je le remplacerai par un panneau capacitif.

Les quatre fils du panneau tactile sont reliés à TP8 à TP11.

Contrôleur de panneau tactile

Pour faire l'interface avec le panneau tactile résistif, j'ai opté pour le distingué STMPE811 de STMicroelectronics (IC11). Capable de communiquer avec le μC par SPI ou I²C (utilisé ici), il possède beaucoup de registres et permet

un contrôle fin du panneau tactile. Différents paramètres sont ajustables pour adapter le panneau tactile à l'afficheur. Il possède également quelques fonctions matérielles intéressantes comme le suivi de fenêtre ou la mesure de température. Il peut déclencher une interruption lors d'un contact, informant le μC qu'il peut lire les coordonnées : pas besoin de lire les registres périodiquement.

Rétroéclairage

L'AP5724 de Diodes Inc., IC12, est un petit convertisseur-élévateur pour LED blanches en boîtier DFN à six pastilles qui commande la luminosité du LCD. Comme c'est un convertisseur-élévateur, il a besoin de quelques composants externes : une inductance de puissance (L2), une diode Schottky (D1) et des condensateurs (C17 & C18). Il est commandé par un signal MLI depuis le μC . Il est capable de piloter jusqu'à six LED, bien assez pour l'afficheur Nokia qui n'en a que deux. La résistance R5 détermine le courant maximal à travers les LED.

Lecteur MP3

Dans les deux premiers prototypes, c'est le μC qui se charge de la lecture des fichiers musicaux. Un convertisseur N/A sert à attaquer les oreillettes. Seuls les fichiers WAV sont lus et tout — lecture des fichiers, décodage des en-têtes, etc. — est fait par le logiciel. Lire des fichiers MP3 a l'air sympa, mais, comme je n'aime pas l'idée de décoder par logiciel des fichiers MP3, il faut une autre solution. C'est comme ça que je découvre le fantastique VS1053 de VLSI Solution (IC14), capable de décoder et jouer différents formats (Ogg Vorbis, MP3, AAC, WMA, FLAC et même les fichiers MIDI). Il suffit de lui envoyer les données par SPI. Cette puce fournit également des fonctions pour régler le son (volume et tonalité, etc.) et possède un amplificateur pour casque intégré. Que demander de plus ? Grâce à IC14, tout ce que le μC a à faire, c'est d'envoyer périodiquement des paquets de données à la puce décodeuse.

Le VS1053 est petit et ne nécessite que quelques condensateurs et résistances et un résonateur à quartz. Son cœur est alimenté en 1,8 V (C_{VDD}), d'où l'ajout du régulateur de tension IC13. Le transistor T1 commande la mise sous tension du décodeur pour économiser l'énergie. La **figure 4** montre l'application de lecture MP3 de la nWatch.



Figure 4. Grâce à une puce de décodage dédiée, la nWatch peut être utilisée en lecteur MP3.

Bluetooth Low Energy (BLE)

Le module Bluetooth (IC2) utilisé dans la nWatch est basé sur la puce Bluetooth LE CC2540 de Texas Instruments, et peut être utilisé pour échanger des données avec un ordiphone. Cela permet d'afficher des notifications ou de mettre à jour l'heure et la date depuis le téléphone. Ce module est un bon compromis entre prix et dimensions. La portée dépend de nombreux facteurs ; il est possible de la régler en changeant la valeur des registres gérant la puissance de transmission. La portée est plutôt bonne pour une petite antenne sur circuit imprimé. Le module peut réveiller le μC lorsqu'il reçoit des données ; celles-ci seront transmises au μC par un simple lien sériel.

Un problème avec ce module : il consomme beaucoup plus de courant lorsqu'il est connecté. Mieux vaut donc le déconnecter après chaque échange et ne le reconnecter que lorsque c'est nécessaire.

Capteurs

La nWatch est équipée de trois capteurs de position, choisis pour leur popularité dans les cercles amateurs (beaucoup de bibliothèques sur l'internet). Il s'agit d'un ensemble traditionnel : gyroscope, accéléromètre et magnétomètre.

Les deux premiers sont combinés dans IC8, un MPU6050, puce de suivi de mouvement à six axes d'InvenSense. Elle communique avec le μC via un bus I²C, et est dotée d'un régulateur de tension LDO (à faible chute) séparé pour l'alimenter même lorsque le convertisseur DC/DC principal est inactif. Nota : cette puce contient un processeur de mouvement numérique programmable (DMP). Il est donc possible de changer son micro-

logiciel et lui faire faire des calculs, ce qui allège la charge du μC principal. Le micrologiciel est chargé sur le capteur juste avant l'initialisation. Cela permet au capteur d'offrir des fonctions supplémentaires telles que la détection de tapotement ou chute libre. Des interruptions peuvent être produites lorsque cela arrive et peuvent réveiller le μC .

Le magnétomètre à trois axes (IC7) est le populaire HMC5883 de Honeywell, une petite puce en boîtier QFN qui communique également via un bus I²C, mais pas le même qu'IC8, tout simplement pour avoir des pistes courtes et éviter l'ajout de vias superflus ; ils sont difficiles à souder et prennent de la place. Comme le μC possède deux ports I²C, j'ai tout simplement relié les capteurs au bus le plus proche.

Horloge en temps réel

Supposons que vous programmiez une alarme pour vous rappeler d'un événement. Si cet événement est trop loin dans le futur et que la montre est laissée sous tension, sa batterie sera à plat avant l'événement. Remède : une horloge en temps réel à faible consommation (RTC) dotée d'une fonction alarme. Il suffit de programmer l'alarme de la RTC avant de mettre la montre hors tension. Quand la RTC atteint la date et l'heure de l'événement, elle commande la mise sous tension du régulateur IC6 (via le transistor T2) et la montre est allumée.

Une bonne RTC est l'AB0805 d'Abracom (IC9). Elle possède une interface I²C et ne consomme qu'environ 50 nA, bien moins que ce qu'il est possible d'atteindre avec le μC en sommeil profond.

Alimentation

La nWatch se porte au poignet et doit donc consommer le moins possible. Permettre à l'alimentation en 3,3 V d'être coupée est donc crucial. Pour ce faire, j'ai utilisé un régulateur de tension (IC6) doté d'une broche d'activation (SHDN) qui commande sa sortie. La montre peut être (re)mise sous tension de quatre façons. Le moyen le plus évident et le plus facile à mettre en œuvre est l'action sur le poussoir S3. Lorsque le poussoir est actionné, une tension positive est appliquée à la broche SHDN via D5 et le régulateur démarre, alimentant le reste du circuit. D2, D3 et D4 remplissent la même fonction que D5, mais pour l'ensemble gyroscope et accéléromètre (IC8), la RTC (IC9) et le module BLE (IC2) respectivement.

IC6, un MCP1603 de Microchip, est un régulateur type *buck* synchrone à haute efficacité 500 mA, 2,0 MHz, destiné aux applications alimentées par batterie, parfait pour notre projet. Il se contente d'une inductance (L1) et d'un condensateur (C8) pour fonctionner.

IC4 et IC5, deux régulateurs de tension LDO alimentent les composants toujours sous tension (tant que la batterie dure) : IC2, IC8 et IC9. IC5 alimente aussi le μC et empêche la perte du contenu des registres secourus par la batterie, lorsque le régulateur 3,3 V est éteint. Ces registres se trouvent dans une zone protégée spéciale du μC et leur contenu est préservé aussi longtemps que la tension sur la broche VBAT est suffisamment élevée.

IC4 alimente le module BLE afin qu'il n'interfère pas trop avec le reste du circuit.

Comme déjà dit, le régulateur de tension LDO IC13 fournit le 1,8 V nécessaire au cœur du décodeur MP3 (IC14).

R22 et R25 forment un diviseur de tension permettant au μC de mesurer la tension de batterie. T4 et T5 permettent de déconnecter ce diviseur pour économiser l'énergie.

Batterie LiPo avec chargeur

La nWatch est alimentée par une batterie LiPo 150 mAh. Sa capacité, qui paraît faible, permet toutefois de maintenir la montre en veille pendant plus d'une semaine. Si vous ne regardez l'heure que de temps en temps, une charge de batterie durera au moins cinq jours. La batterie est équipée d'un circuit de protection qui la protège des courts-circuits et décharges profondes. Lorsqu'elle est complètement déchargée, ce circuit la déconnecte du reste du circuit jusqu'à ce qu'un chargeur lui soit relié. Lorsqu'une telle déconnexion se produit, toute la mémoire est effacée, même la date et l'heure de la RTC. Pour que la restauration des données soit facile, j'ai écrit une petite appli pour mon ordiphone (voir plus loin).

La charge de la batterie est contrôlée par une puce de charge pour une cellule, le MCP73831 (IC3) de Microchip, qui gère automatiquement le processus de charge. Il ne lui faut qu'une résistance pour régler le courant de charge.

Connecteurs

La nWatch possède trois connecteurs principaux : une fente pour carte

micro-SD (K1) sur le fond du boîtier, un connecteur micro-USB (K2) et une barrette à 3 voies (TP5, TP6 & TP7) utilisée pour programmer le μC et/ou déboguer l'exécution avec un programmeur ST-Link. C'est la manière la plus facile et la plus rapide de programmer la montre. Le connecteur micro-USB est utilisé pour charger la batterie LiPo et permet d'échanger des données par USB. Lorsque l'on relie la nWatch à un ordinateur (Windows, Linux ou Mac OS), un périphérique de stockage de masse est détecté, ce qui donne accès au contenu de la carte SD. Le port USB permet également de charger des programmes sur la montre grâce au chargeur de démarrage intégré. Cette méthode est toutefois bien plus lente que le programmeur externe. Pour programmer l'exécutable (HEX ou BIN) par USB, il faut d'abord convertir le fichier au format spécial DFU (*device firmware update*). Il n'est pas possible de déboguer via le bus USB ; considérez l'USB comme une fonction supplémentaire plutôt que le principal port pour le développement.

Le circuit imprimé

Le contour du circuit imprimé est adapté au boîtier imprimé et au connecteur jack de 3,5 mm. Pour que la montre soit la plus fine possible, j'ai monté le connecteur « dans » la carte plutôt que sur celle-ci. Cela permet de réduire l'épaisseur de l'ensemble.

Même si la carte comporte des pistes étroites (≥ 6 mil), vous pourrez quand même la fabriquer vous-même avec le procédé photosensible. Mais, fabriquer la carte à la maison n'attirera sans doute que les purs et durs : il y a 200 vias à souder et ce n'est pas une mince affaire. Le μC dans son gros boîtier LQFP à 144 broches prend beaucoup de place sur la carte qui est du coup très dense ; je ne pense pas que j'aurais pu la faire plus petite. J'ai bien pensé à un moment à passer sur une carte à 4 ou 6 couches et utiliser des puces en boîtier BGA, mais je n'ai pas poussé l'idée plus loin : je voulais que la nWatch soit une carte de développement à faible coût pour amateurs, pas seulement un prototype onéreux. Des composants plus petits auraient rendu l'assemblage de la carte significativement plus difficile et l'utilisation de boîtiers BGA aurait requis des outils spécialisés. Je me suis donc contenté d'une carte double face.

Le placement des composants sur la carte est crucial lorsque l'on conçoit



Figure 5. Pour réduire l'épaisseur de l'ensemble, la batterie LiPo est placée au-dessus de la puce SRAM.

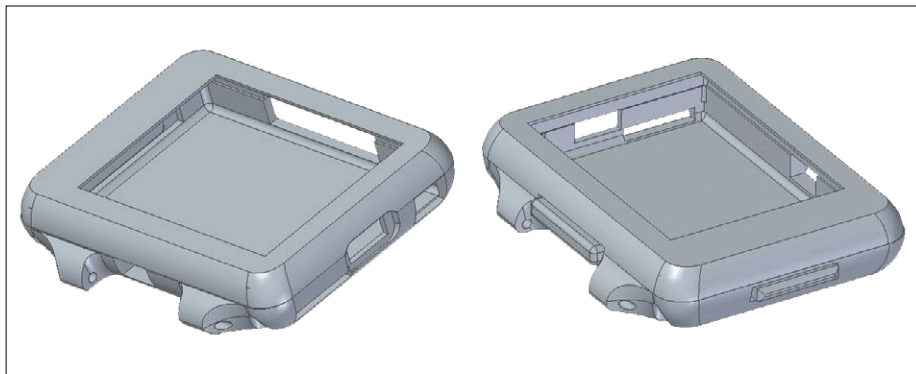


Figure 6. Rendu en 3D du boîtier de la montre.

une montre : chaque millimètre compte. Essayez de placer les composants avec des hauteurs semblables du même côté de la carte : le composant le plus haut de chaque côté déterminera la hauteur totale de la carte. La batterie LiPo est aussi un composant à problèmes. Le mieux, si vous pouvez, c'est de la placer dans une encoche fraisée dans la carte. Lorsque ce n'est pas possible, essayez de ne placer que des composants très plats (résistances, etc.) sur une zone de la carte. Dans mon cas, je n'ai pas pu éviter de placer la mémoire SRAM sous la batterie (**fig. 5**). Elle ne fait que 1 mm d'épaisseur et c'était le meilleur compromis. D'autres composants, comme le connecteur jack de 3,5 mm ou l'embase USB, peuvent être montés dans des encoches faites dans la carte. Cela réduit aussi l'épaisseur totale.

Composants imprimés en 3D

L'impression 3D est facile ; ce qui est difficile c'est de créer le modèle. Non seulement l'esthétique doit être agréable à l'œil, surtout lorsqu'il s'agit d'une montre, mais le circuit imprimé et tous ses composants doivent aussi tenir dans le boîtier. Malheureusement, il n'y a pas

de solution miracle : mesurez avec précision toutes les dimensions, réalisez un premier prototype qui ne conviendra pas, ajustez, et répétez ce processus trois ou quatre fois.

Les montres ne sont pas vraiment des objets faciles à concevoir ; elles sont petites et fines. Commencez par dessiner une forme approximative, tout en gardant à l'esprit les dimensions de la carte, de la batterie LiPo et du LCD avec panneau tactile. Ensuite, mesurez, essayez, jurez sans doute un peu et itérez jusqu'à ce que tout tienne.

Les composants de la nWatch sont conçus avec Solid Edge (**fig. 6**). Je voulais une montre étanche aux projections d'eau, mais c'est trop difficile : les connecteurs posent problème, surtout la fente pour carte SD. J'ai créé un boîtier en deux parties, fixées par quatre vis. C'est une bonne solution, car une fois les vis retirées, l'accès à la carte principale, la batterie LiPo et le LCD avec panneau tactile est facile. Le boîtier de la montre possède deux fentes rectangulaires pour les boutons. Le bouton de réveil est le plus gros, afin qu'il soit facile à actionner ; il se trouve au-dessus de la jonction avec le bracelet. Les boutons ont également été imprimés.

L'apparence du boîtier est meilleure lorsque celui-ci est imprimé avec une faible épaisseur de couche, le pas vertical sera aussi petit que possible. Avec un pas de 0,2 mm sur l'axe Z, le résultat est satisfaisant. Mon imprimante 3D Prusa Mendel Iteration 2 (I2) 3D possède une résolution et une stabilité suffisantes pour créer un boîtier sympa, donc la plupart des imprimantes 3D devraient convenir ; les composants sont plutôt faciles à imprimer.

Construisez votre nWatch

1. Rassemblez tous les composants

Commencez par rassembler tous les composants, surtout ceux qui sont difficiles à trouver. Vous saurez vite à quoi vous en tenir. Exception faite du panneau tactile, sans doute le plus difficile à trouver, il y a quelques composants qui poseront problème, le connecteur jack de 3,5 mm avec broches au milieu est un bon exemple. Le mien vient d'une vieille tablette dont la référence m'échappe ; je n'ai pas trouvé de modèle semblable sur l'internet. Ceux que j'ai trouvés ont soit une empreinte un peu différente soit un boîtier plastique incompatible ou bien les deux. Vous pouvez ajuster l'encoche

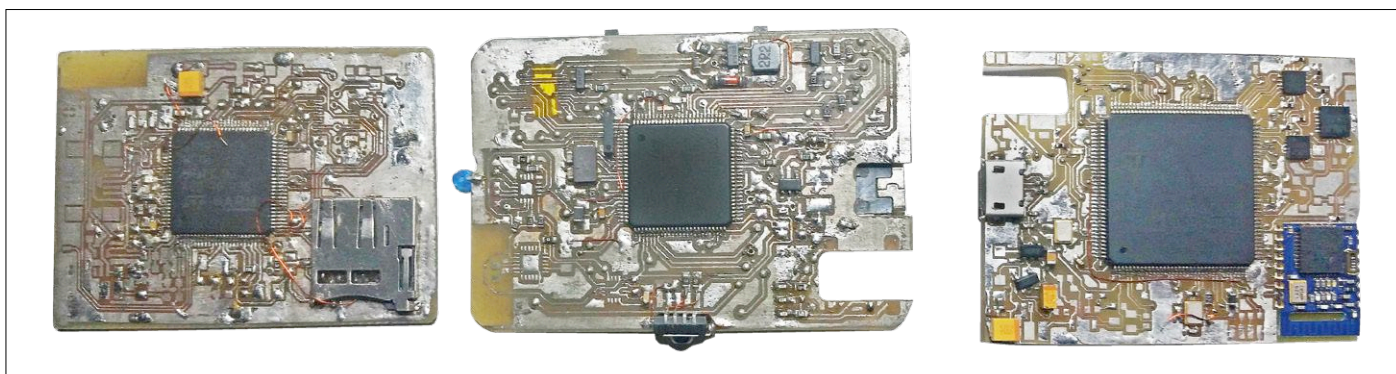


Figure 7. Plusieurs itérations ont été nécessaires avant d'arriver à la version finale de la nWatch.

dans le circuit imprimé avec une petite lime ou bien relier le connecteur aux pastilles avec des fils.

Les poussoirs sont moins problématiques, mais pourraient être difficiles à trouver. Je n'ai pas leur référence exacte, juste leurs dimensions. La meilleure méthode est sans doute de comparer les dimensions et l'apparence.

2. Assemblez le circuit imprimé

Une fois tous les composants rassemblés et le circuit imprimé sur votre bureau, passez à l'assemblage de la carte (**fig. 7**). La soudure de la plupart des composants ne sera pas trop difficile pour peu que vous utilisiez un petit fer. Les plus petits composants sont les résistances et condensateurs en boîtier 0402 ; il faut un peu de pratique. Une station à air chaud est plus ou moins nécessaire pour monter les quartz CMS et les puces en boîtier QFN (sans ça vous ne pourrez pas atteindre les pastilles sous le boîtier). Aucune raison d'avoir peur d'un premier contact avec les QFN toutefois. Si vous préparez les choses correctement, vous n'aurez qu'à positionner la puce approximativement et appliquez l'air chaud ; elle s'alignera d'elle-même. Faites attention à ne pas chauffer trop longtemps : le composant pourrait cramer. Tout le reste n'est que classique soudure de CMS. Vérifiez que toutes les broches des puces les plus grosses (μ C, VS1053, etc.) sont soudées correctement. Parfois, elles ont l'air soudées alors qu'elles se trouvent juste un peu au-dessus de la surface de la carte, mais sans toucher la pastille.

3. Assemblage final

Une fois la carte assemblée, vérifiez qu'elle ne comporte pas de courts-circuits. Lorsque vous êtes sûr de vos soudures, reliez la batterie et collez-la à la carte au-dessus d'IC10, la puce SRAM.

Mon panneau tactile est livré avec du ruban adhésif sur ses bords ; je retire le film de protection et je le colle sur le LCD. Attention à son raccordement à la carte. Les panneaux n'ont pas tous le même brochage ; il faut jouer aux devinettes pour trouver l'ordre des quatre broches. Dans mon cas, j'ai dû croiser deux fils. Par chance, vous ne ferez pas de dégâts si vous ne le raccordez pas correctement ; n'hésitez pas à tester autant de combinaisons que nécessaire. Si les coordonnées sont inversées ou que les actions ne sont pas détectées, permutez simplement les fils jusqu'à ce que tout fonctionne (**fig. 8**).

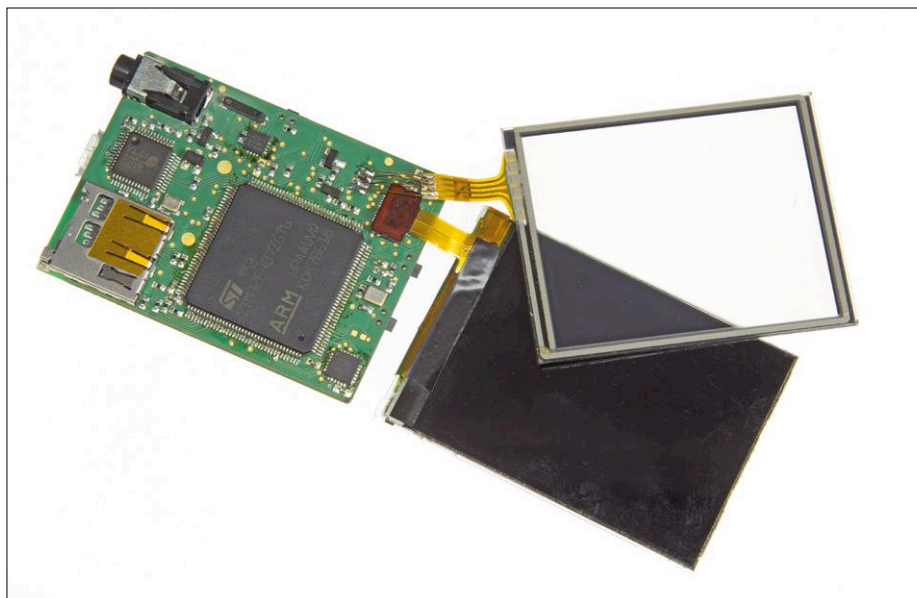


Figure 8. Prototype assemblé avec l'afficheur et le panneau tactile en place.

Après le raccordement à la carte principale, mettez le connecteur LCD en place et placez l'afficheur de l'autre côté de la carte, par-dessus la batterie.

L'assemblage du boîtier est vraiment simple (**fig. 9**). Préparez les surfaces où les deux moitiés se rencontrent avec du papier de verre. Vous n'obtiendrez pas un bon ajustement sans ça. Pensez aussi à nettoyer les ouvertures au fond du boîtier et les fixations pour le bracelet : elles seront sans doute obstruées par du filament sur la pièce brute d'impression. Insérez ensuite le bouton latéral suivi de la carte (n'oubliez pas de d'abord retirer la carte micro-SD). Placez

le bouton de réveil dans sa fente et couvrez l'ensemble avec la partie supérieure du boîtier. Insérez et serrez les quatre vis et voilà !

N'importe quel bracelet de taille standard convient, mais n'oubliez pas qu'il faut aussi deux barres de fixation à ressorts.

Le logiciel

Le logiciel est une partie essentielle de toute carte de développement ; c'est vrai aussi pour une montre intelligente. Tout en développant différentes versions du matériel, j'ai aussi écrit du code pour commander la montre et adapté beaucoup de bibliothèques pour qu'elles fonctionnent avec la montre.

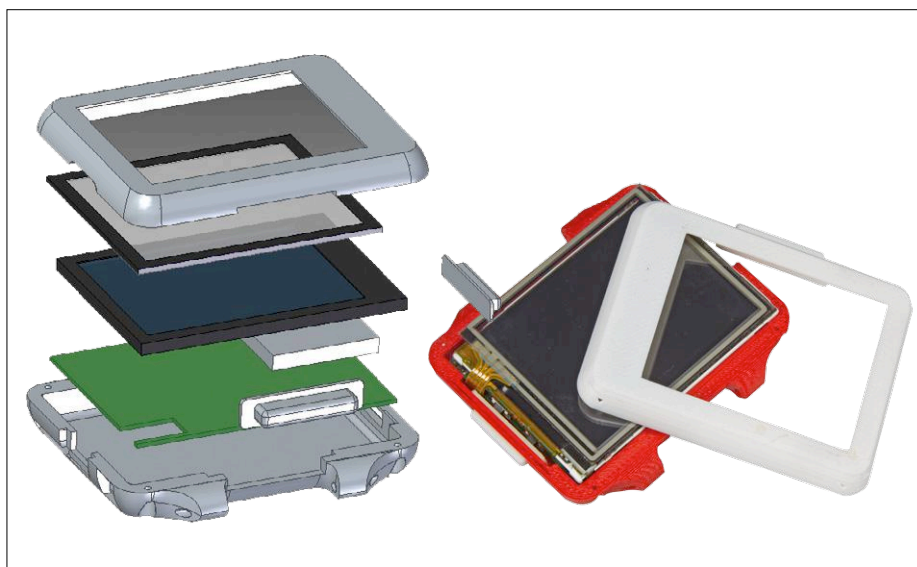


Figure 9. Pour l'assemblage final, empilez les composants comme ceci.



LISTE DES COMPOSANTS

Résistances

R1,R3,R6,R9 = 0 Ω , 0402
 R2,R5 = 10 Ω , 0603
 R4,R7,R8,R14,R16-R21,R27-
 R30,R35,R36,R42,R46-R48,R55 = 1 k Ω ,
 0402
 R10-R13,R31,R32,R63,R66 = 4,7 k Ω , 0402
 R15,R49-R51,R85 = 1 M Ω , 0402
 R54 = 5,6 k Ω , 0402
 R67 = 3,3 k Ω , 0402

Condensateurs

C1, C6 = 1 μ F, 0402
 C2,C7,C16,C18-C21,C39 = 100 nF, 0402
 C3 = 10 μ F, 0805
 C8,C9,C17,C25 = 1 μ F, 0805
 C10,C11,C40,C41 = 22 pF, 0402
 C26 = 4,7 μ F, 0603
 C29 = 2,2 nF, 0603
 C35 = 47 nF, 0402
 C36,C38 = 10 nF, 0402

Inductances

L1 = 4,7 μ H, Murata LQM31PN4R7M00L, 1206

L2 = 22 μ H, Murata LQH31CN220K03L, 1206
 L3-L5 = 10 μ H, 0402

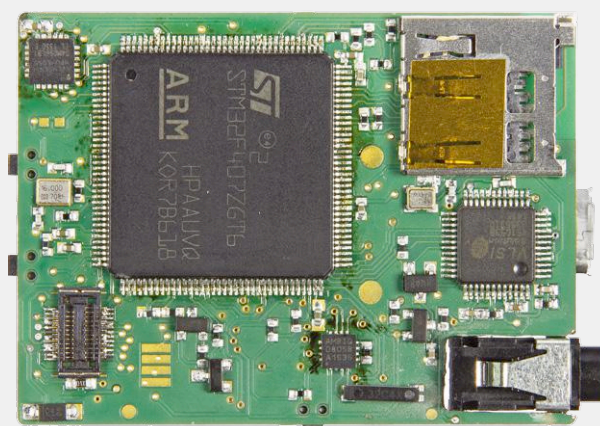
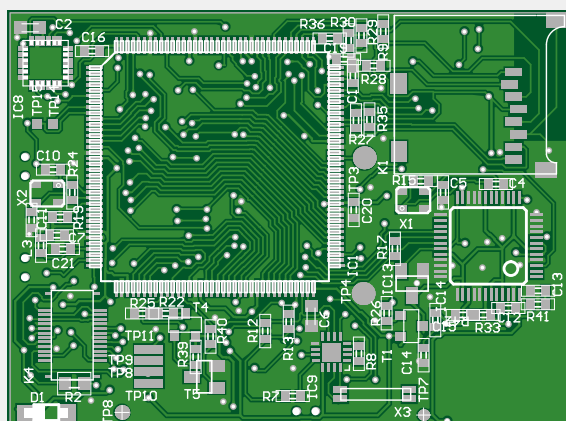
Semi-conducteurs

D1-D5 = SBR1A40S1
 IC1 = STM32F407ZGT6
 IC2 = RF-BM-S02
 IC3 = MCP73831
 IC4, IC5, IC15 = MCP1703T-3302E/CB
 IC6 = MCP1603T-3301/OS
 IC7 = HMC5883L
 IC8 = MPU-6050
 IC9 = AB0805
 IC10 = CY7C1041DV33-10ZSX1
 IC11 = STMPE811
 IC12 = AP5724
 IC14 = VS1053
 LED1 = LED, 0603
 T1-T4 = BSS84
 T5 = BSS123

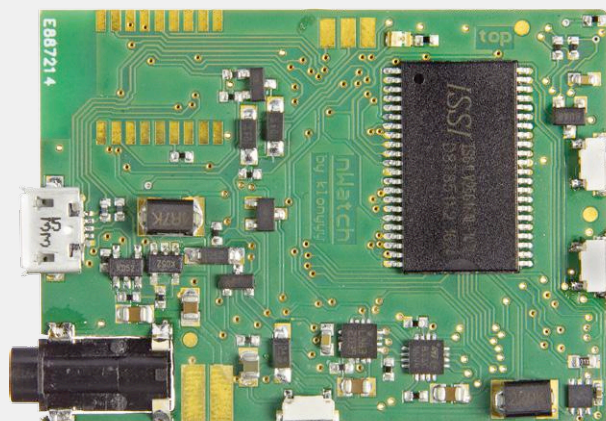
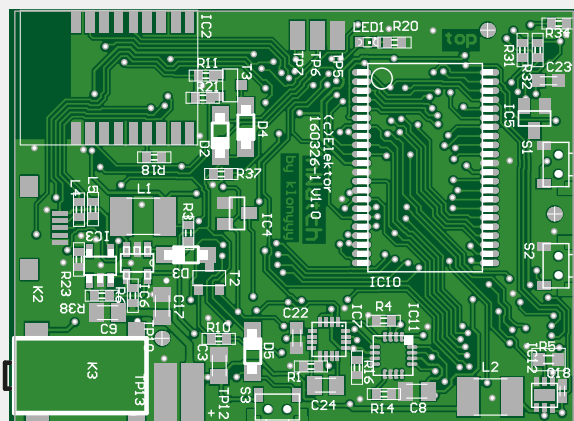
Divers

K1 = connecteur pour carte micro-SD
 MEM2061-01-188-00-A.

K2 = connecteur micro-USB, type B
 K3 = connecteur jack stéréo de 3,5 mm
 K4 = afficheurs à cristaux liquides Nokia 6300
 (ou équivalent) + connecteur
 S1-S3 = interrupteur tactile de bord de carte,
 3,45×2,2×4 mm, TACTDR345H40B160
 X1 = oscillateur à quartz 12,288 MHz,
 SG-210STF
 X2 = résonateur à quartz 16 MHz, 3,2×2,5 mm
 X5 = résonateur à quartz 32,768 kHz, MC-146
 Panneau tactile résistif de 5,08 cm (2")
 Boîtier imprimé en 3D
 Bracelet de montre et deux barres de fixation
 à ressorts
 Batterie LiPo de 150 mAh
 Circuit imprimé réf. 160326-1



150% de la taille réelle



Interface utilisateur avec STemWin

Une des bibliothèques les plus importantes pour la nWatch est [STemWin](#). Il s'agit de la bibliothèque graphique qui commande le LCD et dessine tous les éléments de l'interface utilisateur. Basée sur l'emWin de Segger, elle est supportée par STM (le fabricant du µC) et il est donc facile de l'adapter pour n'importe quel µC à base de STM32. Vous trouverez beaucoup d'exemples différents ainsi qu'une documentation complète sur l'internet. Les éléments tels que les boutons et zones de saisie de texte visibles sur les programmes d'exemples vous rappelleront l'ancien look Windows. Mais, et bien heureusement, toute cette partie de l'interface est personnalisable. La bibliothèque est fournie avec plusieurs utilitaires qui vous aideront à créer une interface à votre goût. [GUIBuilder](#) est, à mon avis, le plus utile de tous les outils. Il permet de concevoir un écran et produire automatiquement le code à ajouter à votre projet. Sans aucun doute, cet outil intuitif vous fera économiser du temps.

Multitâche avec FreeRTOS

Ce système d'exploitation en temps réel (RTOS) libre, bien connu des amateurs et professionnels, commande la montre. Grâce à sa prise en charge du multitâche, plusieurs applications pourront tourner simultanément. Il permet d'attribuer des priorités aux tâches, de suspendre leur exécution, de leur attribuer une durée d'exécution et bien plus. Je ne vais pas tout détailler ici ; il est très bien documenté sur l'internet.

Jouer des animations avec libjpeg

Une autre bibliothèque classique que j'ai utilisée est [libjpeg](#), pour décoder les fichiers JPEG. Elle sert à jouer des vidéos simples sur la montre. Son avantage principal est la vitesse : jusqu'à 21 images par seconde avec des images de 240×320 pixels. Mais, cela a un prix : développée à l'origine pour promouvoir le standard JPEG, elle utilise beaucoup de mémoire. Si vous avez besoin d'un décodage vraiment rapide et que vous n'avez pas de contraintes mémoire, c'est la bibliothèque parfaite. Si la mémoire est comptée, mieux vaut utiliser [TJpegDec](#).

Bibliothèque graphique à faible empreinte mémoire

Si [libjpeg](#) ne convient pas, utilisez la bibliothèque *Tiny JPEG Decompressor* ou [TJpegDec](#). Cette bibliothèque minuscule, optimisée pour les systèmes embarqués, a été créée

par ChaN, un développeur japonais bien connu pour ses bibliothèques populaires *FatFS* et *petit FatFS* si utiles avec les cartes SD dans les systèmes embarqués. Comme [TJpegDec](#) a été conçue pour les petits systèmes à µC, elle consomme moins de mémoire que [libjpeg](#). Elle est suffisamment rapide pour afficher une galerie photo, mais vous ne pourrez pas vraiment jouer de la vidéo.

Bibliothèques

de gestion du matériel ST

Les bibliothèques bas niveau STM, bien que très complètes, ont mauvaise réputation. Certains utilisateurs pensent qu'elles sont lentes et que la programmation directe du matériel est une meilleure approche, d'autres n'utilisent que la bibliothèque pour périphériques standards. Il n'y a jamais eu de consensus. Personnellement, je n'ai pas de problèmes avec ces bibliothèques, même s'il y a des fois où je dois configurer un registre matériel moi-même, par ex. dans le cas d'une fonction appelée périodiquement et qui doit s'exécuter rapidement (comme un transfert DMA). Si vous utilisez la dernière version de la bibliothèque d'abstraction matérielle de STM (HAL), il faut adapter mon code pour y inclure les fichiers d'en-tête nécessaires, etc. Certains ont commencé à utiliser le générateur de code d'initialisation [STM32CubeMX](#) pour créer des fonctions d'abstrac-



tion ; je n'ai aucune expérience avec ça et je vous laisserai donc vous débrouiller.

Environnement

de développement intégré

Au départ, le logiciel de la nWatch a été écrit dans l'environnement de développement intégré (EDI) [CoIDE](#) de [CooCox](#). J'ai choisi cet EDI parce que je ne voulais pas de limite sur la taille du code et une interface ressemblant à Eclipse. Je n'ai pas eu à me plaindre de cet EDI, mais il semble avoir été un peu oublié par ses créateurs. La dernière version publiée comporte des bogues et j'ai dû me contenter de celle d'avant. J'ai fini par utiliser [TrueSTUDIO](#) d'Atollic qui avait l'air mieux maintenu. Vous pouvez choisir un autre environnement, même un Eclipse de base avec chaîne de compilation ; toutefois certains environnements limitent la taille du code ou bien ne sont disponibles qu'en version d'essai. Configurer vous-même et complètement Eclipse pour la nWatch vous donnera la migraine et vous découragera.

À propos de l'auteur

Pierre (Piotrek dans sa langue) est un lycéen de 19 ans qui habite en Pologne. Il est amateur de systèmes embarqués et aime programmer des microcontrôleurs, particulièrement les AVR et STM32. Il a démarré sa carrière en électronique en construisant des robots sumo pour des compétitions et s'est récemment intéressé aux *wearable*, ces appareils électroniques que l'on porte dans les vêtements ou comme des bijoux. Lorsqu'il en a le temps, il aime aussi construire des avions radiocommandés.

Fast Forward Award 2016

La nWatch était l'une des concurrentes pour l'*Elektor Fast Forward Award* organisé par Elektor en collaboration avec les directeurs du salon [electronica 2016](#) à Munich. C'était une formidable occasion pour amateurs et professionnels de tous bords de partager et présenter projets, produits et jeunes pousses. L'auteur de cet article a présenté le projet nWatch au public et à un jury international de professionnels. Cela lui a permis de rencontrer plusieurs fabricants pour discuter des composants utilisés dans la montre. Beaucoup d'opinions et de suggestions d'autres participants et visiteurs ont permis à Pierre de partir heureux et la tête pleine de nouvelles idées. Gagner la troisième place dans la catégorie « prototypage » du FFA n'a fait qu'élargir son sourire.

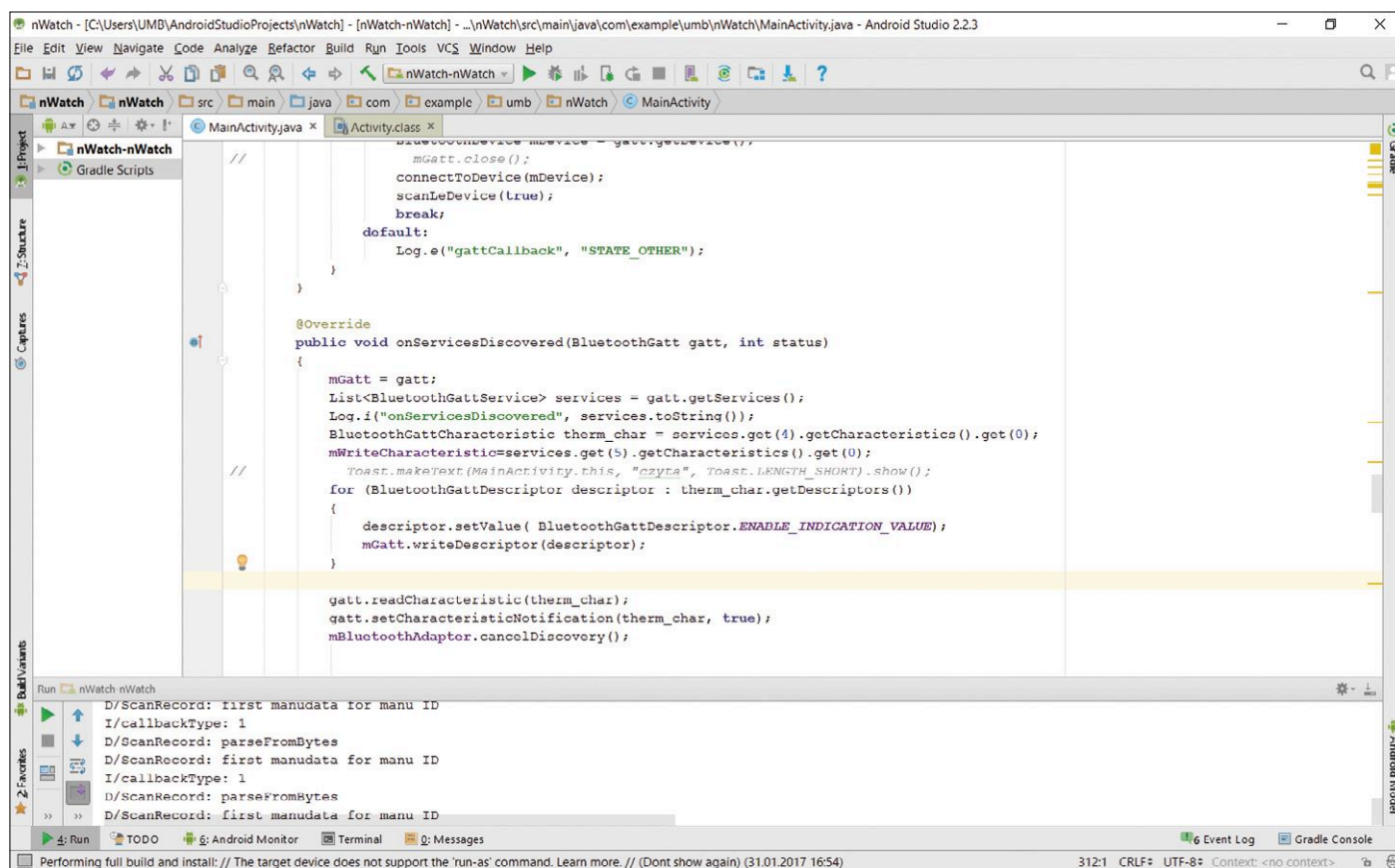


Figure 10. Android Studio 2.2.3 a été utilisé pour créer une appli Android simple pour communiquer avec la montre.

Programmation en circuit

Je vous recommande chaudement d'investir dans un programmeur ST-Link bon marché. Non seulement, il est plus confortable à utiliser que le chargeur de démarrage USB, mais il permet également de déboguer le code. Faire partie des cartes STM Discovery et Nucleo (souvent distribuées gratuitement dans les salons) en fait le meilleur candidat. Tout ce que vous aurez à faire c'est de fabriquer un câble avec un connecteur femelle 2 mm à trois voies.

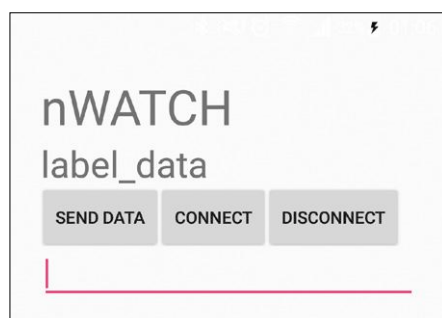


Figure 11. L'appli Android n'a pas (encore) une belle interface, mais elle permet de régler la date et l'heure de la nWatch en tapotant le bouton *Send Data*.

Appli Android

Même si je ne suis pas un programmeur Java, j'ai décidé d'écrire une appli simple pour envoyer des données à la montre depuis un ordiphone. J'ai trouvé un exemple d'appli utilisant le Bluetooth Low Energy et l'ai modifié pour lire l'heure et la date du téléphone et les envoyer sans fil à la montre (**fig. 10** & **fig. 11**). C'est très confortable, car il n'y a pas besoin d'écrire une appli spéciale pour la montre. Lorsque vous démarrez l'appli sur le téléphone, elle recherche les nWatch à portée et, quand une est détectée, il suffit d'utiliser le bouton *send* pour régler la date et l'heure. La prochaine version de l'appli mettra à jour la montre sans intervention de l'utilisateur.

Conclusion

Je crois que j'ai atteint mes objectifs initiaux. Le système que j'ai construit m'a beaucoup appris et m'a permis de lier programmation, électronique, mécanique, conception et impression 3D. Les composants sont bon marché, certains ont même été récupérés sur de vieux téléphones (ce qui malheureusement rend la montre difficile à reproduire). J'ai appris énormément en résolvant

tous les problèmes rencontrés lors de la conception de la montre. J'ai maintenant une bonne idée du temps et des efforts nécessaires pour un projet de cette taille. J'ai aussi appris à apprécier le travail d'équipe : je n'aurais pas été capable de créer seul chacun des composants. Je pense que le résultat vaut le coup. Je voulais avec cet article vous montrer que n'importe qui peut construire une montre de ce type. Ce n'est pas si difficile après tout et, à mon avis, c'est le meilleur moyen d'apprendre quelque chose de nouveau. J'espère que vous avez apprécié cet article autant que j'ai pris plaisir à fabriquer la nWatch. ◀

(160326 – version française : Kévin Petit)

Lien

[1] www.elektormagazine.fr/160326





vol tous azimuts

piloté par **Clemens Valens** (labo d'Elektor)

Arduino Create

Bien que pas encore poussièreuse, la célèbre plateforme Arduino évolue de façon assez spectaculaire. Ces derniers mois, l'équipe d'Arduino a mené discrètement, en coulisse, un projet de plateforme de développement en ligne reposant entièrement sur le nuage : l'*Arduino Create*. Un projet ambitieux qui étend le concept du défunt Codebender jusqu'au niveau de la BBC micro:bit. Les habitués du site Arduino.cc ont eu accès à cette plateforme presque depuis l'origine, mais son existence n'a pas fait l'objet de beaucoup de publicité. À ce stade Arduino Create a deux objectifs principaux. Le premier est de complètement supprimer l'installation fastidieuse des cartes, outils et bibliothèques. L'utilisateur d'Ar-

duino Create se connecte avec son compte, branche une carte compatible sur l'ordinateur et commence à coder — ça doit être aussi simple que ça. La carte est reconnue par l'EDI en ligne, toutes les bibliothèques publiées dans le référentiel Arduino sont disponibles sans installation, l'EDI est toujours à jour, etc. Tout à fait comme mbed d'ARM (sans le jargon du programmeur) ou l'environnement de la BBC micro:bit (qui ne connaît qu'une carte). La différence avec ces plateformes est qu'Arduino Create peut, comme le pouvait

Codebender, programmer la carte lui-même sans que l'utilisateur ait besoin de copier à la main un fichier sur le disque. Pour que ça marche, il faut d'abord installer un logiciel qui permet au navigateur d'accéder à la carte ; ce n'est donc pas complètement transparent, mais une fois fait, le reste est automatique.

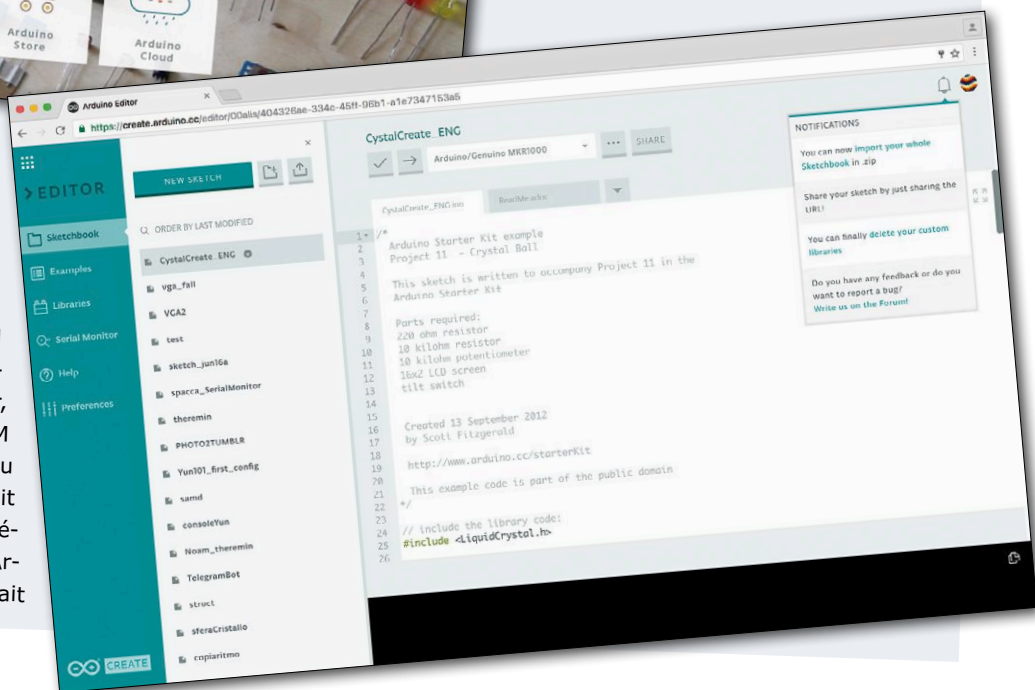
Le second objectif est de rendre le développement d'applications IdO aussi simple que possible par la fourniture gratuite de services avec des exigences peu élevées (*low-threshold*) pour le nuage. Là encore tout comme mbed, mais accessible aux non-spécialistes.

Les partisans du traditionnel Arduino peuvent continuer à utiliser l'EDI hors ligne qui leur est si familier. Toutefois, lors du téléchargement d'une nouvelle version sur le site web, ils noteront qu'Arduino Create est encouragé comme la voie à suivre. Il nous tarde de voir ce que le futur nous réserve. Peut-être un Arduino OS ?

(160404-a)

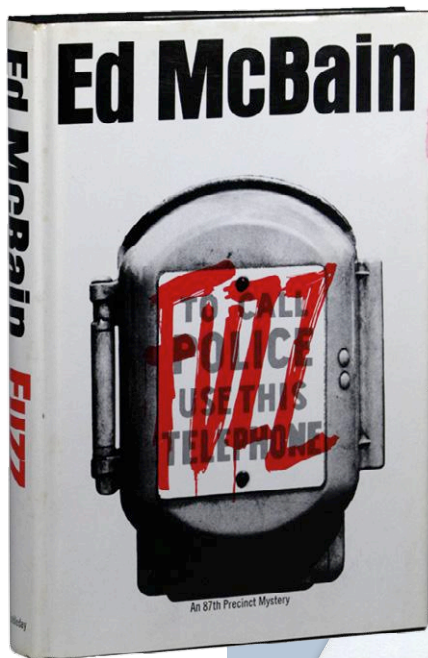


duino Create se connecte avec son compte, branche une carte compatible sur l'ordinateur et commence à coder — ça doit être aussi simple que ça. La carte est reconnue par l'EDI en ligne, toutes les bibliothèques publiées dans le référentiel Arduino sont disponibles sans installation, l'EDI est toujours à jour, etc. Tout à fait comme mbed d'ARM (sans le jargon du programmeur) ou l'environnement de la BBC micro:bit (qui ne connaît qu'une carte). La différence avec ces plateformes est qu'Arduino Create peut, comme le pouvait



8 400 000 000

8,4 milliards d'objets connectés seront utilisés en 2017.



Électronique et littérature

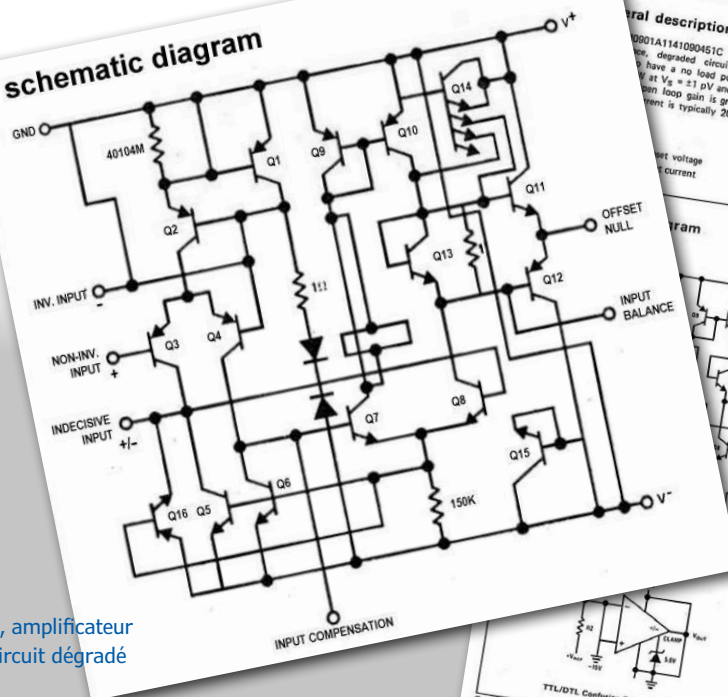
Pour autant que je sache — mais que sais-je ? — littérature et électronique ne font pas bon ménage. Bien sûr il y a « Le scandale de la substitution de Darlington » en 1886 avec Sherlock Holmes, mais il a été écrit bien avant que quelqu'un n'ait l'idée du transistor Darlington (au passage découvert en 1953 et attribué à Sydney Darlington), et il est donc aussi bien plus ancien que l'idée, certes mauvaise, d'en substituer un incorrectement. Beaucoup d'histoires mettent en scène l'électronique et les technologies de façon spectaculaire, mais elles n'ont tendance qu'à montrer l'ignorance du sujet par les auteurs. Imaginez alors ma surprise quand en lisant récemment un roman policier (*Fuzz* de Ed McBain, roman de 1968 de la série 87^e District), je me suis soudain retrouvé devant les schémas d'un circuit et une explication détaillée et crédible de la façon dont il était censé fonctionner. Rares sont les illustrations dans les romans, quelles sont les chances pour qu'il s'agisse d'un schéma ?

(160404-b)

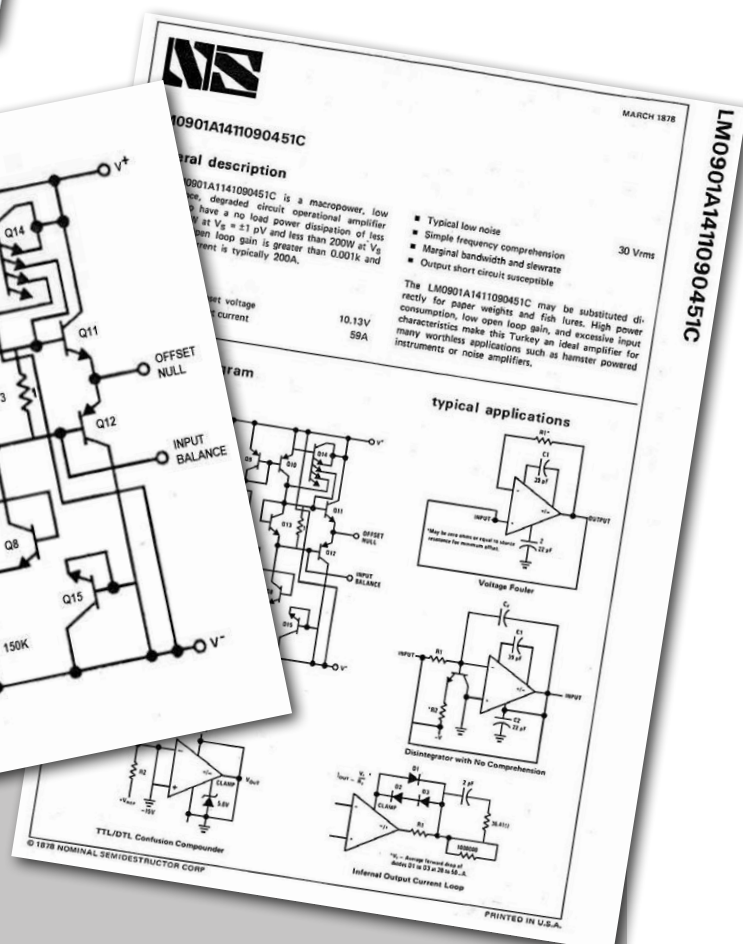
Fuzz de Ed McBain (1968)

...which is in turn set for a specific time, about eight, wasn't it? He'd have had to

schematic diagram



LM0901A1411090451C, amplificateur opérationnel (AOP) à circuit dégradé



Composants particuliers

Le **LM0901A1411090451C** est un AOP à circuit dégradé de macro-puissance, aux caractéristiques médiocres, conçu pour une dissipation de puissance sans charge de moins de 0,553 W pour $V_s = \pm 1$ pV et moins de 200 W pour $V_s = \pm 2$ pV. Le gain en boucle ouverte est supérieur à 0,001k et le courant de polarisation d'entrée typique est de 200 A.

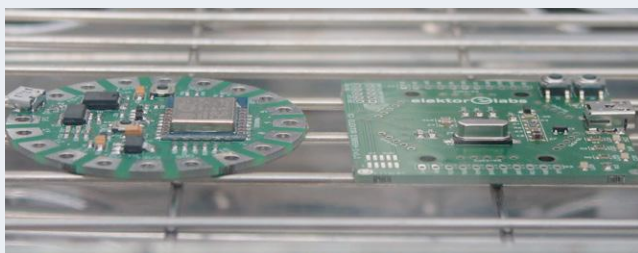
© March 1878, Nominal Semidestructor Corporation

<https://goo.gl/JF75N>

(160404-c)

Améliorez vos compétences en refusion

Vous ne l'avez peut-être pas remarqué, mais le thème principal de cette édition est la conception de circuits imprimés (CI). Comme les CI sont étroitement liés au soudage, l'art du soudage est une compétence importante à maîtriser, même pour un concepteur de CI, car la technique utilisée pour le soudage affecte la conception du CI.



Quand des composants de surface sont soudés dans un four à refusion, la température doit suivre une courbe précise — le profil thermique ou de température — pour garantir des joints de soudure corrects. On trouve les profils de température des composants dans leurs fiches techniques, et tous les composants n'ont pas le même profil. En revanche, toutes les pièces, indépendamment de leur taille et de leur masse, doivent dans un temps donné atteindre une température minimale pour démarrer le processus de refusion sans dépasser la température maximale admissible par la pièce (et les autres). Il est facile de comprendre que c'est un processus compliqué. La norme IPC-7530 pour les profils thermiques existait depuis 2001. Malheureusement elle est maintenant obsolète ; la révision A est censée voir le jour bientôt. Même si son nom suggère une mise à jour, c'est en réalité une complète réécriture. Le nouveau document fournit une myriade d'informations, des définitions jusqu'à la comparaison de soudures, des techniques de mesure des profils thermiques ou de choix du four à refusion jusqu'à la compréhension des défauts de refusion de soudure. L'objectif de cette norme est de fournir des informations utiles et pratiques aux responsables du développement de profils thermiques afin de produire des assemblages électroniques convenables avec et sans plomb. Officiellement cette norme s'adresse aux dirigeants, aux ingénieurs de conception et procédés, et aux techniciens impliqués dans les procédés de soudage de masse, mais les amateurs et autres soudeurs à refusion occasionnels pourront utiliser ce document pour parfaire leurs compétences en refusion. Ceux qui ouvrent le four pendant la phase de refroidissement pour refroidir la carte plus vite devraient lire ce document pour savoir pourquoi ils devraient se montrer plus patients.

Au moment d'écrire ces lignes, l'édition préliminaire d'août 2016 de ce document était disponible pour téléchargement ici : <https://goo.gl/188pjX>

(160404-e)

Outil de labo indispensable

Si vous vous êtes trouvé un jour dans une situation où vous auriez eu besoin d'un miroir dentaire pour examiner un dispositif inaccessible, ces lunettes pourraient être exactement ce dont vous avez besoin. Avec ses prismes à la place de lentilles ordinaires, elles permettent de voir vers le bas en regardant devant, vous voyez ainsi des choses là où votre tête ne peut pas aller. Prévu pour lire sur la plage sans avoir à relever votre lourde tête ou pour discrètement espionner les autres, les lunettes à prismes sont très amusantes. Et si on retournait un des prismes de 180° ? On pourrait alors regarder en haut et en bas en même temps ! Ou en tourner un de 90° vers la gauche et l'autre vers la droite et ressentir la vision d'un cheval.

Disponible partout sur l'internet, cherchez *Prism Glasses* ou *Bed Prism Spectacles*.

(160404-d - version française : Denis Lafourcade)



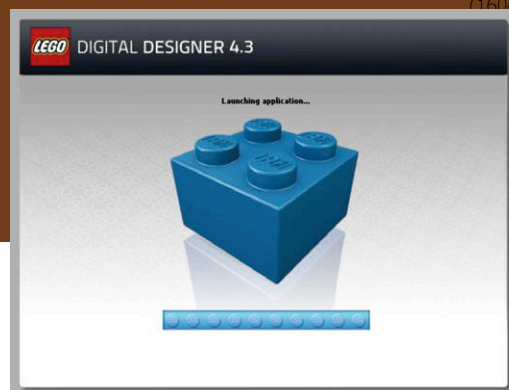
Source : Jannis Hermanns,
<https://jann.is/>

Briquez votre ordinateur

Si vous ne disposez pas de machines-outils à commande numérique ni de découpe laser, et que vous manquez de compétences en perçage et découpe, vous pouvez toujours vous tourner vers LEGO® pour construire un boîtier pour votre circuit. Beaucoup font ça pour leurs Arduino ou Raspberry Pi par ex., et j'ai même vu des graveuses faites avec les petites briques de couleur. Jannis Hermanns de Berlin (Allemagne) a réalisé un joli « clone » de Macintosh Classic à partir de briques Lego, d'une Raspberry Pi et d'un afficheur à papier électronique. Il a utilisé *Lego Digital Designer* (LDD) pour concevoir son boîtier en briques.

LDD peut être téléchargé gratuitement ici : <http://ldd.lego.com>

(160404-f)



Vous voulez participer ? Envoyez vos commentaires, suggestions, trucs et astuces à labs@elektor.com

tirez 5 V d'une cellule alcaline épuisée

essorez-la à fond

Chaque jour, nous jetons des tonnes de piles alcalines (« sèches ») de 1,5 V supposées vides. Si un appareil est alimenté par piles, dès qu'elles sont partiellement déchargées, il ne fonctionne plus. Quel gâchis ! C'est pourquoi j'ai conçu un circuit pour rehausser à 5 V environ la tension d'une cellule épuisée et donc pouvoir la réutiliser : allumage d'une LED, maintien en fonction d'un petit μC ou alimentation de capteurs peu énergivores... et ce jusqu'à ce que la pile soit **vraiment morte**.

Juan Canton (Mexico)

À partir d'une tension de pile de l'ordre de 0,8 V, ce convertisseur rehausseur de tension auto-oscillant fournit une tension de sortie à peine inférieure à 5 V. Avec un rendement de $\pm 60\%$ et un courant de sortie de 5 mA seulement, ce n'est pas une merveille d'efficacité. En fait, l'auteur voulait « vider » à fond des piles de 1,5 V épuisées en utilisant des composants faciles à trouver et bon marché. Les éléments essentiels du circuit sont l'inductance L1 et les transistors. Pour ces derniers, pensez que les 2N3906 et 2N3904 sont les transistors PNP et NPN des Amériques, remplaçables par des BC560 BC550. C'est possible, mais

pas si facile. Les capacités parasites et d'autres paramètres des composants vous joueront des tours.

Le cœur du circuit est un oscillateur, T1-T2-L1-C1, partiellement commandé par T4, qui influence la tension de sortie sur K1. La bobine (pas une self) L1, épaulée par C2 qui fait office de réservoir et de dispositif de lissage, sert au stockage de l'énergie. Remarquez la référence 1N4732A, une « diode de régulation de tension » pour laquelle NXP annonce : tension nominale de 4,7 V, I_{direct} de 53 mA, et I_{inv} de 10 μA sous 1 V pour V_{inv} . L1 est elle une « vraie » inductance, pas une self. La sélection du composant par notre stagiaire du labo a donné satisfac-

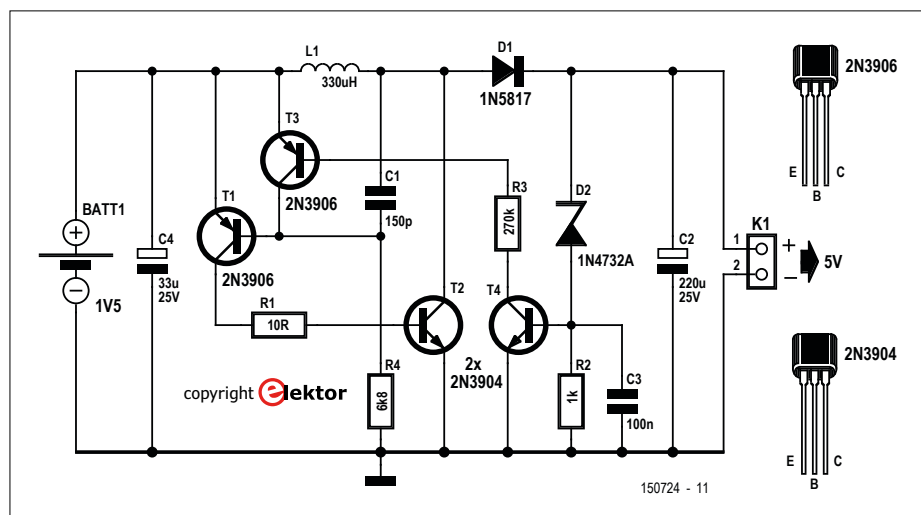
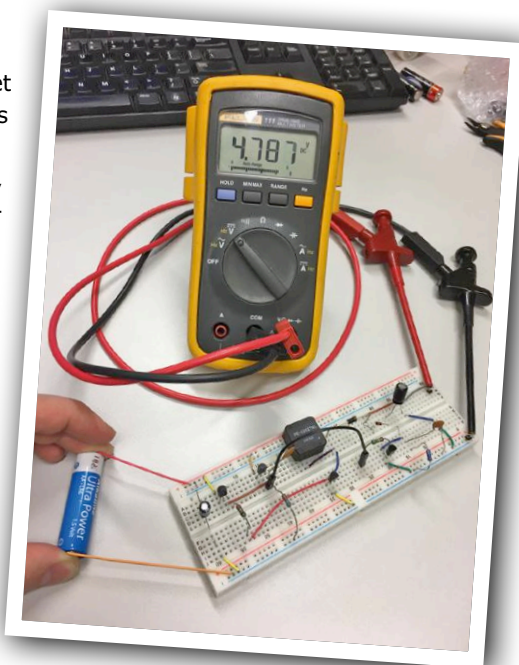


Figure 1. Un convertisseur rehausseur de 5 V pour piles de 1,5 V épuisées.

tion ; cependant, l'utilisation d'une valeur légèrement supérieure semble promettre de meilleurs résultats. La PE52627NL de Pulse Engineering (réf. 1209550 chez Farnell) est spécifiée pour une résistance CC maximale de 0,78 Ω .

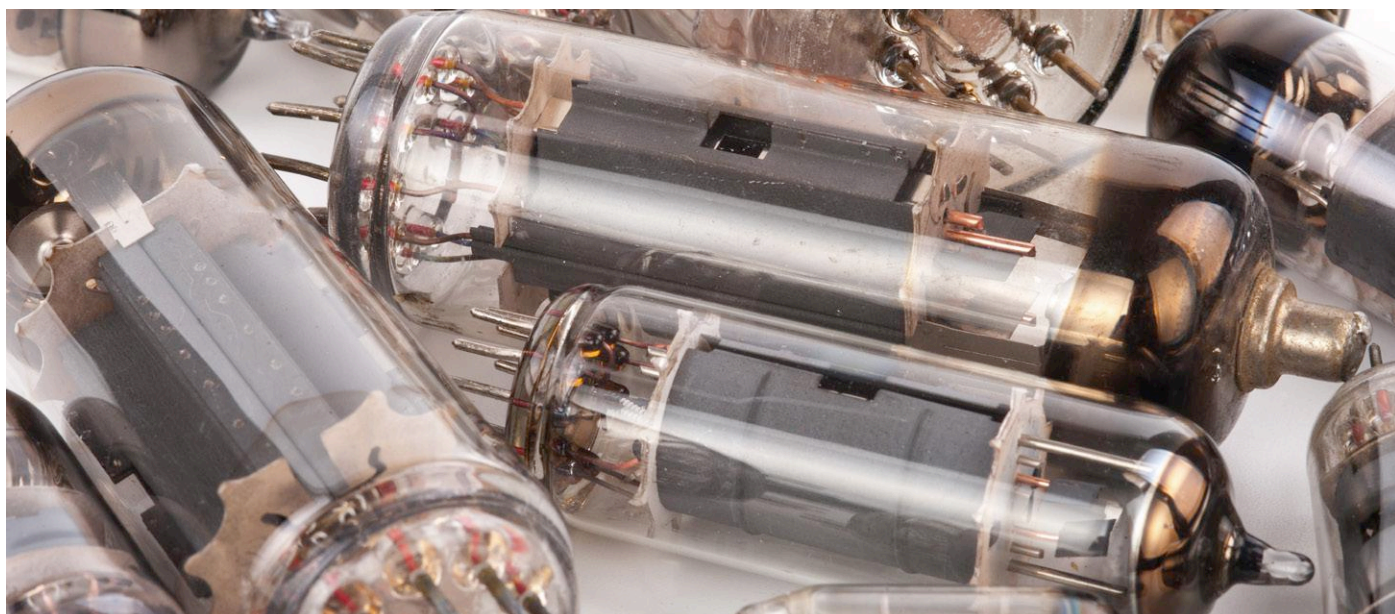
La 1N5817 est une diode de redressement Schottky (c.-à-d. à faible chute de tension) de 1 A. N'utilisez pas de diode silicium classique 1N400x, car elle fera perdre de la tension en sortie.

Note : il est possible d'alimenter le circuit avec une tension plus élevée, sans dépasser la tension de sortie ni mettre la charge en difficulté – le réseau qui règle la tension de sortie bloquera l'oscillation. Cependant, respectez certaines limites. ◀

(150724 - version française : Guy Raedersdorf)

ce bon vieux lampemètre Tektronix 570

on s'en refait un ?



Pour savoir ce que valent encore tous ces tubes électroniques de récupération, il me faut un traceur de caractéristiques, un lampemètre, disons-nous. Disposer d'un « vrai », c'est une utopie, en refaire un avec un microcontrôleur moderne, c'est plus réaliste. Je peux ainsi mesurer le courant cathodique en fonction de la tension d'anode pour différents potentiels de grille.

Charles van den Ouweland
(Pays-Bas)

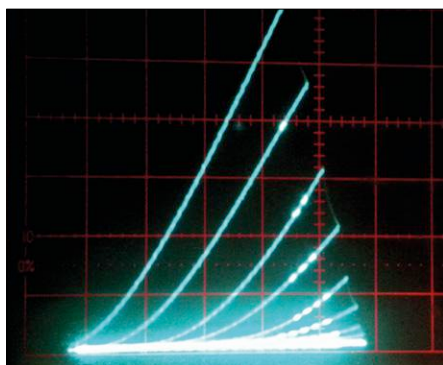


Figure 1. C'est ainsi qu'apparaissent les caractéristiques produites par le lampemètre.

Vous non plus, vous ne pouvez rien jeter ? Nous voici face à des appareils en panne, mais tellement « vintage », avec lesquels nous pourrions faire plaisir à la famille, aux amis ou aux voisins (trois catégories qui ne se superposent pas nécessairement), plutôt que de démolir des composants qui pourraient peut-être encore servir. C'est ainsi que j'ai entassé au grenier pendant des années des tubes électroniques apparemment intacts et qui ne font que s'empoussiérer.

Que faire de ces composants récupérés sur de vieux appareils ? S'en servir, bien sûr, même pour des expériences risquées. Les condensateurs, résistances, bobines et ainsi de suite, on les mesure facilement, idem pour les transistors et les diodes,

mais avec les lampes de radio, c'est une autre histoire. Les dégâts graves, on les voit à l'œil nu, les courts-circuits internes, l'ohmmètre les détecte, mais c'est à peu près tout. Pas très malin de fabriquer tout un circuit pour y monter un tube dont on ne sait même pas s'il est bon. Que ferait-on du circuit après ?

Le Tektronix 570

Si vous n'avez pas de projet en vue avec ces tubes, rien ne vous empêche de les vendre. En trifouillant un peu sur eBay, je suis tombé sur un site français qui propose des tubes accompagnés de leurs caractéristiques sur photos prises sur un vieux Tektronix 570.

Pour tracer ces caractéristiques (**fig. 1**),



l'appareil envoie sur le tube à tester différentes tensions de grille (0 V, -1 V, -2 V, etc.) et fait varier la tension d'anode de zéro à quelques centaines de volts. Sur l'écran du tube cathodique, on voit alors le courant de cathode en fonction de la tension d'anode pour ces différents potentiels de grille ce qui permet d'en déduire si le tube est capable de fonctionner dans une application précise. Ma première idée a été de faire pareil. Seulement voilà, le Tektro 570 est devenu depuis un objet de collection. Même si par chance vous en trouvez un, il faudra largement ouvrir votre portefeuille. En plus, le 570 est un vrai mastodonte, il mesure 42×33×62 cm et pèse de l'ordre de 35 kg ! Comme tout électronicien qui se respecte, je me suis dit illico : fais-le toi-même ! En partant d'un oscilloscope existant, avec des semi-conducteurs modernes et un microcontrôleur, ce doit

Figure 2. Le Tektronix 570 originel est un mastodonte au sens propre du terme.

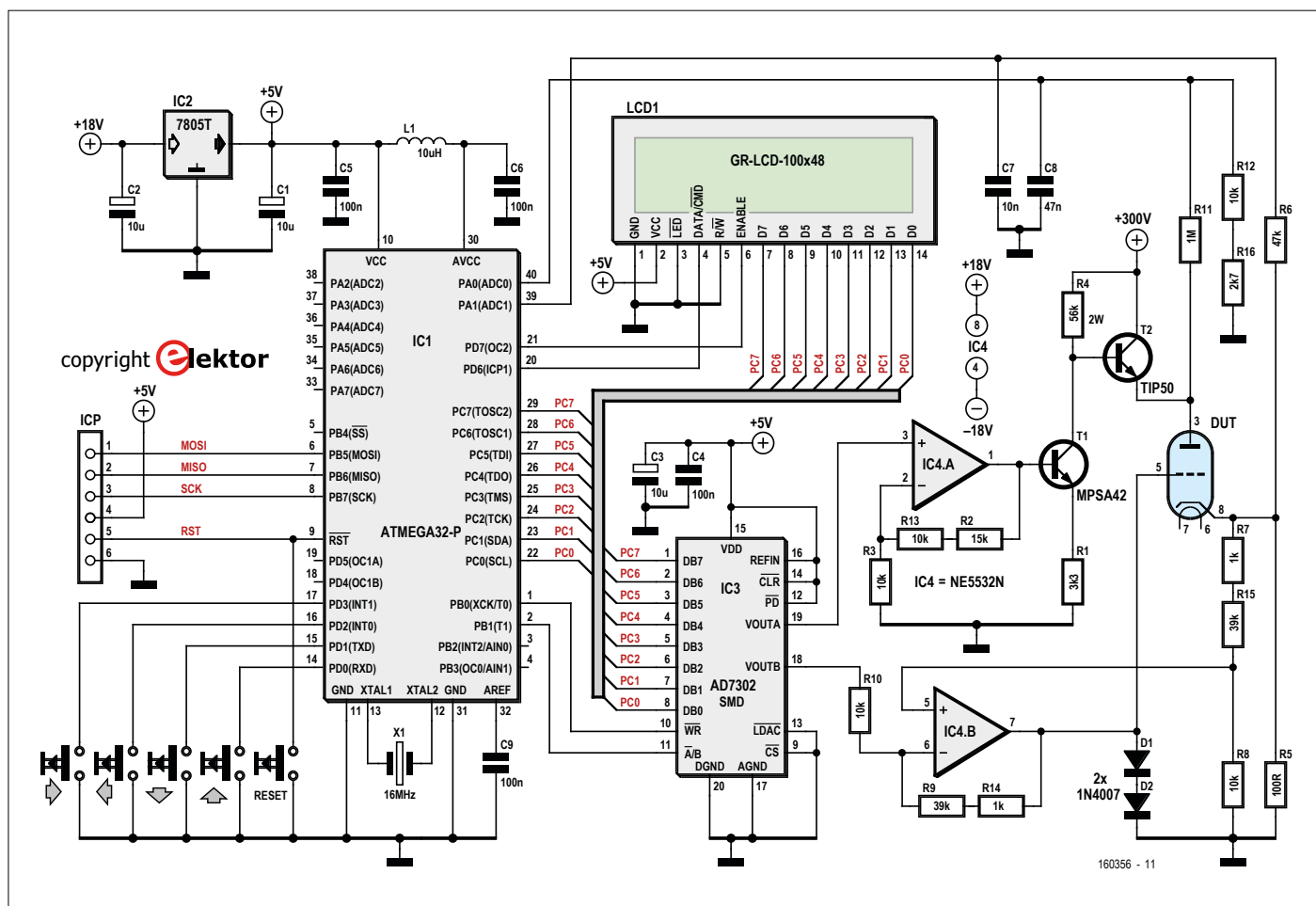


Figure 3. Le schéma du lampemètre.

être réalisable pour une fraction du prix et pèsera bien moins lourd. L'idée du Neutronix 570, comme on l'a appelé dans les couloirs, était née.

Le schéma

Le circuit est conçu pour effectuer rapidement deux sortes de mesures :

- des variations de potentiel de grille sous différentes tensions d'anode ;
- des variations de tension anodique pour différentes tensions fixes de grille.

Toutes ces opérations, un microcontrôleur les commande et un petit écran avec quatre boutons-poussoirs lui sert d'interface utilisateur. Le logiciel est ainsi fait que les caractéristiques mesurées peuvent être visibles sur LCD aussi bien que sur un oscilloscope externe. Le Neutronix 570 est et reste construit sur plaque d'expérimentation, il n'y a pas de circuit imprimé.

Survol du schéma

Le schéma du traceur de caractéristique est à la **figure 3**. À gauche, le microcontrôleur IC1 est un ATmega32-P. Il offre beaucoup de lignes d'E/S, est disponible en boîtier DIP, bon marché, d'autant meilleur qu'il m'en restait quelques-uns en stock !

À gauche de IC1, on voit le connecteur de programmation ICP (en circuit) et les quatre boutons à flèche plus celui de RàZ. Le bus de données d'une largeur de 8 bits du microcontrôleur relie d'un côté l'écran LCD1 et de l'autre, le double convertisseur N/A IC3 qui fournit de quoi former les différentes tensions nécessaires au tube à tester.

Enfin, à droite, il y a deux amplificateurs opérationnels, celui du haut, IC4.A, approvisionne à travers deux transistors à haute tension la tension d'anode pour le tube à l'essai (DUT), tandis que IC4.B, celui du bas, fournit la tension de grille.

Alimentation

Nul n'ignore que les tubes fonctionnent d'habitude avec une haute tension qui peut être très élevée. Le schéma mentionne différentes tensions d'alimentation, voyons d'où elles proviennent. C'est un simple 7805T (IC2 de la figure 3) qui assure, à partir du +18 V d'une alimentation de laboratoire, l'alimentation en 5 V de l'électronique numérique et de l'afficheur. Autant dire que ce 7805 nécessite un refroidissement suffisant ! La même alimentation de labo fournit

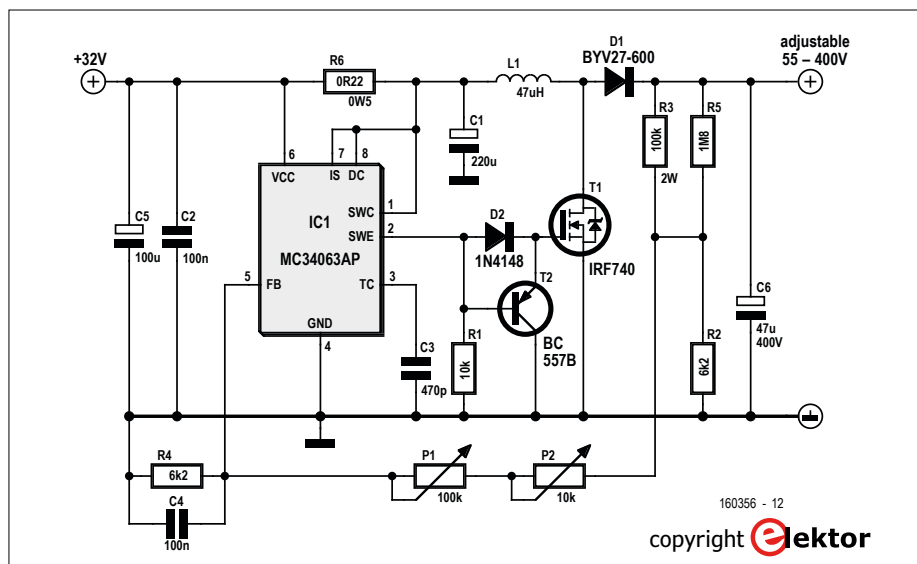


Figure 4. L'alimentation haute tension dont la tension d'entrée de 32 V est produite par une vieille carte d'imprimante.

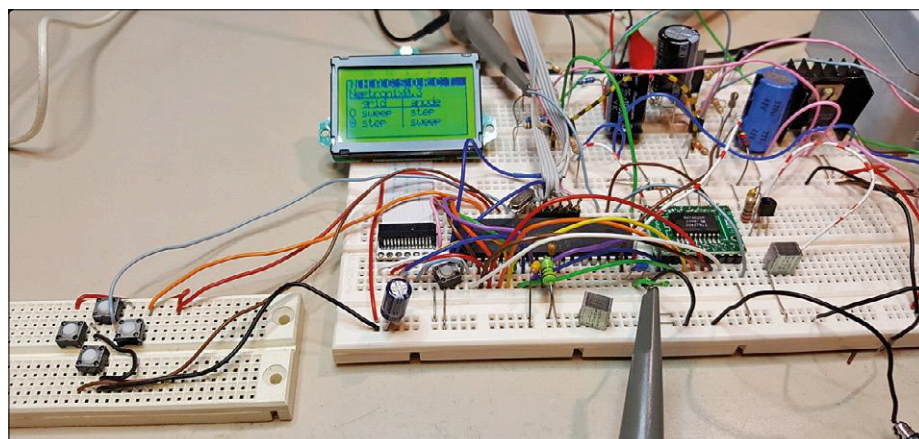


Figure 5. Mon lampemètre tel qu'il a été décrit.

aussi la tension négative de -18 V , de manière à délivrer du $\pm 18\text{ V}$ aux deux amplis op. Nous pouvons supposer que toute paillasse d'électronicien comporte une alimentation double de laboratoire, nous n'entrerons pas dans les détails à ce propos.

La haute tension, c'est plus compliqué. Pas plus que moi, vous n'avez sans doute pas sous la main une alimentation CC réglable à haute tension. J'ai donc dû trouver une solution. Ce n'est heureusement pas tellement grave, voyez le schéma de la **figure 4**. C'est une application standard du convertisseur survolteur MC34063AP.

J'ai pour ma part récupéré d'une imprimante Canon Pixma mise au rancart l'alimentation dont le convertisseur élévateur donne une tension de 32 V. Si jamais, la

tension d'entrée dont vous disposez est vraiment trop basse, vous aurez du mal à obtenir une haute tension suffisante. La solution : un doubleur de tension en sortie.

Quelques détails

Le double convertisseur N/A IC2, un AD7302, n'est hélas pas disponible à prix abordable en version DIP. J'ai donc installé la version CMS de cette puce sur une petite carte de liaison compatible avec les plaques d'expérimentation. Mon lampemètre tient donc facilement sur ce genre de supports.

La tension d'anode pour l'essai des tubes monte tout au plus à 320 V. Pour ce faire, l'ampli op IC4.A multiplie par 3,5 la tension de la sortie A du CN/A (0 à 5 V). Le transistor T1 la multiplie encore d'un



Figure 6. L'onglet de démarrage.



Figure 7. Les paramètres de l'anode.



Figure 8. Les paramètres de la grille.



Figure 9. La sélection de sortie.



Figure 10. L'onglet de mise en route.

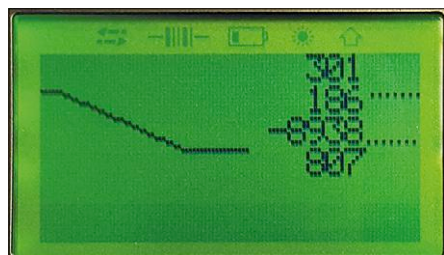


Figure 11. L'écran d'étalonnage.



Figure 12. Calcul de la pente.

facteur 17. Le transistor T2 assure un courant suffisant.

S'il ne fallait faire varier que la tension d'anode (*sweep*), les valeurs exactes de gain de IC4.A et T1 seraient sans importance. Mais je veux aussi réaliser le mode par lequel le potentiel de grille varie pour différentes valeurs de tension d'anode. C'est le but du diviseur de tension composé de R11, R12 et R16, qui répercute un quatre-vingtième de la tension d'anode vers l'entrée du CA/N ADC0 du microcontrôleur pour régler précisément le niveau voulu.

L'autre sortie du CN/A va à l'ampli op IC4.B. Comme la tension de grille doit être négative, cet ampli op inverse la tension de la sortie B du CN/A de 0 à 5 V vers le domaine entre 0 et -18 V. Les diodes D1 et D2 protègent le circuit de tout excès de tension.

La mesure du courant cathodique correspond à la chute de tension sur la résistance R5 de 100 Ω entre cathode et masse. Elle est appliquée par R6 à la deuxième entrée du CA/N ADC1 du microcontrôleur. Pour compenser la chute de tension sur R5, le diviseur de tension R7, R15 et R8 envoie un cinquième de la tension de cathode sur l'entrée positive de IC4.B.

La **figure 5** montre à quoi ressemblait le lampemètre en phase de développement.

Interface utilisateur

Pour manier le testeur de tubes, on dispose de quatre boutons-poussoirs (haut, bas, gauche, droite) et d'un afficheur LCD sur lequel s'inscrit une série d'onglets pour opérer différents réglages. On navigue entre eux au moyen des boutons à gauche et à droite, puis on ouvre l'onglet choisi par le bouton bas. Les boutons à gauche et à droite permettent alors de désigner le réglage voulu. Voyons de quoi il s'agit.

Start (N)

Dans l'écran de démarrage (**fig. 6**), on

choisit le mode de fonctionnement avec les boutons haut et bas :

- égrener les polarisations de grille sous différentes tensions d'anode (*step plate/anode, sweep grid*) ou
- égrener les tensions d'anode pour différents potentiels de grille (*step grid, sweep plate/anode*).

Le bouton flèche à droite mène à l'onglet suivant.

Anode (A)

Sous l'onglet A (**fig. 7**), on règle les tensions minimale et maximale d'anode et on choisit les valeurs avec les boutons haut/bas. En mode *step plate, sweep grid*, on sélectionne l'intervalle de niveau, c'est-à-dire la valeur à ajouter au minimum ou à retrancher du maximum.

Grille (G)

En G, ce sont les tensions extrêmes de grille et l'intervalle que l'on définit (**fig. 8**). Ce dernier détermine la valeur à ajouter au minimum ou à retrancher du maximum en mode *step grid, sweep plate*.

Output (O)

Il s'agit ici de choisir d'observer le résultat sur un oscilloscope ou sur l'afficheur LCD (**fig. 9**).

Run (R)

On entre dans le processus R (**fig. 10**) pour lancer la mesure en poussant sur le bouton bas. La mesure une fois réalisée, le résultat apparaît à l'écran. Le paramètre *sweep* s'affiche chaque fois sur l'axe horizontal.

Quelques précisions selon le type d'affichage. Sur oscilloscope, en mode *sweep grid, step anode* : la **grille** est sur l'axe des x, la cathode sur l'axe des y. En mode *sweep anode, step grid*, l'**anode** est sur l'axe des x, la cathode sur l'axe des y.

Sur l'afficheur LCD, les mesures s'effectuent assez lentement et le micrologiciel

calcule la moyenne de l'ensemble des mesures avant de passer à l'affichage. Sur oscilloscope, en revanche, les mesures arrivent très vite sur l'écran pour éviter le scintillement.

Calibrate (C)

Dans ce mode, pousser sur le bouton bas (**fig. 11**) fait voir un graphe de la tension d'anode (0 à 320 V) en fonction de la valeur du CN/A (0 à 255). Les valeurs numériques qui s'inscrivent dans cet exemple représentent successivement la tension maximale pour une valeur 0 du CN/A, la tension minimale pour une valeur de 255 sur le CN/A, la pente et le point d'intersection de la ligne droite avec l'axe des y.

D'un coup sur le bouton haut, le graphe fait place à la courbe théorique sur les mêmes axes.

Transconductance (T)

Sous cet onglet (**fig. 12**), on peut mesurer la pente S (mA/V) du tube (*transconductance*). On commence par régler la tension d'anode à sa valeur maximale sous l'onglet A, puis la tension de grille au maximum aussi, sous l'onglet G. On mesure alors le courant I1. Ensuite, on baisse d'un niveau la tension de grille pour mesurer le courant I2. La pente se calcule comme suit :

$$S = (I_1 - I_2) / \begin{matrix} \text{différence de potentiels} \\ \text{de grille} \end{matrix} \quad [A / V]$$

Dans l'exemple de la figure 12, la tension d'anode est de 200 V et les potentiels de grille 0 V et -1 V. Les valeurs mesurées sont $I_1 = 600 \mu A$ et $I_2 = 11\,000 \mu A$. Ce qui donne une pente de -10,4 mA/V.

Les onglets H (tension/courant de filament) et S (grille-écran) ne sont pas encore implémentés.

Le micrologiciel du lampemètre peut être téléchargé tel quel depuis la page de ce projet au labo d'Elektor (cherchez *vacuum tube curve tracer* sur www.elektormagazine.fr/labs). Vous y trouverez aussi les dernières informations sur les développements ultérieurs et vous pouvez y poster vos remarques et contributions propres. Un programmeur AVR ordinaire permet d'envoyer le micrologiciel dans le microcontrôleur par l'interface de programmation ICP. ■

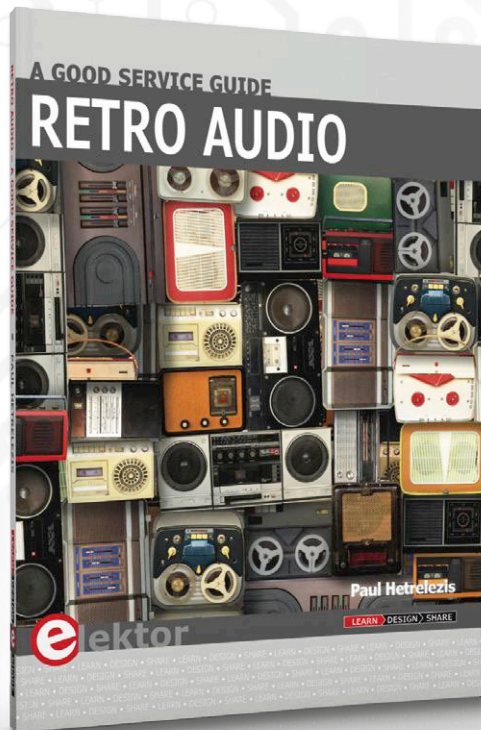
(130356 - version française : Robert Grignard)

L'auteur

Charles van den Ouweland (Breda, 1963) a étudié l'électrotechnique à l'université technique d'Eindhoven (Pays-Bas). Il travaille actuellement comme architecte de logiciel chez ProRail, mais continue de s'occuper d'électronique pendant ses temps libres, principalement de combinaisons entre les microcontrôleurs modernes et les « bons vieux » tubes.



Retro Audio (livre en anglais)



Aujourd'hui il y a un regain d'intérêt pour les disques vinyle et les tourne-disques, ce qui a donné naissance à un marché parallèle de la réparation. Le but de ce livre (en anglais) est d'apporter au lecteur, électronicien amateur ou confirmé, des connaissances ainsi que des solutions en matière d'audio d'antan. Cet ouvrage contient des tableaux avec des données utiles, des trucs et astuces pour l'entretien ainsi que des informations pas si communes.

Ce guide d'entretien est très concret. Il commence par un tour de l'atelier de réparation et des dispositifs de test et outils nécessaires. Ensuite deux chapitres sont consacrés aux sources d'information et d'approvisionnement en pièces détachées. Le lecteur est alors prêt à effectuer des diagnostics et des tests, notamment sur les résistances, condensateurs et inductances. Grâce à ce livre, il apprend à gagner du temps tout en effectuant des réparations de qualité. Les derniers chapitres sont consacrés aux problèmes spécifiques à chaque appareil : lecteur de cassette, tourne-disque, lecteur de CD, micros...

www.elektor.fr/retro-audio

elektor



projet 2.0

corrections, mises à jour et courrier des lecteurs

e-BoB BL600 (1)

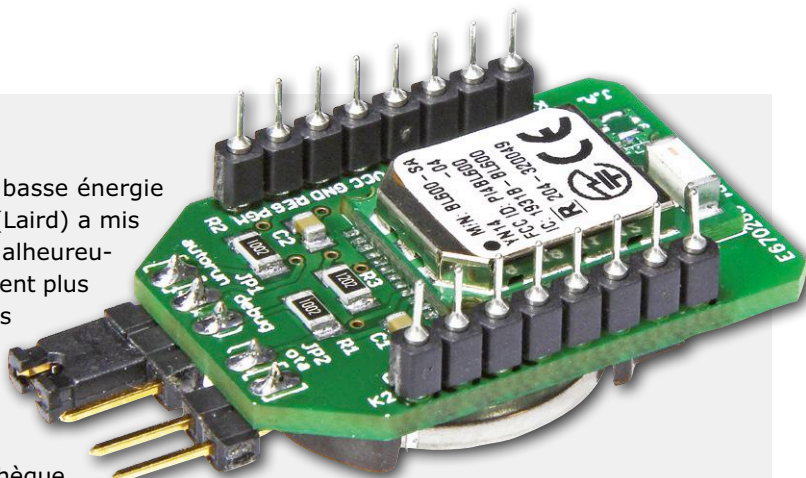
Elektor 03/2015, p. 34 (140270)

Depuis le premier article sur le module Bluetooth à basse énergie BL600 dans le numéro de mars 2015, son fabricant (Laird) a mis à jour son logiciel (micrologiciel et bibliothèques). Malheureusement, il semble que les versions précédentes ne soient plus disponibles. L'exemple proposé dans la série d'articles de 2015 est basé sur la version 1.5.7.0 révision 5 du micrologiciel et les bibliothèques associées. Il est possible qu'il ne soit pas compatible avec le nouveau micrologiciel et les nouvelles bibliothèques.

En tout cas, il est important de n'utiliser que la bibliothèque associée à un micrologiciel ou une révision donnés. Si vous obtenez le diagnostic « BL600: Cross Compiler not found », votre micrologiciel et la bibliothèque téléchargée ne sont pas compatibles. Les commandes AT pour identifier le micrologiciel et la révision du BL600 sont les suivantes (il faut commencer par connecter le module au PC par une liaison série) :

- micrologiciel : AT I 3<Enter>
- révision : AT I 0<Enter>

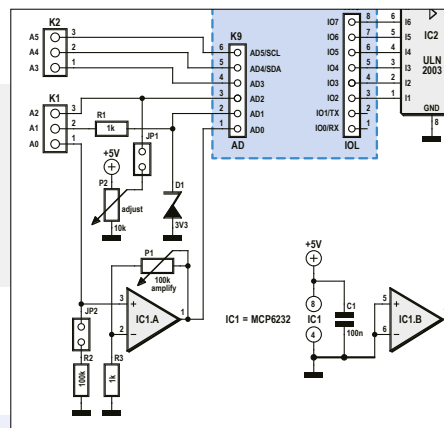
Voir aussi l'article « eBoB BL600 (2^e partie) » (www.elektormagazine.fr/150014) pour plus de détails sur le micrologiciel. L'article « passerelle Bluetooth 2 vers Bluetooth LE » (www.elektormagazine.fr/150753) contient la procédure de programmation du micrologiciel du BL600. La reprogrammation du BL600 requiert un programmeur Segger J-Link EDU ou compatible.



shield IdO pour Arduino

Elektor 01/2017, p. 17 (160169)

Dans le deuxième paragraphe de cet article, lire « Entrée analogique 2 » au lieu de « Entrée analogique 3 »



bienvenue dans la section PARTAGER

Elektor 01-02/2017, p. 108 (160252)

La question du rédacteur d'Elektor Thijs Beckers « pourquoi la polarité des prises de courant des chauffe-eau peut jouer un rôle » a reçu beaucoup de réponses, merci à tous ! En voici une petite sélection :

Le problème provient du fait que la symétrie de distribution du courant n'est pas conservée au niveau du consommateur. Un exemple que je connais depuis longtemps concerne les tubes luminescents. Comme le transformateur de ballast est conçu comme transformateur économiseur, le branchement n'est pas symétrique, c'est-à-dire qu'il dépend si c'est la phase ou le neutre qui est relié à la base du transformateur. Dans l'un des cas, on observe dans l'obscurité une petite lueur du tube, qui disparaît dans l'autre cas.

Lothar Freißmann

La cause du problème est sans doute une électrolyse. Il y a toujours une petite composante continue sur le réseau alternatif, ce qui provoque parfois la formation d'un dépôt sur les électrodes. On peut l'éviter dans une certaine mesure par une mise à la terre de l'appareil.

Quand on retourne la prise, l'effet électrolytique s'inverse. En fait, le dépôt passe sur l'autre électrode de l'élément chauffant.

J'ai ici, au Canada, un problème analogue, c'est-à-dire un dépôt de calcaire dans mon chauffe-eau rapide. J'ai équipé l'élément chauffant d'un tuyau en parallèle afin de pouvoir le détartre plus facilement : avec une pompe, j'y fais circuler du vinaigre en circuit fermé aussi longtemps qu'il me paraît nécessaire.

Clinton Millet, Canada

Le manuel de mon chauffe-eau précédent « Remeha Calenta » mentionnait un paramètre 43 « Détection de la polarité du réseau ». Après l'activation, on aurait l'affichage d'un code d'erreur en cas de prise de courant mal branchée. Ce qui est possible en Allemagne et aux Pays-Bas, mais pas en Belgique et en France, où la géométrie des prises de courant l'interdit. Après retournement de la prise, le code d'erreur SU9 s'affiche : prise branchée à l'envers. Le réglage d'usine du chauffe-eau mettait le paramètre 43 à 0 : la prise mal branchée n'aurait donc pas été détectée.

Après l'allumage, l'électrode d'allumage est utilisée pour la mesure du courant d'ionisation afin de s'assurer de la bonne mise à feu. Selon le manuel, le courant d'ionisation ne doit pas dépasser 3 μ A. Dès lors, les divers kits d'entretien du chauffe-eau contiennent préventivement une électrode de rechange. La haute tension est produite par un régulateur à découpage susceptible de créer davantage de parasites si la prise est mal enfichée. En voilà peut-être un peu trop pour un si petit courant de mesure sur une électrode presque complètement usée.

Peter van de Meerendonk



broches d'alimentation d'un ampli-op...

Elektor 01-02/2017, p. 110 (160257)

Comme j'ai travaillé à la conception de circuits intégrés, je peux imaginer la raison suivante de l'existence de brochages différents pour un même circuit intégré fabriqué en grande série : il y a des clients importants qui ont besoin d'un brochage différent pour simplifier le routage de leurs propres cartes ; le fabricant n'a alors qu'à modifier les connexions et le marquage, pour un coût marginal.

Lothar Freißmann



microphone MEMS minuscule

Elektor 11/2016, p. 70 (160083)

L'article prétend qu'il est difficile de se procurer des microphones spéciaux pour les ultrasons. Pas vrai ! ☺

En tant que vieil oiseau de nuit (quand, sinon, a-t-on la paix ?) j'ai aussi un intérêt particulier pour les chauves-souris. J'ai déjà construit plusieurs convertisseurs pour ultrasons : tout d'abord purement numériques (avec un diviseur 7493) ensuite à mélangeur linéaire abaisseur de fréquence (avec un SO42P). Pour cela il me faut bien entendu des micros de qualité raisonnable.

Vous en trouverez sur l'internet à très bon prix : cherchez « HC-SR04 Ultrasonic Range Finder Distance Measuring Module Sensor Arduino Pi Pic ».

La carte ne contient pas seulement un microphone récepteur, mais une puce qu'on peut aussi utiliser pour d'autres expériences, et un émetteur d'ultrasons optimisé comme « haut-parleur » (ohé, les vieilles zappettes de téléviseur à ultrasons !). Encore moins cher : Ebay propose ces cartes par lots de cinq ; et nul besoin d'une loupe pour trouver les composants. Il s'agit d'unités à électrets, utilisables avec toutes sortes de détecteurs de chauves-souris sans grosses modifications.

Et voici un autre avantage : comme l'ouverture du micro est assez large, la partie réception s'adapte à l'utilisation d'un réflecteur parabolique (facile à construire soi-même en papier rigide), si bien qu'on peut réaliser une antenne directionnelle permettant de localiser avec précision les mammifères volants.

Pour répondre d'avance à la question : oui, les micros fonctionnent jusqu'à plus de 100 kHz, ce qui n'est d'ailleurs guère nécessaire, car la plupart des chauves-souris d'ici ne « montent pas si haut ».

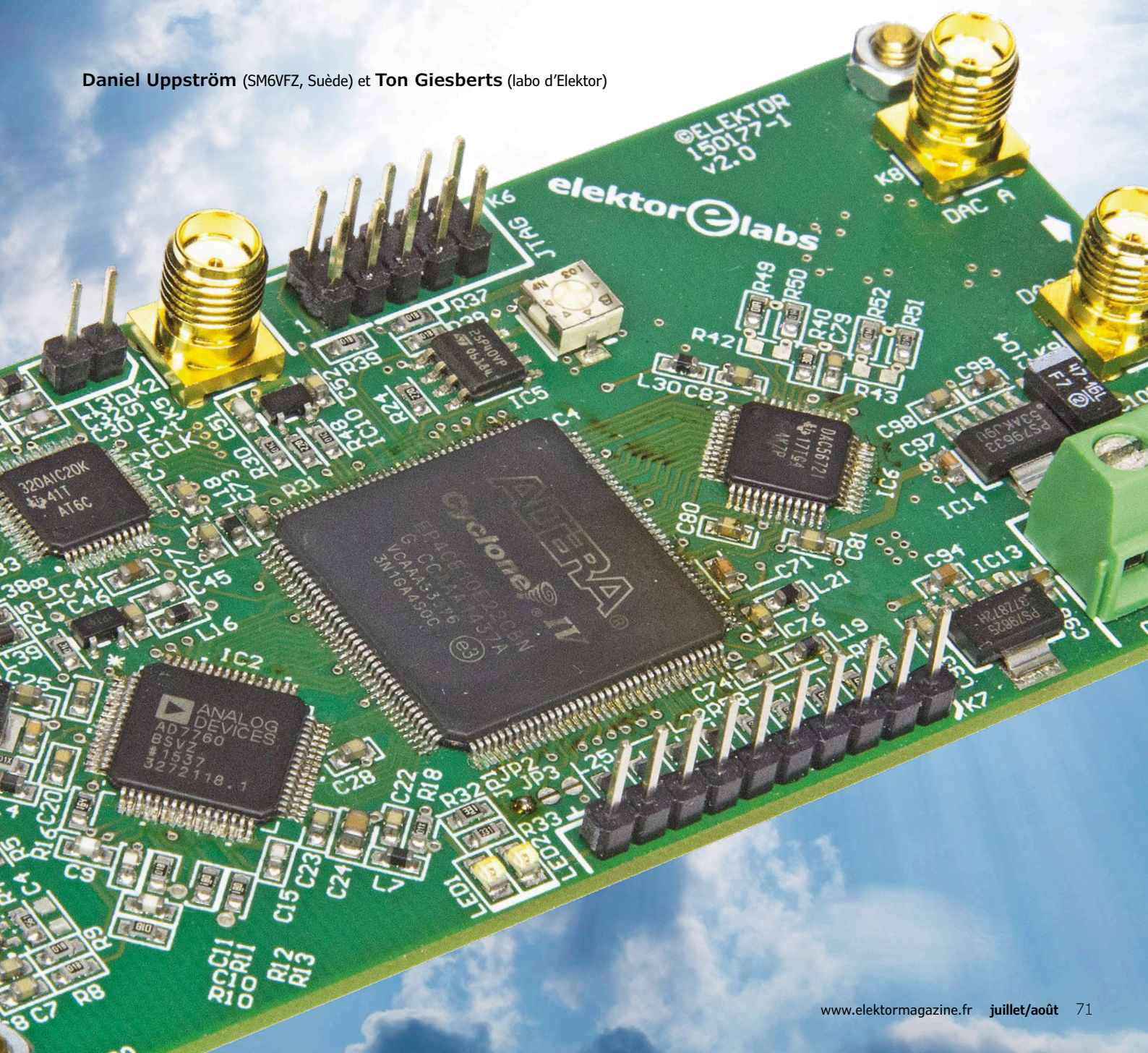
Peter Krengel



carte DSP à FPGA pour SDR à bande étroite

émetteur-récepteur
complètement programmable

Daniel Uppström (SM6VFZ, Suède) et Ton Giesberts (labo d'Elektor)



Voici une carte de traitement du signal numérique basée sur un FPGA qui comporte tout le nécessaire pour réaliser le travail en bande de base des modes traditionnels à bande rétrécie, comme SSB, radiotélégraphie (CW) et modulation d'amplitude (AM). Elle apporte une solution à de nombreuses difficultés et limitations rencontrées avec les plateformes SDR simples. Capable d'émettre autant que de recevoir, elle réunit les éléments indispensables à un transceiver en ondes courtes, VHF, UHF SHF et micro-ondes, elle est idéale pour toutes les applications radio scientifiques, radioamateurs et autres missions sans fil sérieuses.

Du bon vieux temps...

Il fut un temps où la technique radio focalisait l'attention d'une foule d'amateurs enthousiastes et d'ingénieurs en électronique. Nombre d'entre eux ont construit leurs propres appareils. Avoir une licence de radioamateur signifiait qu'on s'y connaissait en électronique. Avec les avancées technologiques, les amateurs ont eu plus de mal à suivre et la plupart ont fini par acheter des postes tout faits. La crème des émetteurs-récepteurs des années 80 et 90 faisait appel à de nombreux blocs analogiques complexes, hors d'atteinte des amateurs qui auraient voulu les imiter, non par manque de connaissances, mais en regard du prix et du volume.

Les progrès de l'internet ont permis une nouvelle approche des moyens de communication et les transceivers du commerce ont commencé à reculer. En acceptant des compromis dans le but de réduire la taille des radios et leur prix, on couvrait encore un large spectre de fréquences. La baisse des prestations essentielles des radios, sélectivité du récepteur et clarté du signal transmis, est

partiellement compensée par des décorations numériques comme les afficheurs couleur. Pourtant, sans se laisser impressionner par l'apparence et les gadgets, pour qui préfère la qualité radiophonique, il est à présent de nouveau possible de se lancer dans une production personnelle.

... à la radio logicielle (SDR)

Cependant, retourner à la technique analogique de pointe n'est sans doute pas la meilleure solution, il est plus intéressant de regarder ce qui se passe dans la sphère de la radio logicielle, où la majorité du traitement du signal radio se fait en numérique et dans le logiciel. La puissance actuelle des processeurs et la qualité des convertisseurs d'analogique à numérique (CA/N) disponibles peut vraiment aider dans la technique SDR. Beaucoup d'applications pour électroniciens enthousiastes et radioamateurs traitent totalement les signaux numérisés par logiciel sur PC ou processeur numérique de signal (DSP). Une méthode souple, mais assez peu efficace du fait que, sans l'aide de multiplicateurs matériels et de ce genre de périphériques, le traitement du signal requiert énormément d'opérations. Parmi les plateformes SDR populaires, on en trouve démunies d'un bon filtrage analogique avant numérisation, avec une basse résolution du CA/N et par conséquent de pauvres prestations spécialement sous de puissantes interférences.

Il est préférable de construire en partie antérieure une bonne radio analogique avant de recourir au traitement du signal par logique programmable. De nos jours, les FPGA sont disponibles à des prix abordables vu le nombre impressionnant de portes logiques, de mémoires et de blocs multiplicateurs prêts à l'emploi. C'est vraiment ce qu'il faut pour le traitement de signaux complexes.

Faisons-la nous-mêmes

Daniel a lancé le projet en 2013 en construisant un prototype et écrivant le code VHDL pour les blocs DSP. Par rapport à un code d'exécution DSP créé à

partir de zéro, il est souvent plus facile de mettre en œuvre une chaîne de traitement de signaux complexes en logique programmable, puisqu'on peut définir chaque étape indépendamment des autres. Il serait incommode d'expliquer ici ce processus de développement, mais tout le code écrit et toutes les informations nécessaires sont à votre disposition gratuitement sur [1].

Malgré toutes ses qualités, la combinaison seule de logique programmable et de CA/N ne donnera jamais une radio. Il lui faut aussi un appareil à radiofréquence pour effectuer la conversion de fréquence, le filtrage analogique et l'amplification. Cette construction fera l'objet d'une présentation dans une suite de cette série. Un panneau de commande pour la syntonisation de la radio, le réglage de volume, etc. suivra également. Ces trois ensembles formeront alors une radio complète dans un boîtier adéquat.

Notez que le système peut aussi être commandé par un PC branché par l'intermédiaire d'un adaptateur USB/série ou un Raspberry Pi qui communique par son port I²C ou sériel. Une interface graphique écrite en Python est aussi disponible pour montrer le fonctionnement.

Description du circuit

La carte DSP à FPGA dispose d'un CA/N à 24 bits pour échantillonner à la fréquence intermédiaire en mode récepteur, un FPGA Cyclone IV pour le traitement du signal, un CN/A à haute vitesse pour l'oscillateur local et la production du signal à transmettre, une interface audio pour microphone et haut-parleur, un oscillateur très stable à haute précision et une interface I²C ou UART pour un contrôleur hôte (**fig. 1**).

Le cerveau de la carte DSP à FPGA est un FPGA EP4CE10 Cyclone IV d'Intel, anciennement Altera (IC4). On peut configurer cette puce dans pratiquement n'importe quelle fonction numérique. Sous la houlette du micrologiciel, les circuits logiques sont configurés au démarrage à partir d'une mémoire

INFOS SUR LE PROJET



FPGA DSP SDR

TX RX

radio amateurs



débutant

connaissableur

→ expert



env. 4 h



soudage de CMS, programmeur JTAG (USB Blaster), Quartus Lite



env. 200 €

externe (IC5) dite de configuration. Pour l'entrée et la sortie audio, il y a un CODEC (codeur/décodeur) TLV320AIC20K (IC3) qui contient deux canaux CA/N et CN/A à 16 bits de résolution avec un taux d'échantillonnage maximal de 25 000 échantillons par seconde (éch./s). Il possède un filtre passe-bas numérique intégré à 8 kHz à côté d'un amplificateur de microphone avec étage de sortie capable de fournir 250 mW sur 8 Ω . Toutes les entrées et sorties audio sont équipées d'un réglage programmable du gain et de l'atténuation. On branche le HP sur K2, le microphone sur K3, tous les autres canaux disponibles sont reliés à K4 pour servir à d'autres appareils.

L'entrée principale de la carte, K1, accepte un signal différentiel, normalement dans la gamme de centaines de kHz. Le signal passe par l'amplificateur différentiel IC1 et un filtre passe-bas discret avant d'être numérisé sur 24 bits par le CA/N IC2, un AD7760. Cette puce a plusieurs connexions d'alimentation séparées par tension et des filtres passe-bas. C'est le FPGA qui lui donne son horloge principale après que IC8 en ait remonté le niveau à 5 V.

Il y a encore sur la carte IC6, un CN/A rapide à deux canaux, le DAC5672. Ses deux sorties différentielles A et B sont rendues asymétriques par transformateur, puis subissent un filtrage passe-bas qui permet de produire des signaux jusqu'à 50 MHz. Les deux sorties de CN/A sont disponibles sur les connecteurs coaxiaux du type SMA K8 et K9. IC7, un oscillateur à quartz compensé en température (TCXO), fournit l'horloge principale à 20 MHz de la carte. Ce signal est envoyé au FPGA qui le distribue aux circuits périphériques. Le réglage fin de la fréquence se fait avec P1. Le connecteur K5 permet de relier une source de référence externe, si on en dispose. Dans ce cas, le TCXO sera éteint par T1. Les signaux du TCXO et la référence externe sont tous deux envoyés au FPGA en passant par des inverseurs câblés sans tampon et comme ce sont des amplificateurs analogiques (IC9 et IC10), l'amplitude peut rester modérée.

Le signal d'horloge subit une division dans le FPGA pour obtenir une basse fréquence pour la LED1 qui indique que l'horloge fonctionne et que le FPGA est configuré. LED2 est réservée.

L'alimentation de 5 V de la carte est connectée à K10 d'où elle est distribuée

Caractéristiques techniques

- FPGA + DSP + Audio CODEC
- Radio logicielle à bande étroite
- Opère à toute fréquence radio entre 0 et plusieurs GHz
- Architecture superhétérodyne
- Seconde FI non nulle
- Alternativement récepteur (RX) et émetteur (TX)
- Modulateur démodulateur SSB Weaver

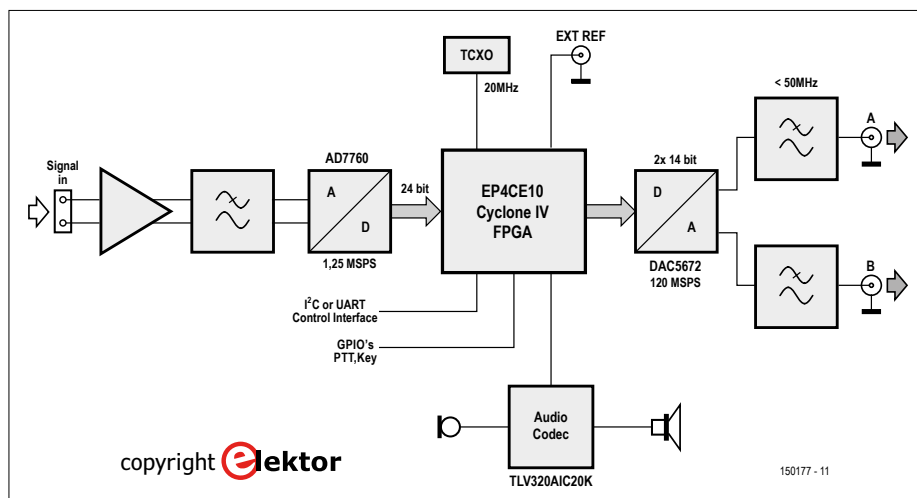


Figure 1. Synoptique de la carte de DSP à FPGA.

aux quatre régulateurs IC11 à IC14 à faible déchet (LDO) pour former les tensions de 1,2 V, 1,8 V, 2,5 V et 3,3 V nécessaires aux différents composants sur la carte.

L'interface vers le contrôleur hôte pour la communication par I²C ou UART arrive sur deux broches sélectionnées par le cavalier soudé JP1. Ces lignes sont disponibles sur K7 parmi sept autres lignes d'E/S destinées normalement au bouton-poussoir d'activation du microphone (PTT) et au manipulateur morse. Ces

broches peuvent aussi servir pour l'E/S audio I²S.

Les cavaliers JP2 et JP3 n'ont pas de fonction dans le micrologiciel actuel. Le connecteur K6 fournit l'interface JTAG pour programmer le FPGA et la mémoire de configuration.

Topologie radio

Avec une carte radio adéquate, on peut faire travailler la carte DSP à FPGA en principe à n'importe quelle fréquence entre 0 et plusieurs gigahertz. Les

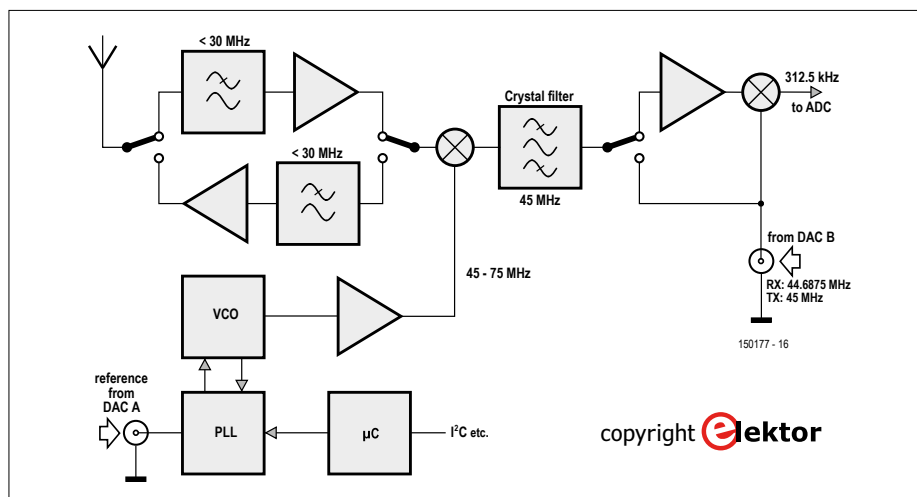


Figure 3. Synoptique d'une carte radio supplémentaire.

applications typiques sont quand même en ondes courtes (OC), en dessous de 30 MHz et dans la bande amateurs de 2 m (144-146/148 MHz).

Pour la réception des OC, on peut se construire une radio simple comme à la **figure 3**. On fait passer le signal d'antenne par un filtre passe-bas avant de l'amplifier. En le mélangeant avec le signal d'un oscillateur local (LO) issu du CN/A A, on convertit vers 45 MHz, la fréquence voulue, puis on l'envoie à un filtre à quartz. On amplifie le signal à fréquence intermédiaire (IF) avant de l'appliquer à un second mélangeur qui le convertit vers une nouvelle FI centrée sur 312,5 kHz (nous y reviendrons) à l'aide d'un second oscillateur local fourni par le CN/A B. Le second signal FI est alors numérisé par le CA/N et transmis au FPGA où il sera traité et finalement démodulé dans la gamme audio.

En comparaison de nombreuses autres radios logicielles, cette structure est avantageuse pour deux raisons.

1. Principe du superhétérodyne à double conversion

Le signal est d'abord mélangé pour obtenir une FI élevée pour le filtrer. Puis il est converti vers une FI plus basse avant la numérisation. C'est utile parce que les signaux puissants à des fréquences différentes de celle à recevoir sont éliminés par le filtre à bande étroite situé en amont de la chaîne. Tout bénéficie pour la sélectivité comme pour la plage dynamique, ce qui permet de recevoir des signaux faibles dans le voisinage de puissants.

Les projets simples de SDR utilisent souvent le principe de la FI zéro : un seul mélangeur, pas de filtre et juste une numérisation à fréquence audio autour de la porteuse. Cela limite la possibilité de traiter des interférences sérieuses, du fait que des produits de mélange indésirables peuvent alors atterrir dans la gamme audio et provoquer la distorsion du signal utile.

Des solutions plus particulières consistent à échantillonner simplement une large portion du spectre de fréquence (*direct sampling*) et effectuer l'accord dans le domaine numérique. Une méthode qui permet d'observer de larges plages du spectre, mais quand on veut conserver une bonne sélectivité et une ample dynamique, les impératifs de linéarité de l'étage d'entrée et de résolution utile du CA/N deviennent très stricts. De

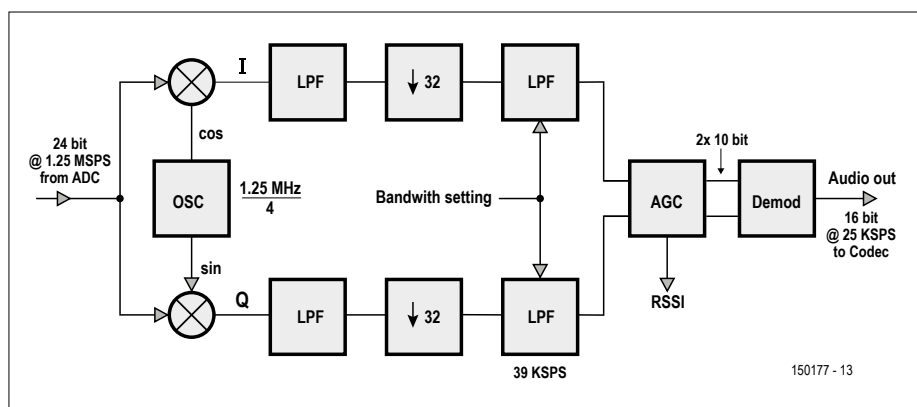


Figure 4. La chaîne de traitement du signal implémentée dans le FPGA.

hautes fréquences d'échantillonnage et une haute résolution impliquent d'habitude des coûts élevés et de fortes demandes de puissance résultent souvent en pauvres compromis.

2. La seconde FI est basse, mais pas nulle

Ceci signifie qu'un signal d'interférence dans la bande passante du filtre (normalement 15 kHz) ne peut pas se mélanger pour produire une fréquence qui interfère avec le signal reçu. Ce genre de produits de mélange seront échantillonnés, mais éliminés numériquement après parce qu'ils sont renvoyés sous la fréquence voulue du signal centrée sur 312,5 kHz. Comme cette fréquence n'est pas très élevée, elle ne requiert qu'un taux d'échantillonnage dans la gamme de 1 à 2 Méch./s pour lequel on trouve des CA/N de 24 bits d'un prix

raisonnable. Avec 24 bits, la plage dynamique numérique est de plus de 100 dB, ce qui rend superflu le contrôle de gain dans les blocs analogiques et du même coup simplifie énormément les circuits du récepteur.

Traitement du signal numérique – réception

C'est le boulot du FPGA de traiter le signal d'entrée numérisé. La **figure 4** est un synoptique simple des blocs utilisés. Le CA/N numérise le signal d'entrée centré sur 312,5 kHz à un taux de 1,25 Méch./s. Avec 24 bits par échantillon à 1,25 Méch./s, il y aurait à traiter 30 Mbit/s. C'est nettement plus que nécessaire, les premiers traitements numériques consistent donc à déplacer vers le bas et à sous-échantillonner le flux d'entrée. Le signal est multiplié par deux séquences : [0, 1, 0, -1, 0,...] une

Listage 1. Implémentation en VHDL d'un convertisseur abaisseur au ¼ du taux d'échantillonnage.

```

if ns = 0 then
  Ia(write_pointer) <= signed(Data_in); -- 1
  Qa(write_pointer) <= to_signed(0,24); -- 0
  ns := 1;
elsif ns = 1 then
  Ia(write_pointer) <= to_signed(0,24); -- 0
  Qa(write_pointer) <= signed(Data_in); -- 1
  ns := 2;
elsif ns = 2 then
  Ia(write_pointer) <= (not signed(Data_in)) + 1; -- -1
  Qa(write_pointer) <= to_signed(0,24); -- 0
  ns := 3;
elsif ns = 3 then
  Ia(write_pointer) <= to_signed(0,24); -- 0
  Qa(write_pointer) <= (not signed(Data_in)) + 1; -- -1
  ns := 0;
end if;

```

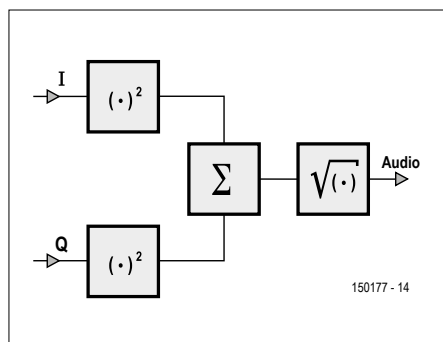



Figure 5. Démodulation du signal à modulation d'amplitude (AM).

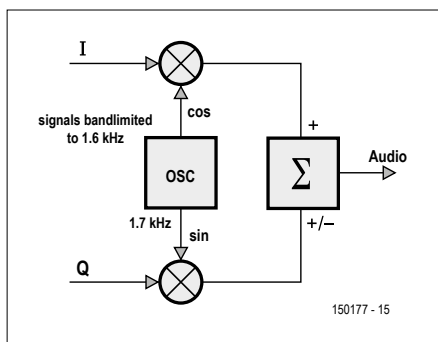


Figure 6. Démodulation des signaux à bande latérale unique (SSB) et à onde entretenue (CW).

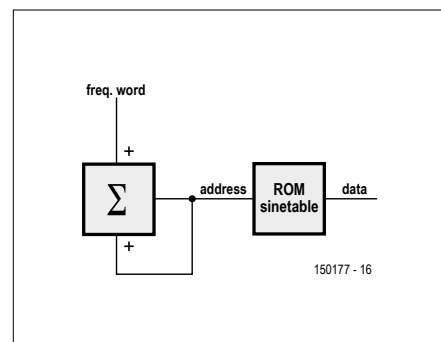


Figure 7. Le principe opératoire du DDS. Le bloc marqué d'un sigma est l'accumulateur de phase.

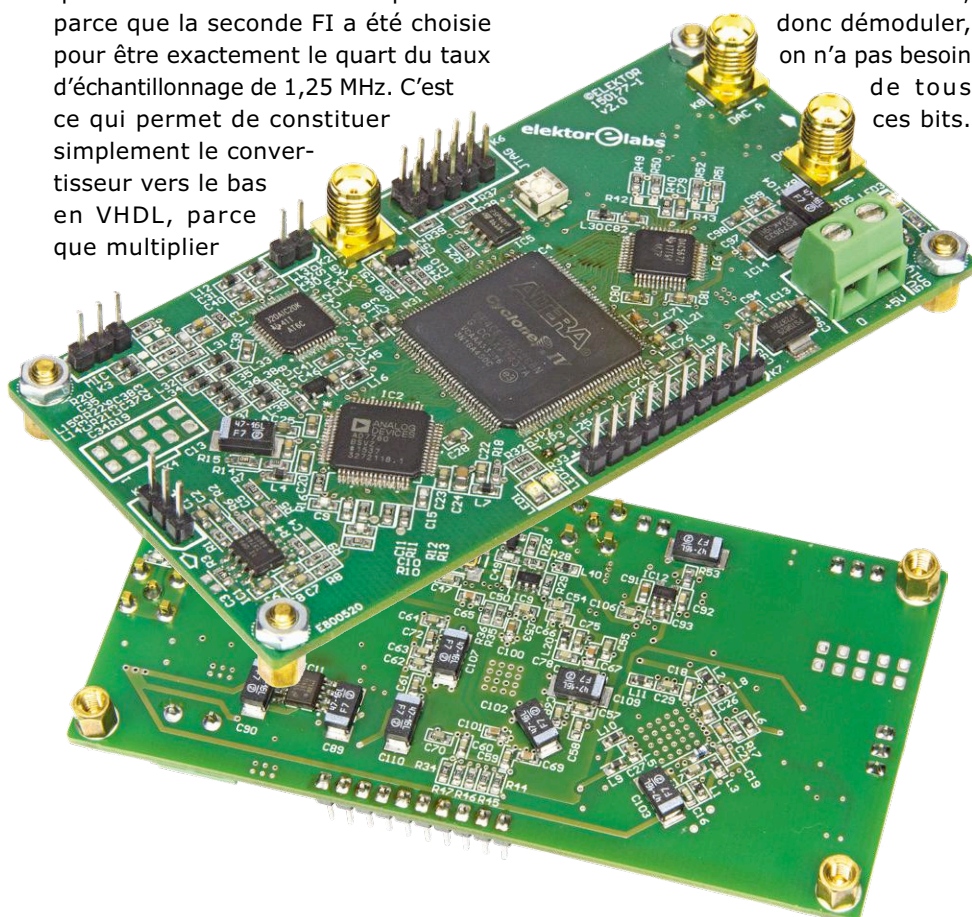
onde sinusoïdale exactement au quart du taux d'échantillonnage et [1, 0, -1, 0, 1, ...] une onde cosinusoidale (identique à la première séquence, mais décalée d'une position, soit 90 °) ; le signal à 312,5 kHz devient un signal en quadrature centré sur 0 Hz. Ces signaux sont alors transmis à des filtres passe-bas qui les divisent par 32 (simplement en éliminant 31 échantillons sur 32) pour arriver à un taux voisin de 39 kéch./s. Ces deux signaux sous-échantillonnés contiennent maintenant chacun une information de 0 à env. 10 kHz ; combinés, ils représentent l'information qui se situait à ± 10 kHz autour de 312,5 kHz dans le signal d'entrée.

Si vous doutez de l'explication, sachez que la formule marche uniquement parce que la seconde FI a été choisie pour être exactement le quart du taux d'échantillonnage de 1,25 MHz. C'est ce qui permet de constituer simplement le convertisseur vers le bas en VHDL, parce que multiplier

par 1, 0 et -1 est facile (**listage 1**). Avec d'autres fréquences, il aurait fallu mettre en œuvre une table de sinus et des multiplicateurs à 24 bits.

Dans l'étape suivante, on soumet encore les deux signaux à un filtre passe-bas. Ici, les fréquences de coupure doivent s'adapter au mode opératoire choisi. En AM, elles sont à 5 kHz pour une largeur de bande de 10 kHz. En téléphonie en SSB (bande latérale unique), elles sont aux alentours de 1,6 kHz (bande passante de 3,2 kHz) et pour la bande étroite CW (tout ou rien, code morse), on ne filtre que quelques centaines de hertz. Une résolution de 2×24 bits représente encore une énorme quantité de données à interpréter. Pour en extraire l'information intéressante,

donc démoduler, on n'a pas besoin de tous ces bits.



La tâche du bloc suivant, le contrôle automatique de gain (AGC), est donc de surveiller en permanence la force du signal et de le graduer pour en sortir des flux de seulement 10 bits à démoduler. Le bloc AGC fournit aussi l'indication de la force du signal reçu (RSSI).

Vient ensuite le démodulateur. Démoduler un signal AM, c'est en extraire l'amplitude en utilisant le théorème de Pythagore pour deux vecteurs orthogonaux : $\sqrt{I^2 + Q^2}$, cf. **figure 5**. Pour démoduler la SSB (et le morse), l'opération est hélas un peu plus compliquée. On a eu recours à la méthode de Weaver [2] [3]. L'idée de base (**fig. 6**) est de séparer la largeur de bande du signal en deux et de produire localement un signal en quadrature à basse fréquence pour le mélanger de nouveau au milieu de la bande. Choisir entre la bande latérale supérieure (USB) ou la bande inférieure (LSB) n'est plus que l'affaire d'un signe lors de l'addition. Le signal audio démodulé passe enfin par un filtre passe-bas pour retirer toutes les composantes HF de distorsion dans l'AGC ou le démodulateur. Il est alors ré-échantillonné pour correspondre au taux voulu par le CODEC audio et y est envoyé par interface série standard PCM. Le CODEC audio programmable peut amplifier et attaquer directement un HP ou un écouteur.

Les signaux de l'oscillateur local (LO) à la sortie des deux CN/A sont produits par deux blocs de synthèse numérique directe (DDS) dans le FPGA. Ce genre de synthétiseur consiste en un registre à accumulation de phase dont la valeur augmente d'une quantité fixe à chaque coup de l'horloge à 120 MHz du CN/A, laquelle provient d'une boucle à phase asservie (PLL) du FPGA ; l'augmentation détermine donc la fréquence. La valeur de l'accumulateur sert d'adresse



LISTE DES COMPOSANTS

Résistances :

Sauf indication contraire : 1 %, 100 mW, 0603

R1, R2, R3, R4 = 560 Ω
 R5, R6, R25, R28, R30, R34, R35, R36, R44 à
 R47, R48, R53 = 10 k Ω
 R7, R32, R33 = 330 Ω
 R8 à R11, R19 à R22, R37, R38, R39, R41 = 1 k Ω
 R12, R13 = 18 Ω
 R14, R17, R18 = 10 Ω
 R15, R49 à R52, R54 à R58 = 100 Ω
 R16 = 160 k Ω
 R23 = 47 k Ω
 R24, R29, R31 = 27 Ω
 R26 = 100 k Ω
 R27, R42, R43 = non installées
 R40 = 2 k Ω
 P1 = potentiomètre ajust. 10 k Ω 4,5 mm, CMS

Condensateurs :

Sauf indication contraire : 5 %, 0603

C1, C2, C5 à C8, C14, C16 à C19, C20, C22,
 C23, C25, C27, C28, C29, C42, C44, C46,
 C47, C48, C50, C52, C53 à C64, C66, C68,
 C70, C72 à C82, C92, C95, C98, C105,
 C106 = 100 nF
 C3, C4, C33, C34, C35 = non installés
 C9 = 5,6 pF $\pm 0,25$ pF
 C10, C11 = 27 pF, 1 %
 C12, C21 = 10 nF
 C13, C43, C89, C90, C102, C103, C104, C107
 à C110 = 47 μ F 16 V, tantale, 0,35 Ω , CMS
 boîtier C
 C15 = 33 pF
 C24, C26 = 10 μ F 6,3 V, 0805
 C30, C31, C32 = 10 pF
 C36 = 1 nF
 C37, C38 = 22 nF
 C39, C40, C41, C93, C96, C99 = 1 μ F, X7R
 C45, C49, C51 = 100 pF
 C83, C85, C86, C88 = 120 pF
 C84, C87 = 180 pF
 C65, C67, C69, C71, C91, C94, C97, C100,
 C101 = 2,2 μ F, X7R

Inductances :

L1 à L4, L6 à L25, L30 à L40 = 1 k Ω @
 100 MHz, 200 mA, 0603
 L5 = 15 nH, 5 %, 170 m Ω , 700 mA, f_{res.} 4 GHz
 L26 à L29 = 180 nH, 2 %, 640 m Ω , 400 mA,
 0805
 TR1, TR2 = TC4-1WG2+

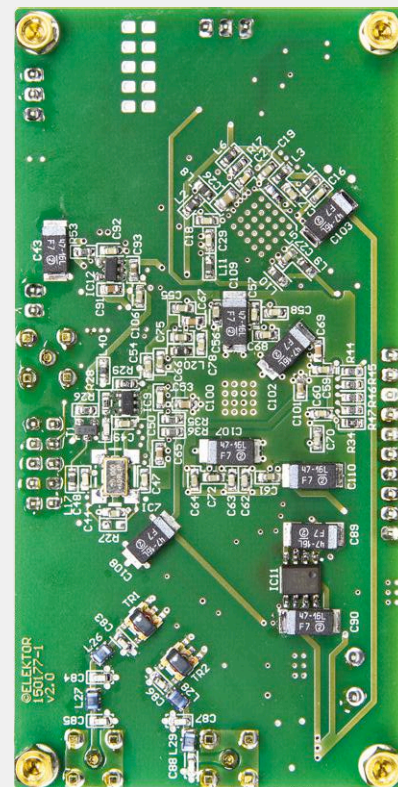
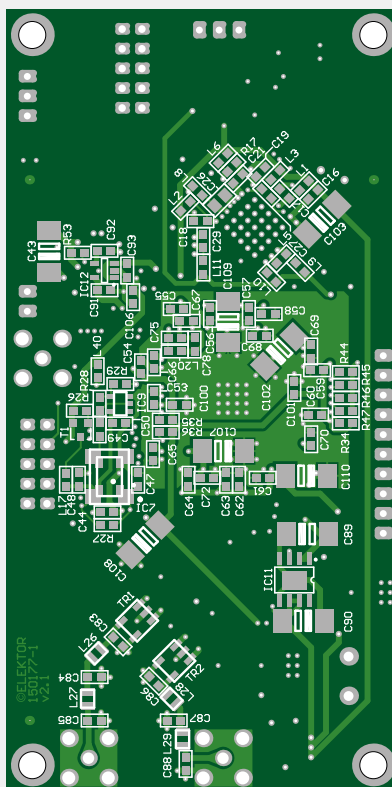
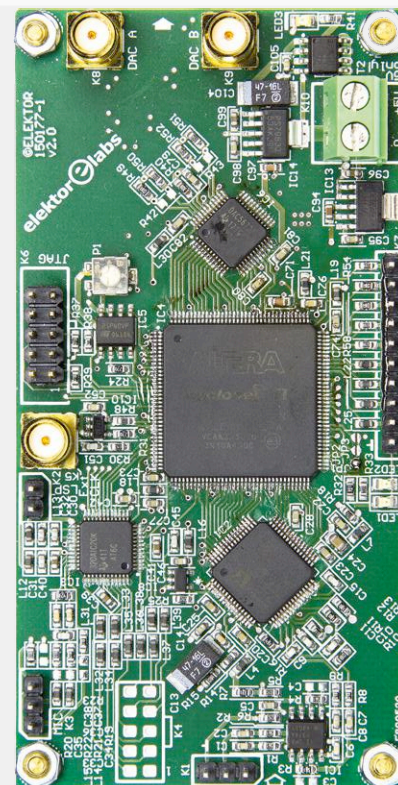
Semi-conducteurs

IC1 = OPA2374AIDG4
 IC2 = AD7760BSVZ
 IC3 = TLV320AIC20KIPFB
 IC4 = EP4CE10E22C8N
 IC5 = M25P40-VMN6PB
 IC6 = DAC5672IPFB
 IC7 = 20 MHz oscillateur à quartz, ajustable,
 5 \times 3,2 mm
 IC8, IC9, IC10 = 74AHC1GU04W5-7
 IC11 = LP3891EMR-1.2/NOPB

IC12 = TPS73018DBVT
 IC13 = TPS79625DCQG4
 IC14 = TPS79633DCQ
 LED1, LED2, LED3 = verte 0805
 T1 = PMV65XP
 T2 = IRF9321PBF

Divers

K1, K3 = embase à 3 picots, au pas de 2,54 mm
 K2 = embase à 2 picots, au pas de 2,54 mm



K4 = embase à 14 picots, au pas de 2,54 mm
 K5, K8, K9 = prise pour jack droit SMA, 50 Ω
 K6 = embase à 10 picots (2 \times 5),
 au pas de 2,54 mm
 K7 = embase à 10 picots, au pas de 2,54 mm
 K10 = embase borne à 2 vis encartable,
 au pas de 5 mm
 circuit imprimé réf. 150177-1



Figure 8. Le prototype construit par l'auteur, il opère en USB à 144 MHz.

Liens

- [1] www.elektormagazine.fr/150177
- [2] Donald K. Weaver Jr., « A Third Method of Generation and Detection of Single-Sideband Signals », www.h4.dion.ne.jp/~ja5fp/weaver.pdf
- [3] Daniel Uppström, « Weavers metod för SSB », ESR Resonans 2/2014, http://resonans.esr.se/ESR_Resonans_2014_2.pdf (en suédois)
- [4] www.elektormagazine.fr/labs/fpga-dsp-radio-for-narrow-band-communications-150177-i
- [5] <https://github.com/ast/dspdsdr>

dans une table de sinus à consulter et cette lecture est transmise au CN/A (fig. 7). Le mot de fréquence pour le CN/A A qui forme l'oscillateur local accordable a une largeur de 25 bits. Cela correspond à un pas de fréquence de $120 \times 10^6 / 2^{25} = 3,57$ Hz. Il y a en outre un réglage fin qui travaille sur des valeurs fractionnaires pour une résolution finale de 0,44 Hz.

Pour le second LO, qui est à fréquence fixe pour la conversion entre la première FI et la seconde, le DDS est réalisé à plus basse résolution pour préserver les ressources du FPGA, sa précision est d'environ 38 Hz.

Programmation du FPGA

Le micrologiciel compilé pour ce projet est disponible dans un fichier de configuration JTAG indirecte (*trx.jic*) qui configure le FPGA en passerelle pour ensuite charger la mémoire de configuration par le truchement du FPGA. On le fait à l'aide d'un câble de téléchargement USB-Blaster™ de connexion à un PC par USB pour former une interface JTAG pour la carte. Ce genre d'adaptateur ne coûte pas cher sur les sites commerciaux populaires.

Le logiciel à installer sur le PC s'appelle

Quartus Lite, à télécharger gratuitement du site d'Intel/Altera. Si l'EDI complet ne vous intéresse pas, n'en prenez que les applications nécessaires pour la programmation seule. La suite Quartus est disponible tant pour Windows que pour Linux.

Interface de commande

À l'allumage, le micrologiciel passe de la mémoire de configuration vers le FPGA, qu'il faut encore initialiser et commander par I²C ou port sériel. Le FPGA ressemble à une puce ordinaire dont on doit régler quelques registres. Tout transit vers le FPGA compte cinq octets : deux pour donner l'adresse du registre destinataire, les 38 bits restants sont les données de configuration.

On peut lire quelques signaux d'état ainsi que la valeur de force du signal reçu (RSSI), mesurée plusieurs fois par seconde pour en être informé.

Dans les pages [1] [4] et [5] du projet, vous trouverez une description détaillée de cette interface avec toute la cartographie des registres, le micrologiciel et son code source dans un dépôt GIT.

Une radio indépendante, on la commande d'ordinaire par une carte à µC avec afficheur et boutons, comme Platino d'Elektor. Un prochain article présentera une

carte adéquate. Pour des essais ou si l'on préfère un système informatique, un Raspberry Pi peut faire l'affaire. Pour RPi ou tout ordinateur sous Linux, il existe une applet en Python basée sur la suite graphique GTK qui rend la commande facile. Sinon, l'applet se connecte à la carte FPGA par le port sériel. Il y a aussi un mode d'identification expérimental qui permet la radiocommande. L'idée est de mettre en œuvre un serveur sur un RPi à proximité de la radio et de le connecter au RPi par l'applet depuis un ordinateur distant. Avec le pullulement de parasites d'origine humaine dans les centres urbains et les limitations et règlements qui y empêchent l'usage de grandes antennes, la télécommande intéresse de nombreux radioamateurs et radioécouteurs d'ondes courtes (SWL). Il devrait aussi être possible d'amener l'audio du FPGA directement sur les ports PCM/I²S du RPi au lieu de passer par l'audio analogique en opération à distance, reste à en vérifier la faisabilité. Un document joint au code source explique comment télécharger l'applet, installer le nécessaire et brancher le RPi à la carte FPGA.

Conclusion

Ce projet est un sérieux point de départ pour les radioamateurs et enthousiastes qui veulent construire leur radio sans compromis sur les prestations. Nous espérons aussi qu'il incitera des adeptes du numérique à découvrir le monde de la radio et du traitement du signal.

Il laisse place à de nombreux perfectionnements et compléments. S'il suscite l'intérêt auprès des lecteurs, nous proposerons d'autres cartes radio évoluées. Le code VHDL pour le FPGA à source ouverte est disponible sur [5], toute idée d'ajout ou de perfectionnement est la bienvenue.

Finalement, la carte est assez généraliste pour servir à d'autres projets centrés sur l'utilisation des FPGA en traitement numérique du signal. ◀

(150177 – version française : Robert Grignard)

DANS L'É-CHOPPE

→ 150177-1 : circuit imprimé nu pour carte DSP à FPGA

→ 150177-91 : carte DSP à FPGA prête à l'emploi

optocoupleurs

drôle de composant n°31

Neil Gruending (Canada)

Parfois il est nécessaire d'isoler des circuits les uns des autres, pour la sécurité, pour réduire le bruit ou même simplifier une fonction. Un exemple classique est la photorésistance que Gibson et Fender utilisaient pour ajouter des effets de trémolo dans les premiers amplis pour guitare. Cette photorésistance était une forme primitive de coupleur optique, ou optocoupleur, qui faisait varier la résistance d'une cellule au sulfure de cadmium (CdS) avec une source de lumière pour moduler la polarisation de l'amplificateur et créer des effets de trémolo. Ces optocoupleurs étaient faciles à fabri-

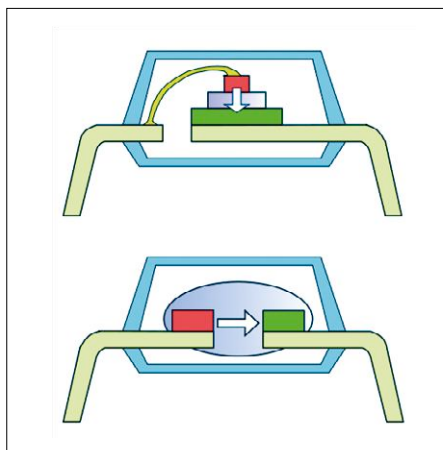
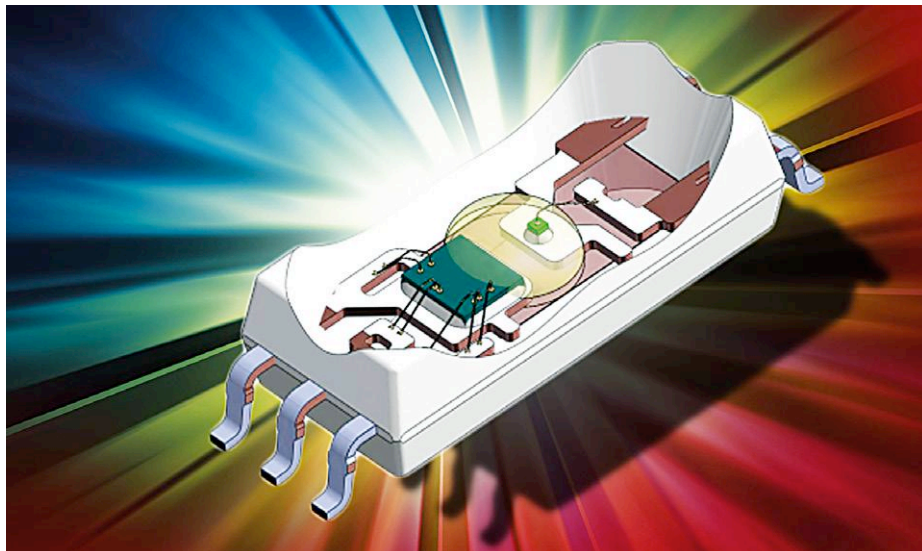


Figure 1. Structures d'optocoupleur [1].

quer à partir de composants discrets et des entreprises comme VacTec ont fini par les intégrer en un unique composant dans les années soixante.

En général, un optocoupleur utilise de la lumière pour relier des circuits isolés l'un de l'autre. Cette zone d'isolation permet au circuit de supporter des surtensions élevées (kV) et du bruit qui, normalement, détruiraient les composants sensibles. Mais isoler deux parties d'un circuit n'est pas très utile si l'on ne peut pas faire passer de signal entre les deux ; c'est pourquoi les optocoupleurs comportent d'un côté une source de lumière et de l'autre un détecteur (**fig. 1**). Cette structure fonctionne très bien pour les

petits signaux, mais n'est pas pratique pour transmettre de l'énergie comme avec un transformateur d'isolation. La zone d'isolation implique aussi que les deux côtés du circuit seront toujours isolés, même si l'optocoupleur cesse de fonctionner. Cela rend les optocoupleurs idéaux pour protéger les circuits.

L'étendue de la zone entre la source lumineuse et le récepteur définit la tension d'isolation : plus la zone est grande, plus cette tension est élevée. Les composants qui n'ont besoin que de quelques kV d'isolation ont typiquement une structure planaire, comme le composant du haut de la figure 1. Le composant du bas utilise un dôme de silicone où la lumière circule horizontalement pour permettre la création de zones d'isolation plus étendues et donc pour atteindre des tensions d'isolation plus élevées.

Les optocoupleurs utilisaient à l'origine des lampes incandescentes ou au néon comme source de lumière, mais ils ont rapidement évolué dans les années 70 quand les LED sont apparues. Les LED constituent une grande avancée sur les lampes au néon : meilleures linéarité et

rapidité, moins sensibles aux changements de température. Les détecteurs ont également changé, des cellules CdS aux photodiodes et phototransistors. Les photodiodes sont utilisées pour les interfaces logiques à haute vitesse. Les phototransistors sont plus lents et leur courant de sortie est lié à celui de la LED. Une photorésistance est un exemple d'optocoupleur linéaire : une variation du courant dans la lampe (émetteur) entraîne un changement identique de la résistance de la cellule CdS (récepteur). Les optocoupleurs numériques, quant à eux, sont faits pour transmettre des signaux binaires et donc optimisés pour la vitesse. En fait, certains optocoupleurs, comme ceux de la famille HP 6N137/HPCL2601 comportent même un circuit additionnel du côté du détecteur pour augmenter encore un peu plus leur vitesse. Les optocoupleurs sont utilisés depuis longtemps et toujours utiles aujourd'hui, même avec l'apparition des isolateurs à couplage magnétique et capacitif. J'espère vous avoir donné un bon aperçu de ces humbles composants. ◀

(160136 – version française : Kévin Petit)

Lien

[1] https://upload.wikimedia.org/wikipedia/commons/thumb/1/10/Optoisolator_topologies_both.svg/220px-Optoisolator_topologies_both.svg.png

MicroPython et PyBoard

de la LED qui clignote ... au serveur web qui fait clignoter une LED

Clemens Valens (labo d'Elektor)



Depuis la première édition de 1991, le langage de programmation Python s'est largement répandu. Il est devenu l'un des langages préférés de nombreux programmeurs. Pensé au départ pour les applications sur PC, Python a aussi été porté sur les systèmes embarqués où il s'appelle MicroPython (uPy) ; c'est un sous-ensemble de Python qui s'exécute sur des plateformes comme la carte BBC micro:bit et l'ESP8266. Il tourne également sur la PyBoard, la carte de démonstration uPy officielle.

Le projet Micro Python (initialement écrit en deux mots), pyboard comprise, a été lancé sur Kickstarter fin 2013 et il a recueilli plus de six fois le montant initial prévu de 15 000 £. La promesse de prendre en charge les modules CC3000 (Wi-Fi), WIZ820io (Ethernet) et NRF24L01+ (sans fil à faible consommation) n'y est sans doute pas pour rien. Quatre ans plus tard, voyons

ce qu'il en est sorti. Jouons un peu avec la pyboard et uPy.

Prise en main de la pyboard

Vous aurez besoin d'une pyboard v1.1 (PYBv1.1, joliment emballée !), d'un câble micro-USB et d'un ordinateur (j'ai utilisé un portable sous Windows 10). En raccordant la carte à l'ordinateur, une fenêtre s'ouvre immédiatement et affiche

le contenu du lecteur détecté et baptisé PYBFLASH. Le système d'exploitation devrait également détecter un port série.

Après avoir ouvert un programme pour terminal série (Tera Term), il suffit de le connecter au port de la carte pyboard et d'appuyer sur une touche pour voir l'invite de Python apparaître (**fig. 1**):

```
MicroPython v1.8.2 on 2016-07-13;  
PYBv1.1 with STM32F405RG  
Type "help()" for more  
information.  
>>>  
>>>
```

Bravo, rapide comme l'éclair ! Vous pouvez commencer à travailler moins d'une minute après avoir connecté la carte (une minute essentiellement passée

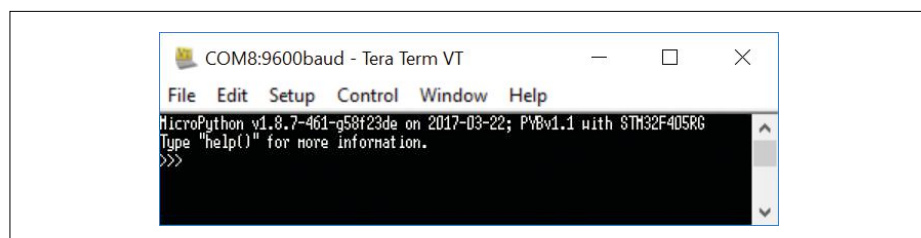


Figure 1. La REPL.

à promener la souris pour ouvrir les menus de l'OS et y naviguer).

REPL

Vous êtes maintenant confronté à un environnement de programmation interactif simple, capable de saisir des expressions individuelles, de les évaluer et d'afficher le résultat. Il s'agit d'une boucle Lire-Évaluer-Imprimer ou REPL (*Read-Evaluate-Print loop*). Il est important de connaître ce sigle, car avec Python, vous le rencontrerez souvent. Ceux qui ont programmé en BASIC seront familiers avec ce type d'interface même si, à l'époque, elle ne s'appelait pas REPL. Python est, comme le BASIC, un langage de programmation interprété et, comme le BASIC, il peut aussi être compilé pour tourner plus vite. En fait, Python ressemble beaucoup au BASIC.

Étape suivante : faire clignoter une LED

Comme l'invite le suggère, tapez « `help()` » et appuyez sur la touche Entrée. Une liste de commandes possibles s'affiche : vous pouvez démarrer. En regardant cette liste de plus près, vous devinerez rapidement de quoi la carte est capable. Construite autour du µC STM32F405RG ARM Cortex-M4F, la carte dispose d'une horloge en temps réel (RTC), d'un convertisseur analogique/numérique (CA/N), d'un convertisseur numérique/analogique (CN/A), de quatre LED, d'interrupteurs, d'un accéléromètre, d'un générateur de nombres aléatoires (RNG), des interfaces I²C, SPI, UART, de lignes d'E/S et elle peut piloter des servomoteurs. Comme certaines fonctions de commande de ces périphériques sont également listées, l'aide s'avère relativement complète. Il suffit de regarder la carte pour repérer le connecteur de carte SD.

À l'invite, tapez la commande ci-dessous et appuyez sur <Entrée>.

```
>>> pyb.LED(1).on()
```

Remarquez qu'une LED rouge à côté du poussoir USR s'illumine. Répétez la commande en tapant « 2 » au lieu de « 1 » (utilisez les flèches pour retourner à la commande précédente et l'éditer), une LED verte s'allume juste à côté de la rouge. La 3^e LED est jaune, la 4^e bleue. À l'exception de la jaune, les LED sont extrêmement lumineuses et mieux vaut ne pas les regarder en face. Continuons

en allumant la LED(3) ; à titre d'exercice, je vous laisse le soin d'éteindre les LED. Pour faire clignoter une LED indéfiniment, il faut une boucle, par ex. *while*. Pour que la boucle ne s'arrête pas, utilisez une condition qui sera toujours vraie, par ex. la constante booléenne VRAI (*TRUE*). Saisissez la commande (n'oubliez pas le « : » en fin de ligne)

```
>>> while True:
```

Ensuite, tapez une commande LED. Le terminal indente automatiquement le texte conformément aux règles Python (trois points). Entrez une commande de délai pour fixer la fréquence de clignotement, 250 ms, par exemple. Le programme ressemble à ceci :


```
>>> while True:
...     pyb.LED(3).toggle()
...     pyb.delay(250)
... 
```


Pour que cette boucle fonctionne, arrêtez le mode indentation en appuyant sur les touches Retour arrière, puis Entrée. La LED3 doit alors clignoter à la fréquence définie. Comme vous l'avez peut-être remarqué, le terminal n'accepte pas les commandes pendant l'exécution de la


Caractéristiques


- Programmation en MicroPython
- Serveur web simple
- 100% sans Arduino


INFOS SUR LE PROJET


 Python


 MicroPython


 pyboard


 serveur de web


 débutant

 connaisseur

 expert

 2 h environ

 pyboard v1.1,
PC,
terminal série

 Env. 35 €

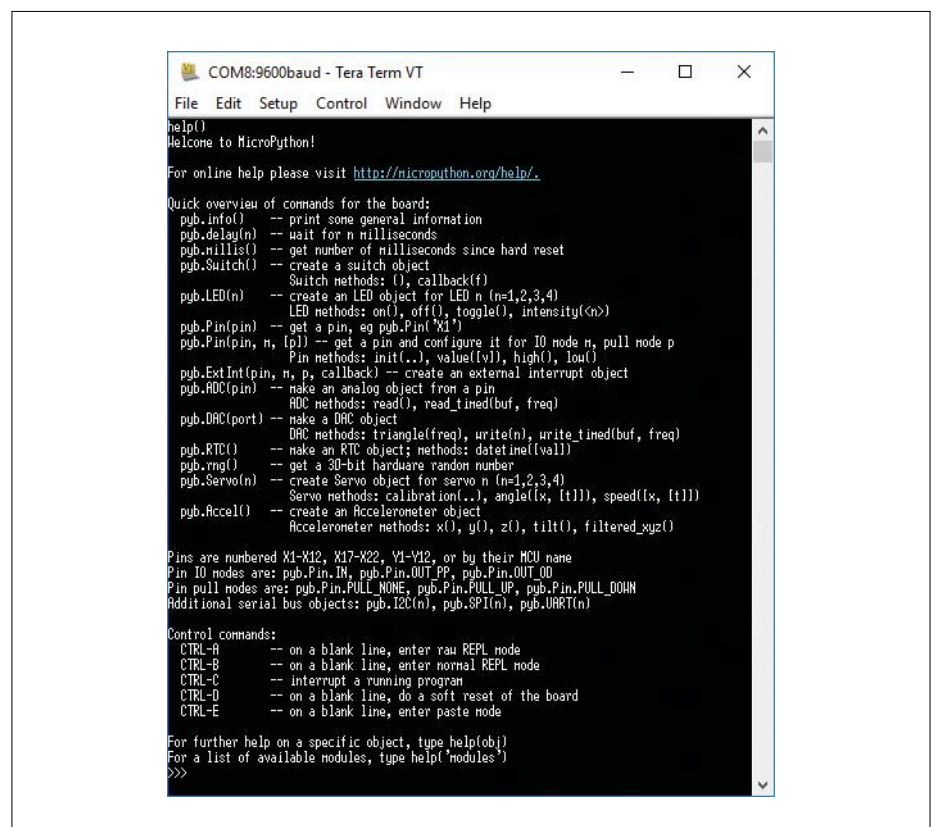


Figure 2. MicroPython contient aussi une aide en ligne complète.

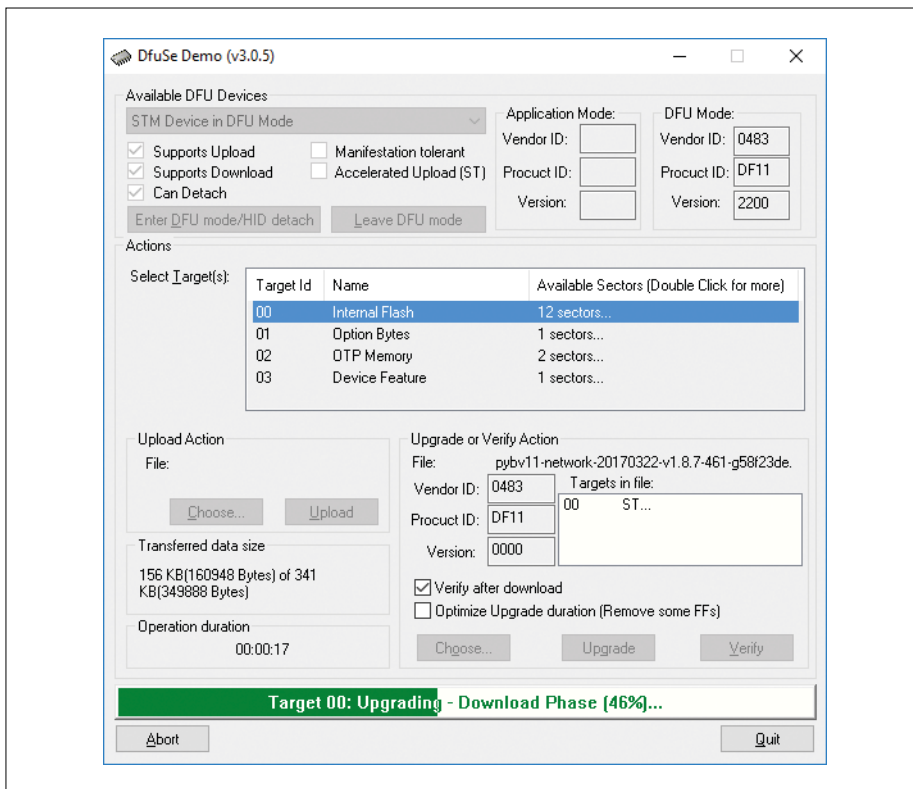


Figure 3. DfuSe en plein chargement d'un nouveau micrologiciel vers la pyboard.

boucle. Pour arrêter la boucle et retourner à l'invite, appuyez sur Ctrl-C (« interruption du programme en cours d'exécution »). Dès lors, toutes les commandes saisies pendant l'exécution de la boucle vont à leur tour s'exécuter. Mieux vaut être prudent et ne pas entrer n'importe quoi !

Accéléromètre embarqué sur la pyboard

Voici comment savoir rapidement si l'accéléromètre embarqué fonctionne correctement.

```
>>> while True:
...     print(pyb.Accel().x()),pyb.
Accel().y(),pyb.Accel().z())
```

La sortie devrait être une liste sans fin de trois valeurs (axes x, y et z). Secouez la carte dans tous les sens et vous verrez ces valeurs changer :

```
-1 -2 21
-1 -2 20
-2 -1 20
...
```

Comme précédemment, faites Ctrl-C pour retourner à l'invite. Ce test est intéressant, car il montre la vitesse d'exécution d'une boucle apparemment simple (lecture d'un capteur et impression

de trois entiers). J'ai obtenu trois lignes par seconde, ce n'est pas très rapide ! Le programme n'est pas traduit en langage machine (compilé). Il est interprété au fur et à mesure, cela explique cette lenteur, car même s'il n'y a qu'une instruction (en omettant le *while*), elle est en réalité très complexe. Il est possible d'améliorer les choses en créant des « modules figés (*frozen modules*) », mais cela sous-entend la compilation complète du microprogramme de la pyboard, ce qui sort du cadre de cet article.

Mon premier programme

Après vous être amusé avec REPL, vous vous demanderez sans doute comment faire pour ne pas avoir à retaper vos instructions à chaque fois. C'est assez facile. Tout ce qu'il vous faut c'est un éditeur de texte.

La pyboard est accessible comme lecteur de carte flash. Avec un explorateur de fichiers, vous verrez le fichier *main.py*. C'est le fichier Python qui s'exécute à la réinitialisation de la pyboard (matérielle à l'aide du poussoir RST ou logicielle par la commande Ctrl-D). Ouvrez ce fichier à l'aide d'un éditeur de texte et ajoutez-y vos commandes. Lorsque c'est fait, enregistrez le fichier et réinitialisez la carte pour lancer votre programme.

Comment sortir d'une impasse

Soyez créatif ! Essayez et corrigez vos erreurs. Si vous arrivez à déstabiliser le système au point que rien ne semble plus fonctionner, il y a deux procédures de secours : le mode protégé et l'initialisation avec les paramètres d'usine. En mode protégé (*safe mode*), les fichiers *boot.py* et *main.py* ne sont pas exécutés, ce qui vous donne accès au système de fichiers (le lecteur USB devrait apparaître). Vous pouvez alors modifier *boot.py* et *main.py* pour corriger les problèmes. Pour passer en mode protégé, maintenez le poussoir USB enfoncé tout en actionnant brièvement RST. Regardez maintenant les LED ; quand seule la jaune est allumée, relâchez USB : la carte est maintenant en mode protégé.

L'initialisation avec les paramètres par défaut (*factory reset*) va un peu plus loin. Elle supprime tous les fichiers de l'espace de stockage interne de la pyboard (et non pas de la carte SD), et restaure l'état d'origine des fichiers *boot.py*, *main.py*, *README.txt* et *pybcdc.inf*. Pour réinitialiser la carte, utilisez la même procédure que pour démarrer en mode protégé, mais quand les LED changent d'état, relâchez USB quand la jaune et la verte sont allumées. La LED jaune clignote trois fois et vous êtes de retour au point de départ : le début de cet article.

Si ces deux méthodes échouent, il y a un dernier recours : reprogrammer le micrologiciel. Pour cela, placez la carte en mode « mise à jour du micrologiciel » (*Device Firmware Update*, DFU). Il faut relier la broche P1 au 3,3 V avec un trombone par ex. avant de réinitialiser la carte avec le poussoir RST. Pour la première mise à jour, vous devrez peut-être installer un pilote DFU. Si Windows ne parvient pas à installer ce pilote, aidez-le un peu. Téléchargez l'archive STSW-STM32080 (ou DfuSe Demo) chez STMicroelectronics et installez-la. Pointez Windows sur <chemin d'installation>\STMicroelectronics\Software\DfuSe v3.0.5\Bin\Driver\ pour qu'il installe le pilote. Si tout s'est bien passé, DfuSe Demo devrait maintenant lister la carte (**fig. 3**).

Le site de uPy propose des micrologiciels précompilés. Attention : choisissez celui qui correspond à votre carte. Dans la zone *Upgrade or Verify Action* (pas *Upload Action*) cliquez sur *Choose*, naviguez jusqu'au fichier DFU à programmer puis

cliquez sur *Upgrade* (pas *Upload*). Fermez DfuSe, retirez le trombone et appuyez sur RST. Vous devrez peut-être procéder à une réinitialisation à l'état d'usine pour initialiser le système de fichier sur la mémoire flash. Dans mon cas, j'ai dû laisser Windows réparer le lecteur de la pyboard après une mise à jour du micrologiciel.

Sur Linux ou Mac OS, prenez l'outil dfu-util ou pydfu.

Vous n'êtes pas le seul à faire des erreurs

Au début de l'apprentissage de l'écriture de programmes uPy, vous rencontrerez évidemment votre lot d'erreurs et de maladresses. L'une des plus communes et des plus frustrantes c'est l'oubli de la touche de tabulation. Parfois aussi, vous pensez l'avoir utilisée, mais en réalité ce n'était pas le cas. Rappelez-vous que Python nécessite une indentation et que le seul caractère accepté est la tabulation. De nombreux éditeurs les remplacent à votre insu par des espaces. C'est pourquoi il faut vous assurer que ce n'est pas le cas. Le logiciel libre Notepad++ est un bon éditeur qui conserve les tabulations et met en évidence la syntaxe Python.

En dehors des problèmes d'indentation, Python peut également présenter quelques problèmes de compatibilité. MicroPython a été porté depuis Python, mais le portage n'est pas parfait. Sur le site uPy, vous trouvez une liste des différences entre CPython (l'implémentation de référence de Python) et uPy. Donc si un jour vous rencontrez une erreur étrange sur un code supposé fonctionner, rappelez-vous de consulter cette liste.

Mise en place d'un serveur web simple

Comme la campagne Kickstarter avait promis la prise en charge du module WIZ810io et que j'en avais quelques-uns, j'ai décidé d'essayer de mettre en place un serveur web simple (**fig. 4**). C'est le moment où les choses se sont compliquées et où les nuages sont apparus. La première étape était de chercher un exemple clef-en-main sur Google ; sans succès. J'ai bien trouvé le code d'un serveur web en MicroPython [4], mais il ne fonctionnait pas sur ma pyboard. Avant que vous ne commenciez à faire de grands gestes : oui, j'ai bien reprogrammé la carte avec

un micrologiciel qui inclut les pilotes réseau ([pybv11-network-20170322-v1.8.7-461-g58f23de.dfu](#)).

J'ai copié/collé le code d'un serveur simple dans `main.py`, mais il ne fonctionnait pas. Comment déboguer ? J'imagine qu'on a le choix entre ajouter des messages de dépannage au programme ou l'exécuter ligne par ligne dans la REPL ; j'ai opté pour la deuxième solution. Cette technique, bien que fastidieuse, m'a rapidement montré que l'appel à la méthode `sendall` de mon objet `socket` était invalide (voir code de [4] pour voir de quoi je parle). Cette méthode n'existe tout simplement pas ! C'est plutôt étrange : la documentation pour socket sur le site MicroPython mentionne clairement cette méthode. Cela m'a amené à jeter un œil au code source du pilote réseau dans le dépôt GitHub de MicroPython où j'ai découvert, après pas mal de navigations, que le pilote WIZnet n'exporte pas la méthode `sendall`. Il lui manque également quelques autres méthodes socket standards ; certaines sont incontournables. Allez comprendre. Il y a quelqu'un chez MicroPython Corp. Inc. Ltd qui n'a pas testé tout ce qu'il

aurait dû. À sa décharge, le module WIZ810io à base de W5200 n'est pas le plus récent.

Bon, même si remplacer `sendall` par `send` fut facile, cela n'a pas fait fonctionner mon serveur web. Il m'a fallu un bon moment pour trouver le problème : le champ `content-length` n'était pas présent dans l'entête HTTP envoyé par mon serveur. Après l'avoir ajouté, ma page MicroPython s'affichait enfin dans mon navigateur (**listage 1**).

En insérant le code fonctionnel dans `main.py`, j'ai eu quelques surprises. Initialement, ça a bien fonctionné, mais alors que j'améliorais le code, quelque chose est arrivé et tout a tourné au vinaigre. Sans entrer dans les détails, le système de fichiers de ma pyboard était corrompu et le contenu de ma page devenu inintelligible. Aussi, il n'était plus possible d'accéder au fichier `main.py` dont la taille était maintenant 0. Après avoir remis en état la pyboard à l'aide des méthodes décrites un peu avant, j'ai pu réinstaller mon petit programmeur et faire tourner le serveur web à nouveau. Cette mésaventure m'a appris à expérimenter dans la REPL plutôt que d'enregistrer un fichier sur la pyboard.

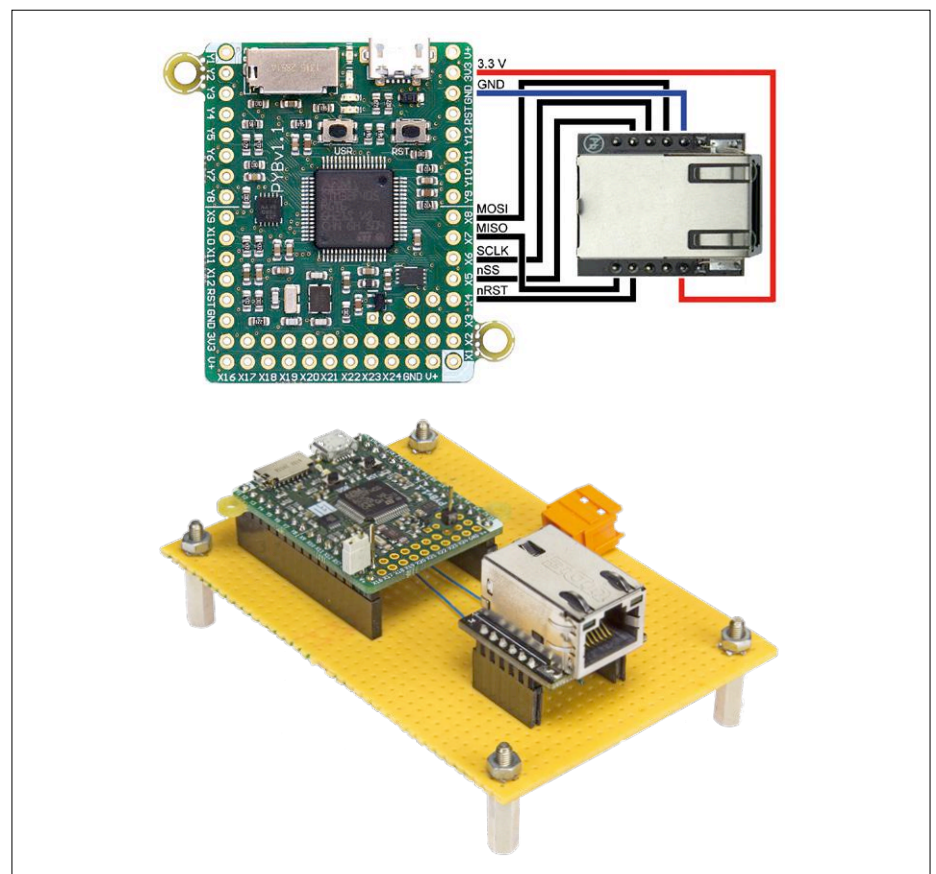


Figure 4. Serveur web MicroPython basé sur une carte pyboard PYBv1.1 et un module Ethernet WIZ810io de WIZnet.

Son système de fichiers est un tantinet fragile. Afin de pouvoir utiliser la REPL confortablement, il vaut mieux utiliser un terminal qui permet de coller un texte de plusieurs lignes, comme Tera Term. Vous pourrez coller des blocs de code entiers jusqu'au point où il faudra insérer un Retour arrière pour réduire l'indentation (par ex. à la fin d'une boucle `while` ou d'un `if-else`) ; vous gagnerez un peu de temps.

J'en veux plus !

Vous trouverez des tutoriels sur le site MicroPython pour apprendre à utiliser les périphériques de la pyboard. Leur lecture est chaudement recommandée, vous gagnerez du temps. Ils présentent également des techniques avancées : par ex. combiner assembleur et uPy. N'oubliez pas que ces tutoriels sont faits pour la pyboard et qu'il est fort possible qu'ils ne fonctionnent pas (de la même manière) avec d'autres cartes.

Conclusion

MicroPython et la pyboard sont un moyen distrayant et bon marché d'accéder à la programmation embarquée et d'apprendre progressivement le langage Python. Toutes les fonctions promises durant la campagne Kickstarter ont été développées, même les plus étendues. Le site uPy avec ses bons tutoriels n'est pas forcément mis à jour rapidement, en cas de doute, consultez les pages du projet sur GitHub. ◀

(160428 – version française : Kévin Petit)

Liens

- [1] www.elektormagazine.fr/160428
- [2] <http://micropython.org/>
- [3] <https://github.com/micropython>
- [4] <https://github.com/RinusW/WiPy/tree/master/AiCWebserver>

Listage 1. Un serveur web simple pour la pyboard.

```
import os
import network
import socket

# Adaptez ces paramètres à votre réseau
my_ip = '192.168.2.33'
subnet_mask = '255.255.255.0'
gateway = '192.168.2.1'
dns = '8.8.8.8' # c'est un serveur public recommandé par WIZNet
# La seule et unique page servie
page_name = 'page1.htm'

# Initialisation de l'interface réseau
nic = network.WIZNET5K(pyb.SPI(1), pyb.Pin.board.X5, pyb.Pin.board.X4)
nic.ifconfig((my_ip, subnet_mask, gateway, dns))
print(nic.ifconfig())
# Lancement du serveur (2x Ctrl-C vous ramènera à la REPL)
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.bind(('', 80))
s.listen(0)
while True:
    conn, addr = s.accept() # addr n'est pas utilisé mais nécessaire
    request = conn.recv(1024) # réception d'une requête HTTP
    # Recherche de la chaîne "Val="
    val_begin = str(request).find('Val=')
    if val_begin > 0:
        # On a trouvé "Val=", extrayons sa valeur
        pyb.LED(1).on()
        val_end = str(request).find(' ', val_begin)
        v = str(request)[val_begin+4:val_end]
        print("Val =", v)
        conn.send(v)
        # LED3 éteinte si v<50, allumée sinon
        if int(v) < 50:
            pyb.LED(3).off()
        else:
            pyb.LED(3).on()
    else:
        # Chaîne "Val=" pas trouvée, on renvoie la page web
        pyb.LED(2).on()
        # Envoi de l'entête HTTP
        conn.send('HTTP/1.1 200 OK\r\nConnection: close\r\nServer: pyboard\r\nContent-Type: text/html\r\n')
        conn.send('Content-Length: ')
        # Il faut envoyer la taille des données que nous allons envoyer
        conn.send(str(os.stat(page_name)[6]))
        # On envoie la taille du fichier
        # Fin de l'entête HTTP
        conn.send('\n\n')
        # Envoi de la page
        f = open(page_name, 'r')
        conn.send(f.read())
    conn.close()
    pyb.LED(1).off()
    pyb.LED(2).off()
```





(presque) tout ce que vous avez toujours voulu savoir sur...

les composants en astronautique

Réponses de **Jaime Estela** (Spectrum Aerospace Group)

En ces temps où de petites entreprises ou même de jeunes pousses peuvent s'offrir leur propre satellite, il vaut la peine de s'interroger sur les particularités de l'électronique en astronautique.

Q Les effets des rayonnements spatiaux endommagent-ils les composants électroniques ?

R Dans l'espace, tous les composants électroniques sont endommagés par rayonnement. Leur robustesse dépend de nombreux facteurs. Certains composants ne jouissent que d'une durée de vie brève dans l'espace, d'autres peuvent rester en service pour de nombreuses années. Les composants spéciaux pour usage militaire peuvent fonctionner très longtemps dans l'espace. Les composants du commerce se comportent différemment. Bien qu'ils ne soient pas conçus spécifiquement pour un usage spatial, de nombreux composants peuvent fonctionner des années dans l'espace avec des détériorations minimales.

Q À quels effets des rayonnements faut-il s'attendre dans l'espace ?

R **Décalages dans la structure cristalline (displacement damage)** : des particules (neutrons, protons, particules alpha, ions lourds) et des photons gamma à haute énergie déforment la grille cristalline et influencent les propriétés semi-conductrices.

Les rayonnements ionisants (dégradation rapide) : des charges sont induites et s'accumulent. Celles-ci augmentent les pertes du composant (courants parasites) jusqu'à son autodestruction.

Single Event Effect : des particules hautement ionisées induisent de fortes charges à court terme (impulsions parasites) dans le semi-conducteur et en altèrent le comportement, corrompant ainsi les données.

Q Quels tests des rayonnements existe-t-il ?

R Pratiquement il existe principalement deux types de tests des rayonnements :

Total Ionizing Dose Test (TID) : un comprimé de cobalt 60 est utilisé comme source gamma. Une telle source radioactive est connue en médecine, mais ici sa puissance est plus faible. Ce rayonnement permet de simuler en peu de temps de nombreuses années d'usage dans l'espace. Les charges induites influencent ainsi les propriétés semi-conductrices de diverses manières.

Single Event Effect Test (SEE) : on utilise un accélérateur de particules qui bombarde les composants avec des protons ou des ions lourds. Le résultat le plus sévère est l'altération

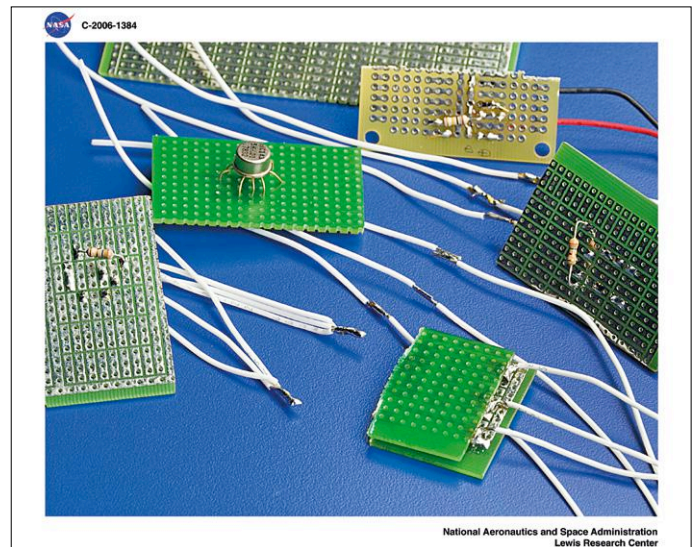


Fig. 1. Des astronautes de l'ISS lors de la réparation de composants électroniques (photo : NASA).

de données par une courte impulsion parasite induite dans le circuit. Cet effet est déterminant pour des circuits numériques.

Q Y a-t-il des différences selon l'orbite des satellites ?

R Oui, plus l'orbite est élevée, plus la dose de rayonnements le sera aussi. Les satellites en orbite terrestre basse (*Low Earth Orbit*, LEO) sont exposés à des doses de rayonnements faibles. À 10 km d'altitude, les passagers d'un avion reçoivent une dose de rayonnement supérieure, les heures de vol des équipages sont limitées pour ne pas mettre leur propre santé en danger. La station spatiale ISS vole à une altitude comprise entre 350 km et 450 km. Cette orbite a été choisie afin d'exposer les astronautes à une dose de radiation la plus faible possible.

Le champ magnétique terrestre représente une protection supplémentaire. Tant qu'un objet se situe à l'intérieur de ce champ, il est protégé des particules hautement chargées : celles-ci sont piégées par le champ magnétique et déviées vers les pôles. Les satellites de télécommunication gravitent en orbite géostationnaire (GEO). Avec cette trajectoire, un satellite semble toujours avoir la même position par rapport à la terre. L'orbite géostationnaire avec son altitude de près de 36000 km est clai-

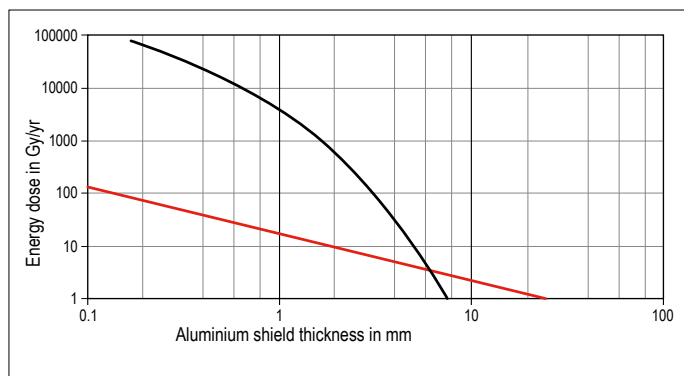


Fig. 2. La dose d'énergie agissant sur le système électronique est indiquée ici en fonction de l'épaisseur du blindage d'aluminium (en noir : faisceaux d'électrons, en rouge : rayonnement électromagnétique).

Source : Wikipédia, domaine public.

rement au-delà du champ magnétique terrestre. Cette orbite est plus contraignante pour l'électronique, car celle-ci doit fonctionner sous une dose de rayonnement supérieure. Les missions dans l'espace lointain sont encore plus délicates. La majorité de l'électronique est protégée au sein de l'infrastructure du satellite.

Q Quels genres de composants sont utilisés en astronautique ?

R Plus le vol est long et l'altitude élevée, plus il y a de mécanismes de protection à mettre en œuvre. Le projet doit répondre aux exigences, ni (nettement) plus, ni moins. Les composants **ITAR** (*International Traffic in Arms Regulations*) peuvent fonctionner sans problème 100 ans en LEO, mais l'usage de ce type de composants chers est exagéré si une mission ne doit durer que cinq ans par exemple. C'est pourquoi les composants **COTS** (*Commercial Off The Shelf*, composants commerciaux de série) sont de plus en plus utilisés. Les satellites GEO doivent fonctionner sans interruption pendant 15 ans dans un environnement nettement plus sévère. L'emploi de composants ITAR s'avère donc pertinent. Régulièrement, il arrive même que l'électronique de tels satellites continue allègrement de fonctionner après leur durée de service. Il existe aussi des fusées de recherche en altitude qui ne volent que quelques minutes (max. 15), on peut renoncer aux composants spéciaux, des composants COTS suffisent alors amplement.

Q Quelle mesure de protection est la plus efficace / la plus pertinente ?

R Presque tous les satellites sont construits avec des **redondances**, à l'exception des pico et nano-satellites où il n'y a pas assez de place pour dupliquer l'électronique. La redondance permet d'augmenter les durées de service. Au début de l'ère des vols spatiaux, tous les systèmes disposaient de plusieurs modules redondants. Un ordinateur de bord par ex. pouvait disposer de huit redondances. L'électronique doit fonctionner sans problème particulièrement pour les vols habités. Certes la redondance augmente la complexité, le besoin en énergie, le poids et le volume du vaisseau spatial, mais elle reste une des méthodes les plus cruciales pour augmenter la fiabilité d'un système de vol spatial.

Tolérance : il est toujours recommandé de concevoir l'électronique à l'aide de composants robustes. Il est même possible de tenir compte de la dégradation connue des composants dès la

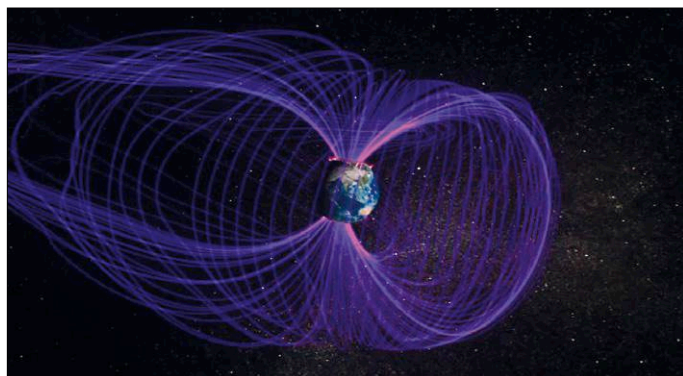


Fig 3. La première mission du satellite américain Explorer 1 a découvert en 1958 une ceinture de particules chargées électriquement aux environs de la Terre (dessin : NASA).

conception, de sorte que l'électronique reste toujours dans les limites attendues de fonctionnement. Mais il n'est pas possible, au vu d'une feuille de spécification, de déduire le comportement d'un composant du commerce en conditions spatiales. Même les fabricants l'ignorent. Ces valeurs sont pourtant très importantes pour toute utilisation dans l'espace, c'est pourquoi elles sont à déterminer expérimentalement.

Blindage : il est courant de protéger l'électronique sensible par la structure du satellite. La protection est d'autant meilleure que le blindage est épais, mais le volume de la structure, et pire encore le poids du satellite, sont alors d'autant plus élevés. Ces deux paramètres peuvent être à l'origine d'une augmentation des coûts initiaux. On peut aussi utiliser d'autres matériaux, mais la plupart sont plus lourds que l'aluminium. Les blindages spéciaux ne sont ainsi utilisés qu'en cas d'absolue nécessité.

Q Quels sont les composants particulièrement vulnérables, lesquels le sont moins ou pas du tout ?

R La différence la plus importante est celle entre composants passifs et actifs. Les composants tels que les résistances sont très robustes, les semi-conducteurs en revanche sont fortement dégradés par les rayonnements. La technologie bipolaire est en général plus robuste que la technologie CMOS. Mais cela dépend de nombreux facteurs tels que la géométrie des couches, les matériaux, l'encapsulation, etc. Les diodes, transistors, circuits intégrés, quartz, CCD et composants optoélectroniques sont sensibles aux **rayonnements ionisants**, alors que transistors FET, CCD et CMOS APS (*Active Pixel Sensor*), mais aussi les composants optoélectroniques et tous les circuits intégrés sont sensibles aux **Single Event Effects**.

Aujourd'hui, l'ESA et la NASA encouragent l'usage des FPGA, car le matériel programmable permet la télémaintenance. En outre les FPGA intègrent un grand nombre de fonctions sur un seul composant, l'électronique en devient plus compacte. Il existe aussi un processeur de l'ESA du nom de LEON, un cœur logiciel IP pouvant être intégré dans un FPGA [1].

Q Quelles peuvent être les conséquences des dommages / dégâts ?

R En cas de dégradation, l'électronique peut par ex. demander plus d'énergie. À partir d'une certaine dégradation, le composant est détruit.

Si le comportement du composant dans l'espace est connu, on peut en tenir compte lors de la conception. Un exemple : si un ampli a un gain de cinq, et à la fin de la mission ce gain se réduit à trois (évalué par simulation ou test de qualification), on en tiendra compte lors de la conception du circuit de sorte que sa fonction soit garantie durant toute sa durée d'utilisation. On peut ainsi compenser la dégradation du circuit.

Utiliser des composants en électronique spatiale dont le comportement serait inconnu revient à jouer à la roulette russe. La méthode la plus professionnelle, mais aussi la plus onéreuse, est la qualification des composants individuels. Il y a de nos jours des opérateurs de satellites qui souhaitent volontiers faire des économies de temps et d'argent, et font des tests très courts (parfois même de cartes entières). Si la carte tombe en panne, personne ne sait alors quel composant mettre en cause. La robustesse d'une carte est toujours définie par celle de son maillon le plus faible. Mais on a de bonnes chances de concevoir des cartes plus robustes grâce à une connaissance du comportement de chaque composant.

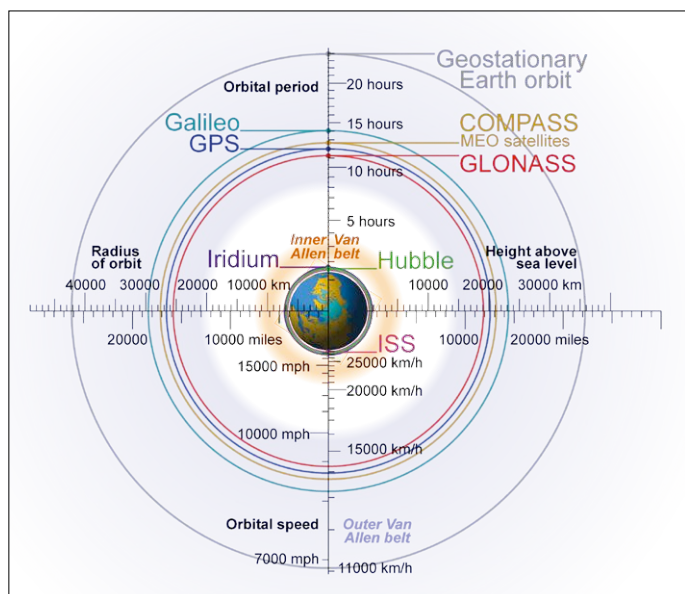


Fig. 4. Comparaison des orbites terrestres de divers satellites connus (dessin : Geo Swan, CC BY-SA 3.0).

Un test TID dure environ deux semaines et suit les recommandations de l'ESA. Parfois un test ne sera effectué que pendant six heures. Mais il en est comme de la cuisson de gâteau : si la recette indique « 200 °C pendant 30 min », on n'obtiendra pas le même résultat en cuisant le gâteau pendant 10 min à 600 °C. Les processus physiques demeurent incontournables !

(160299 – version française : Xavier Pfaff)

Littérature

« Electronics for Space Guide »,
Jaime Estela, bientôt disponible chez Elektor.

DÉCOUVRIR CRÉER PARTAGER

Maîtrisez les microcontrôleurs à l'aide d'Arduino



L'ambition de cet ouvrage est de vous faire entrer dans le monde Arduino puis de vous en faire ressortir victorieux pour vous emmener plus loin dans l'apprentissage de la programmation des microcontrôleurs. Il met la théorie en pratique sur une carte Arduino* avec l'environnement de programmation Arduino. Après ce parcours initiatique inédit, plaisant et ludique, vous programmerez vous-même n'importe quel microcontrôleur. Ce livre sera donc votre premier livre sur les microcontrôleurs avec une fin heureuse !

Tous les programmes présentés peuvent être téléchargés.

Points forts

- objectif double :
Théorie : apprentissage général de la programmation des microcontrôleurs
Pratique : montages sur carte Arduino avec l'environnement de développement Arduino
- réalisations très originales
- matériel peu coûteux ; logiciel gratuit, code source ouvert
- carte* d'expérimentation polyvalente créée par l'auteur
- **NOUVEAU** : carte AVR Playground*, mélange d'Arduino Uno et de carte de développement traditionnelle
- **NOUVEAU** : carte Elektor Uno R4* avec ATmega328PB et plus de périphériques que la carte Arduino Uno R3

* à commander séparément du livre

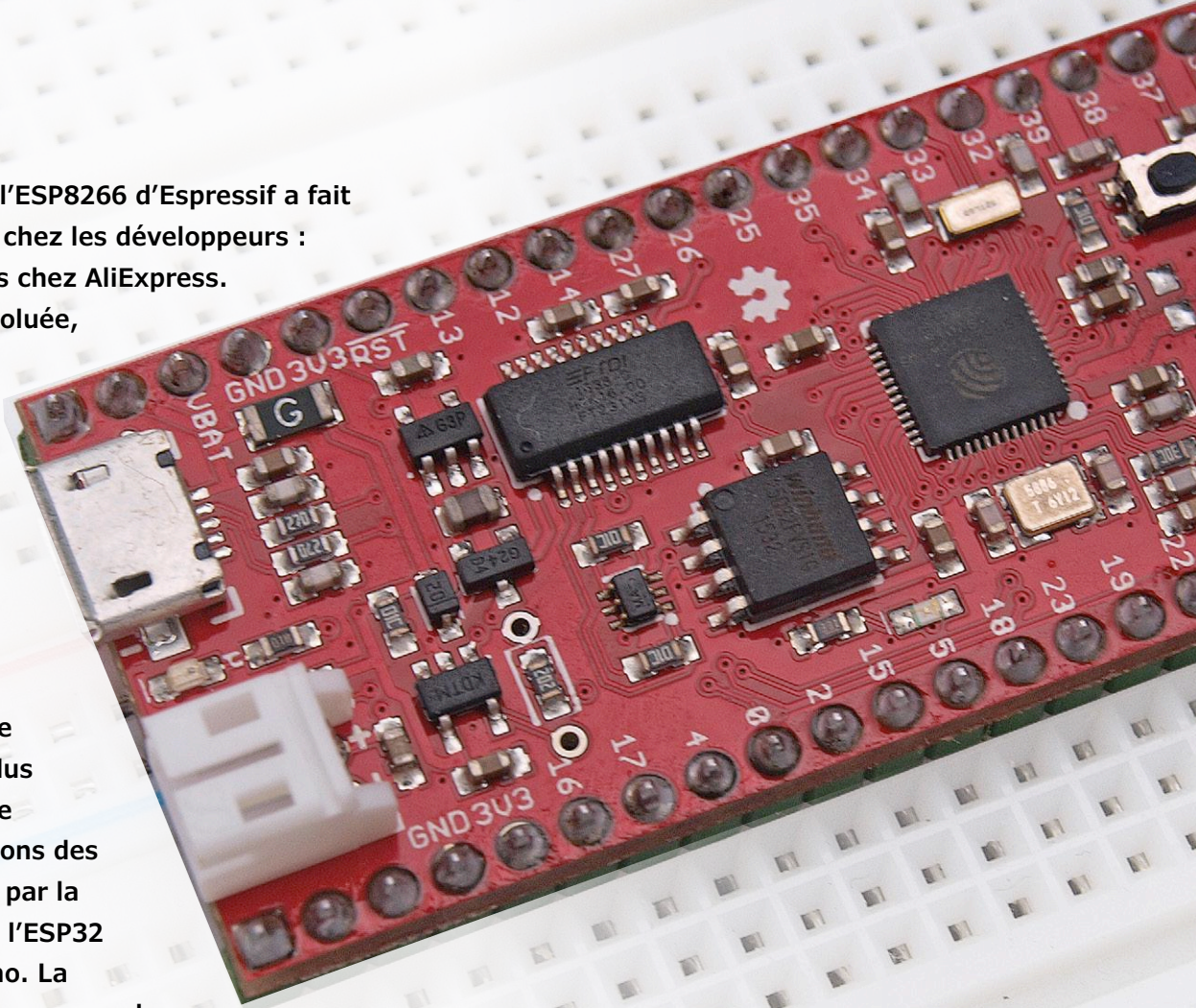
www.elektor.fr/maitrisez-les-microcontrolleurs-3

elektor

mutation de l'ESP8266

découvrons l'ESP32 avec l'EDI d'Arduino

Le prix minime de l'ESP8266 d'Espressif a fait grosse impression chez les développeurs : deux ou trois euros chez AliExpress. Avec sa version évoluée, l'ESP32, Espressif prépare ses systèmes sur puce (SoC) WLAN à l'avenir : en plus d'un processeur à double cœur, ce nouveau SoC est équipé d'un module Bluetooth, d'une plus grande mémoire de travail et d'extensions des E/S. Commençons par la programmation de l'ESP32 avec l'EDI d'Arduino. La prochaine fois, nous verrons les outils de développement ESP-IDF.



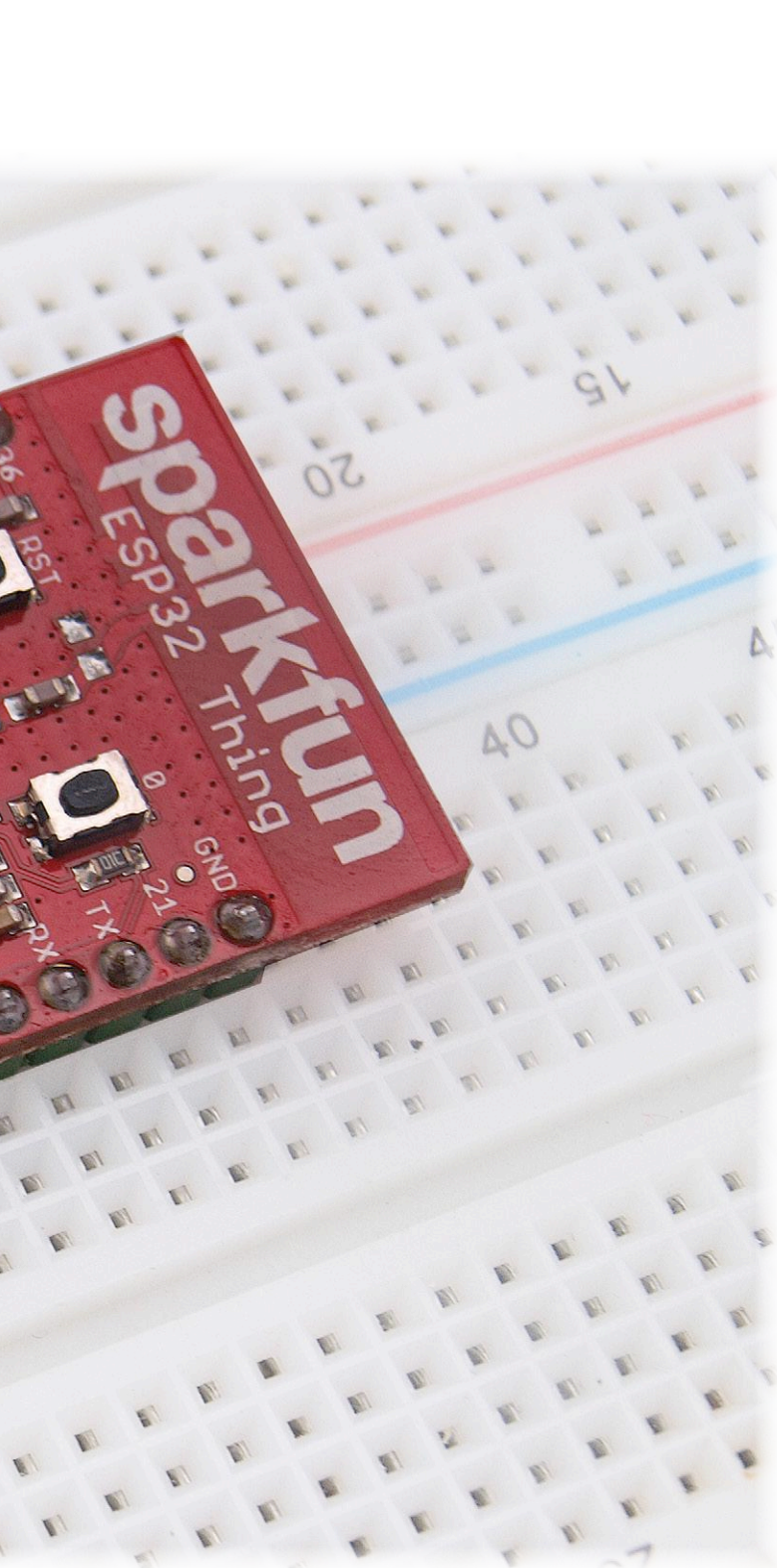
Tam Hanna (Slovaquie)

Après les premières livraisons d'ESP32, il y a eu de gros problèmes de livraison des cartes de liaison compactes (*breakout board*) ; les puces individuelles ont été peu affectées. Le faible écartement des broches pénalise le développement de ses propres cartes pour faire ses premiers pas. J'ai utilisé pour ce projet la carte ESP32 Thing de SparkFun que l'on peut se procurer auprès de Mouser [1]. SparkFun ne livre pas de connecteur mâle pour enficher la carte sur une plaque d'expérimentation ; il faut se procurer séparément les broches nécessaires. À côté de l'ESP32 Thing, on

trouve des douzaines d'autres cartes de différents fabricants. Espressif fournit sur [2] une liste de candidats potentiels qui font *grosso modo* la même chose, pour autant qu'on dispose d'une puce USB de FTDI.

Poste de travail à ESP32

La carte ESP32 Thing est aussi équipée d'une puce USB/série de FTDI. Ce peut être intéressant dans la mesure où l'on possède une carte Arduino ou similaire et que le pilote est déjà installé sur le PC. Pour cet article, je travaille sur le SE Windows 8.1 Entreprise avec l'EDI Arduino en version 1.8.2 installé. Au moment où j'ai écrit ces lignes, Espressif n'avait pas encore introduit son kit de développement logiciel (« core ») dans le



Erreurs agaçantes

Faites attention, les versions précédentes de l'ESP32 présentent différentes erreurs. Cela vaut la peine de lire la liste sous [9].

Pensez alors à ce que vous allez saisir dans le champ [Target Directory](#), par ex. `C:\Users\TAMHAN\Documents\Arduino\hardware\espressif\esp32` comme chez moi.

Cliquez alors sur le bouton [Clone](#) pour lancer le téléchargement du code depuis le dépôt GitHub. Selon l'occupation du service, cela peut prendre un certain temps. Il ne faut pas s'étonner de ce que la fenêtre Git ne réagisse pas de temps en temps. À la fin du travail, une fenêtre multicolore apparaît comme à la **figure 1**. Vous pouvez la fermer sans autre commentaire. Bizarrement, le SDK est fourni en deux blocs. De GitHub, il ne vient que les informations spécifiques à l'environnement d'Arduino. On se sert du programme `C:\Users\TAMHAN\Documents\Arduino\hardware\espressif\esp32\tools\get.exe` pour télécharger des composants ESP32 à ouvrir d'un double clic, de préférence avec l'explorateur Windows. Le programme ouvre une fenêtre de console qui vous informe sur l'état d'avancement du transfert. Quand il a réussi, on trouve un répertoire du nom de `C:\Users\TAMHAN\Documents\Arduino\hardware\espressif\esp32\tools\xtensa-esp32-elf`, qui contient l'environnement de développement ESP-IDF sur lequel nous reviendrons dans le prochain article.

Branchez alors l'ESP32 Thing avec un câble micro-USB sur le PC et lancez la recherche du pilote. Ne soyez pas surpris que l'assistant affiche parfois une souris : pour d'obscures raisons,

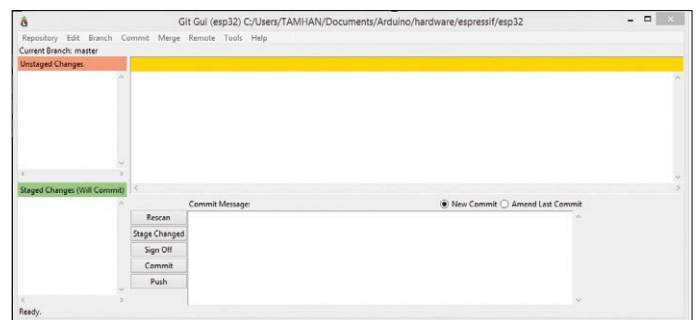


Figure 1. Le SDK Arduino pour l'ESP32 se charge par GitHub.

gestionnaire de carte de l'EDI Arduino. Pour charger les codes, il faut d'abord installer l'interface graphique de Git (disponible en [3]) ; ensuite dans la page de démarrage, saisissez [Git GUI](#). Cliquez alors sur [Clone Existing Repository](#) et, dans le champ [Source Location](#), saisissez `https://github.com/espressif/arduino-esp32.git`.

Comme répertoire cible, je me sers de `C:\Users\TAMHAN\Documents\Arduino`. Veillez à remplacer `TAMHAN` par votre nom d'utilisateur pour l'EDI Arduino sur votre ordinateur. Selon la configuration, il peut y avoir dans ce répertoire un ou plusieurs croquis ainsi qu'un dossier *hardware* dans lequel vous introduisez manuellement les kits de développement logiciel (SDK), mais si ce dossier est absent, ce n'est pas grave.

le pilote du FTDI est affublé de ce symbole. Une fois reconnu, vous pouvez lancer l'EDI Arduino ; sous [Outils](#) → [Carte](#), choisissez la carte *SparkFun ESP32 Thing*. Sur un petit écran, il se peut que vous deviez aller voir plus bas ; dans le menu de sélection, l'EDI Arduino présente toujours « ses » cartes avant celles des autres fabricants.

L'EDI Arduino ne reconnaît pas automatiquement l'ESP32 Thing. Pour atteindre le bon port COM, allez dans le [Gestionnaire de périphériques](#) où l'appareil apparaît comme port série (**figure 2**). Entrez ce port dans l'EDI Arduino pour terminer la configuration. Le débit binaire de chargement est pré-réglé sur 921 600 b/s et, au début, on peut le laisser ainsi. Les déve-

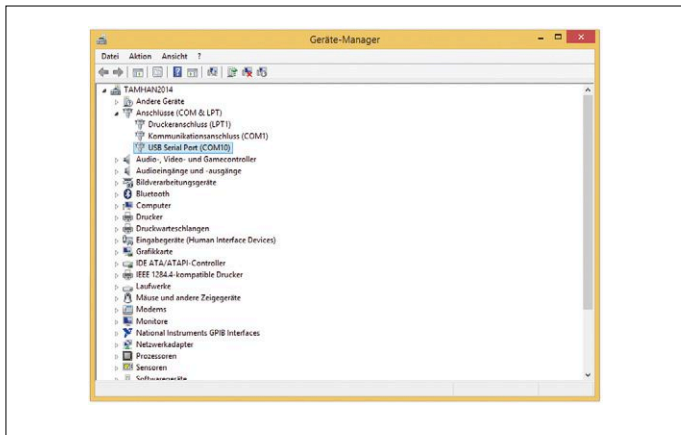


Figure 2. On trouve le numéro du port COM dans le gestionnaire de périphériques.

loppeurs chevronnés sur ESP32 le réduisent parfois, quand il s'agit de transférer une très grande quantité de données binaires pour éviter les erreurs. Cela entraîne un allongement du temps de cycle quand il faut reprogrammer, aussi, une prudence excessive est déconseillée.

Hello World !

Les concepteurs de cartes d'évaluation parent leurs réalisations d'une ou plusieurs LED. Dans le monde d'Arduino, la broche 13 sert à ça. SparkFun se distingue : là, c'est la broche 5.

Le **listage 1** reprend le croquis prévu pour effectuer un galop d'essai, le logiciel est disponible sur la page de cet article [4]. Il ne réalise pas d'exploit technique, il se satisfait d'allumer et d'éteindre périodiquement une LED. La mise en place dure entre 30 et 60 s avant que la LED bleue ne se mette à clignoter.

À quelle vitesse va-t-on ?

La tâche suivante consiste à mesurer le temps nécessaire pour que l'ESP32 commence à exécuter les commandes. C'est que dans les systèmes d'exploitation en temps réel, cette valeur est importante : l'Arduino 101 peut prendre un temps de réflexion de jusqu'à 8 s, ce qui dans les applications d'automatisation peut conduire à des situations dangereuses.

Pour déterminer les broches utiles à la mesure, commençons par l'examen de l'architecture du matériel. Dans l'ESP-IDF, on trouve la structure `gpio_num_t`, elle sert à adresser les E/S

d'usage général (GPIO). Le code – considérablement réduit dans le but d'améliorer la lisibilité – se présente comme suit :

```
typedef enum {
    GPIO_NUM_0 = 0,      /*!< GPIO0, input and output */
    GPIO_NUM_1 = 1,      /*!< GPIO1, input and output */
    . . .
    GPIO_NUM_39 = 39,    /*!< GPIO39, input mode only */
    GPIO_NUM_MAX = 40,
} gpio_num_t;
```

On peut déduire de sa linéarité de structure que l'ESP32 ne travaille pas avec des banques de registres GPIO : chaque broche est décrite de manière univoque par un identifiant propre. À ce stade, inutile d'approfondir les codes – la méthode qui suit incite à se servir directement des identifiants fournis :

```
extern void IRAM_ATTR __digitalWrite(uint8_t pin,
uint8_t val) {
    if(val) {
        if(pin < 32) {
            GPIO.out_w1ts = ((uint32_t)1 << pin);
        }
    }
    . . .
```

Sachant cela, on peut aller voir la fiche technique de l'ESP32 Thing, elle est à disposition sur [5] au format PDF et illustrée à la **figure 3**. Si vous utilisez une autre carte de développement, consultez plutôt la documentation de son fabricant.

Intéressons-nous aux lignes GPIO5 et VUSB. La première sert dans notre programme à piloter une LED, l'autre est reliée directement au port micro-USB et nous fournit des informations sur la présence ou non de tension. Comme nous utiliserons dans la suite un oscilloscope de marque LeCroy 9354AM à quatre canaux, nous pourrions surveiller aussi la broche 3V3 reliée à la sortie du régulateur de tension sur la carte.

Si vous venez du monde Raspberry Pi, soyez très vigilant avec la tension d'alimentation. L'ESP32 y est très sensible, elle fonctionne sous 3,3 V et ne supporte même pas 3,6 V. Contrairement au nano-ordinateur conçu par Upton (RPI), l'ESP32 ne tolère en aucune façon le 5 V ; c'est un problème quand on veut relier la carte à un bus SPI ou I²C.

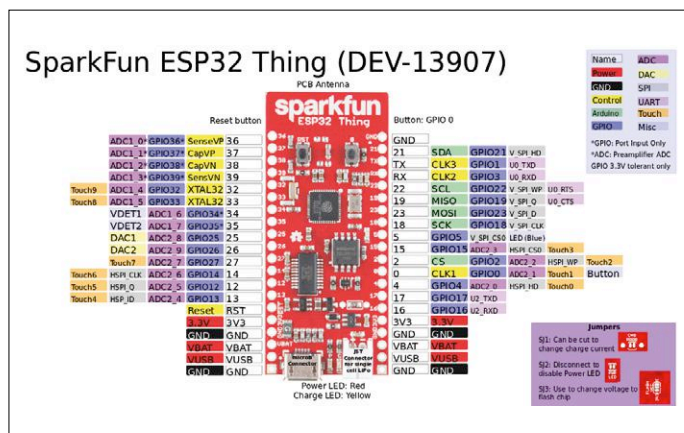
En revanche, elle travaille déjà avec 2,2 V. Donc si vous l'intégrez à l'un de vos montages, il n'est pas indispensable de lui procurer du 3,3 V. La plage de tension admissible entre 2,2 et 3,6 V ressemble beaucoup à celle des cellules au lithium-ion ; c'est probablement, voire délibérément, une application possible.

Mais revenons à nos moutons et analysons à l'oscilloscope le temps de démarrage de l'ESP32 : 2,5 s (cf. **figure 4**).

À ce sujet, on remarque que la carte d'évaluation utilise un chargeur d'amorçage relativement complexe. Pendant ce temps-là, la LED s'allume à plus faible puissance, on le constate bien sur le premier canal de l'oscilloscope.

Spécialisation

Bien qu'il soit tentant de passer à ce stade à l'analyse du domaine de modulation pour mesurer la vitesse de commutation de la sortie, en fait, on rencontre très peu l'ESP32 dans des tâches vraiment en temps réel, soit à cause de la néces-



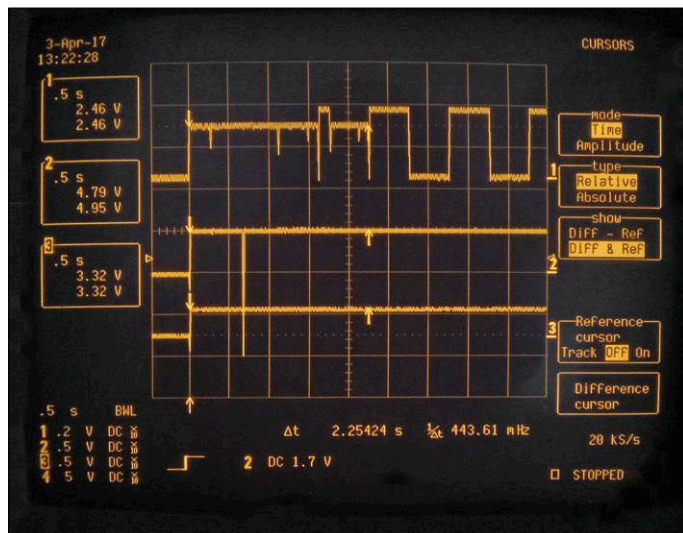


Figure 4. Avant que le programme de clignotement démarre, il peut s'écouler plus de 2 s.

sité de recourir à FreeRTOS, soit pour une question de coût. Il est bien plus intéressant de profiter de la richesse en convertisseurs N/A de l'ESP32. Des utilisateurs d'Arduino ont écrit pour cela la méthode `analogWrite` qui, malheureusement, n'est pas supportée à ce jour par le SDK d'Espressif. Le fabricant propose à la place trois options. Le processeur de MLI LEDC qui peut travailler simultanément sur 16 lignes [6] ; le module Sigma-Delta [7] qui peut jouer sur huit canaux, et le classique CN/A [8] qui d'ailleurs ne traite que deux canaux. Nous préférons cette dernière méthode, la réalisation de petits générateurs de fonctions a toujours été une amusette. Si vous préférez utiliser plutôt d'autres éléments, suivez les liens, ils conduisent à des fichiers d'entête (.h) dans lesquels chaque module est décrit.

Le travail avec le CN/A

Comme toujours, mieux vaut aller d'abord examiner l'implémentation de l'API (*Application Programming Interface*). Ouvrez alors dans le navigateur de votre choix les liens ci-dessus et changez les désinences .h en .c.

Dans le cas du CN/A, la méthode adéquate s'écrit `__dacWrite`, dont le corpus ressemble à ceci :

```
void IRAM_ATTR __dacWrite(uint8_t pin, uint8_t value) {
    if(pin < 25 || pin > 26){
        return; //not dac pin
    }
    pinMode(pin, ANALOG);
    uint8_t channel = pin - 25;
```

Deux choses sont intéressantes ici. D'abord, dans le monde de l'ESP32, les deux canaux sont identifiés par les indices associés à leurs lignes de GPIO. La méthode attend pour cela les valeurs 25 ou 26 qui, en premier lieu, sont converties en identifiant de canal.

Il est curieux de constater qu'Espressif ne fait pas confiance à l'utilisateur de la méthode pour gérer réellement l'état des broches. C'est pour cette raison que la méthode `pinMode` est

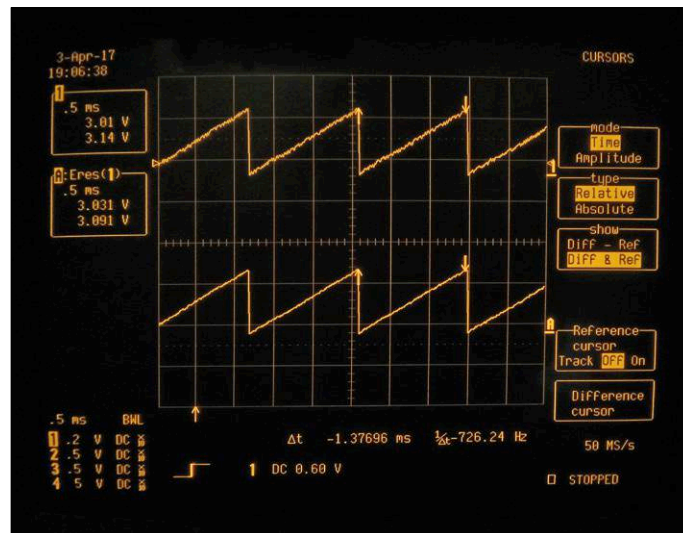


Figure 5. On peut produire toute une série de signaux au moyen du CN/A.

activée à chaque nouvel appel et il est logique que l'efficacité s'en ressente.

Sachant cela, nous pouvons retourner à la réalisation de la forme d'onde de sortie. Notre CN/A a une largeur de 8 bits, donc la variable d'exécution doit aller de 0 à 255 :

```
void loop() {
    for(int i=0; i<255; i++) {
        dacWrite(25, i);
    }
}
```

Reliez un oscilloscope à la broche sélectionnée pour admirer la forme d'onde représentée à la **figure 5**. Le duplicata de la trace est dû à une fonction spéciale de LeCroy - sur la figure 4, à côté de la trace originale est reproduite une version améliorée de 3 bits.

Listage 1. Notre premier programme ESP32 !

```
//File ElektorES032Scratch1

int ledPin = 5;

void setup()
{
    pinMode(ledPin, OUTPUT);
}

void loop()
{
    digitalWrite(ledPin, HIGH);
    delay(500);
    digitalWrite(ledPin, LOW);
    delay(500);
}
```

Liaison sans fil

Espressif fournit un SDK avec un groupe d'exemples de programme qui montrent comment se servir des API pour réseau : vous les trouverez sous [Fichier → Exemples → Exemple pour SparkFun ESP32 Thing](#).

Essayons maintenant la communication avec l'internet ; comme il y a là quelques pièges, réalisons le programme en deux étapes. La première démo sert uniquement à prouver que nous pouvons établir une liaison sans fil.

Il faut d'abord le nom du réseau sous forme d'une chaîne de caractères : capitales ou bas de casse à respecter, donc il faut utiliser autant que possible le presse-papier dans le programme. Pour retrouver le nom de votre réseau Wi-Fi, il est recommandé d'appeler l'interpréteur de commandes en ligne (*cmd*) et de saisir la commande *netsh*.

On y voit beaucoup de choses, mais l'essentiel, voici comment il se présente chez moi :

```
C:\Users\TAMHAN>netsh wlan show all

. . .

Profils sur l'interface Wi-Fi
-----

===== AFFICHER LES PROFILS =====
Il y a 1 profil utilisateur : Tamoggemon Holding k.s.
BAa

Informations sur le profil
-----

Version   : 1
Type      : Réseau local sans fil
Nom       : Tamoggemon Holding k.s. BAa

Paramètres de connectivité
-----

Nombre de SSID : 1
Nom du SSID : Tamoggemon Holding k.s. BAa
```

Dans le code du programme, il faut inclure l'entête WLAN `<WiFi.h>` ; on ajoute ensuite deux autres variables, le nom et le mot de passe à utiliser sur le réseau :

```
#include <WiFi.h>

const char* ssid      = "Tamoggemon Holding k.s. BAa";
const char* password = «XXX»;
```

Viennent ensuite les tâches ménagères ; après établissement de la liaison série vers le PC, fixons un délai pour permettre à l'ESP32 de donner les instructions de configuration :

```
void setup() {
    Serial.begin(115200);
    delay(10);
```

Maintenant, la construction de la liaison proprement dite commence. Ce qui compte, c'est que `WiFi.begin(...)` réponde tout de suite. Vérifions que la liaison a bien été établie sur `WiFi.status()` en interrogeant :

```
WiFi.begin(ssid, password);
while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
}
```

Enfin, donnons l'adresse IP attribuée par le routeur du réseau Wi-Fi :

```
Serial.println("");
Serial.println("connected - IP is:");
Serial.println(WiFi.localIP());
}
```

Le contenu de la méthode `loop` peut rester vide provisoirement. Exécutons alors le programme avec le moniteur série ouvert pour obtenir la sortie comme à la **figure 6**. Notez que nous sommes ici en 115 200 b/s : après le démarrage, le moniteur travaille au débit binaire de 9 600 et ne peut évidemment pas décoder les informations entrantes.

Données issues du réseau

À présent, on peut se risquer à télécharger des informations sur l'internet. On utilise l'objet `WiFiClient` pour communiquer avec les serveurs TCP.

`WiFiClient` ne traite pas des protocoles au niveau « application ». C'est pourquoi il suffit de donner le nom de l'hôte à appeler et le numéro de port pour établir une liaison. En cas d'échec de la connexion, c'est un `false` qui revient :

```
void loop() {
    WiFiClient client;
    const int httpPort = 80;
    if (!client.connect("tamoggemon.com", httpPort)) {
        Serial.println("connection failed");
        return;
    }
```

L'absence d'un pilote HTTP nous oblige à créer une demande valable manuellement. Comme cet article ne décrit pas du tout le protocole HTTP, il n'y a pas d'autres considérations à faire sur les chaînes de caractères qui figurent dans le listage :

```
String url="/test/2017/Elektorfile.txt";
client.print(String("GET ") + url + " HTTP/1.1\r\n"
+ "Host: " + "tamoggemon.com" + "\r\n"
+ "Connection: close\r\n\r\n");
unsigned long timeout = millis();
```

Le traitement des demandes HTTP peut prendre un temps plus ou moins long. Si aucun octet n'est arrivé dans les 5 s, on envoie un signal d'arrêt (*timeout*) :

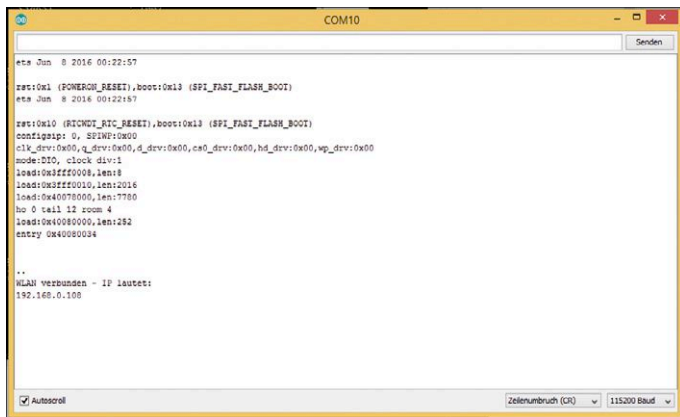


Bild 6. Hourra, nous sommes sur l'internet !

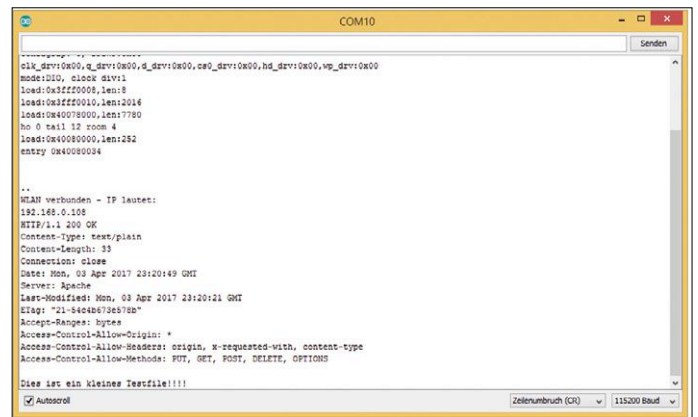


Figure 7. HTTP est très causant.

```
while (client.available() == 0) {
    if (millis() - timeout > 5000) {
        client.stop();
        return;
    }
}
```

Après l'échange de données, on lit peu à peu ce que l'objet `client` a envoyé et on retourne au moniteur sériel :

```
while(client.available()) {
    String line = client.readStringUntil('\r');
    Serial.print(line);
}
```

Comme il est fatigant d'interroger continuellement le serveur, on peut appeler la méthode `delay(...)`. Elle offre à l'ESP32 un repos de 50 s :

```
delay(50000);
}
```

On peut maintenant envoyer le croquis vers l'ESP32. Ne vous étonnez pas que la console se remplisse de diverses sorties : chaque demande HTTP fournit, en plus d'informations demandées, des entêtes parfois longs comme à la **figure 7**.

Conclusion

Les cartes Raspberry Pi Zero W et OrangePi Zero sont des concurrentes sérieuses de l'ESP32. Il y a peu d'écart de prix entre les cartes de liaison compactes. Si vous projetez de réaliser une petite série, l'ESP32 peut vous faire gagner beaucoup d'argent : à partir de quelques pièces, il vaut mieux acheter des puces ESP32 individuelles qui sont nettement moins chères que les cartes de liaison.

Nous n'en avons pas fini des ESP32. Dans la prochaine édition, nous verrons l'API d'origine qui offre quelques autres fonctions et permet d'atteindre de meilleurs résultats. ◀

(160454 – version française : Robert Grignard)

Liens

- [1] <http://eu.mouser.com/new/sparkfun/sparkfun-things/>
- [2] <http://esp32.net/#Hardware>
- [3] <https://git-scm.com/download/win>
- [4] www.elektormagazine.fr/160454
- [5] <https://cdn.sparkfun.com/datasheets/Wireless/WiFi/ESP32ThingV1.pdf>
- [6] <https://github.com/espressif/arduino-esp32/blob/master/cores/esp32/esp32-hal-ledc.h>
- [7] <https://github.com/espressif/arduino-esp32/blob/master/cores/esp32/esp32-hal-sigmadelat.h>
- [8] <https://github.com/espressif/arduino-esp32/blob/master/cores/esp32/esp32-hal-dac.h>
- [9] http://espressif.com/sites/default/files/documentation/eco_and_workarounds_for_bugs_in_esp32_en.pdf

l'électronique dans l'espace

la recherche spatiale pour tous

Jaime Estela

Les satellites sont construits avec des matériaux robustes, car ils doivent affronter des conditions extrêmes sans tomber en panne. Les composants électroniques utilisés sont fabriqués dans cette optique et coûtent très cher. Pour les petits satellites, des composants disponibles dans le commerce offrent une alternative intéressante, mais il faut vérifier qu'ils peuvent être utilisés dans l'espace (par simulation) et qu'ils répondent à des normes internationales.



Mike Hopkins, l'astronaute de la NASA sort un vase Dewar d'une armoire frigorifique pour analyser des échantillons biologiques (photo : NASA).

Les phénomènes physiques qui règnent dans l'espace peuvent altérer les composants et les matériaux. Pour les satellites, il est nécessaire de bien connaître cet environnement pour éliminer ou réduire les défaillances électroniques. Le comportement exact des composants dans

ces conditions hostiles dépend de l'orbite du satellite et peut être simulé par logiciel. Les résultats de ces simulations aident les fabricants de satellites à les mettre au point.

La **figure 1** répertorie les phénomènes physiques qui influent sur la durée

de vie et la fiabilité des composants électroniques.

- **Oxygène atomique** : le rayonnement UV casse les molécules d'O₂ en deux atomes d'oxygène. L'oxygène atomique est très réactif et,

par conséquent, érode la surface de la structure externe des satellites. Cette oxydation modifie les échanges thermiques de la structure du satellite. L'équilibre thermique du satellite dans l'espace est déplacé, ce qui peut avoir de fâcheuses conséquences.

- **Plasma** : les gaz ionisés engendrent des charges statiques qui s'accumulent à la surface externe du satellite. La décharge aléatoire de cette électrisation peut causer des dysfonctionnements du satellite et de ses instruments.

- **Rayonnement** : ces phénomènes peuvent avoir de nombreux effets. Les rayons gamma détériorent les composants électroniques. Les protons et les ions lourds peuvent dans le meilleur des cas modifier les données numériques et au pire littéralement détruire l'électronique du satellite.

- **Micrométéorites et déchets en orbite** : la palme du risque revient aux objets naturels et artificiels qui circulent dans l'espace. Il ne faut pas sous-estimer ce risque, car ces objets peuvent tout à fait endommager ou détruire des satellites. Des accidents se sont déjà produits avec perte pure et simple du vaisseau spatial.

Altération de l'électronique

Au début de l'exploration spatiale, il n'y avait pas d'électronique spécifique pour les missions. Les composants à usage militaire ont été durcis pour les applications spatiales. Dans ce processus, des essais complémentaires étaient effectués pour sélectionner les composants les meilleurs.

D'ailleurs, en 1973, l'électronique de la station spatiale Skylab utilisait des composants militaires. Lors des essais de qualification, les nombreuses défaillances ont montré qu'il fallait améliorer considérablement l'électronique. Ces améliorations ont coûté plus de trois millions de dollars, car il a fallu créer de nouveaux composants et donc réaliser de nouveaux essais de qualification.

L'électronique de la première mission Space Shuttle (1981) faisait de nouveau appel à des composants militaires. Pour améliorer la fiabilité du système,

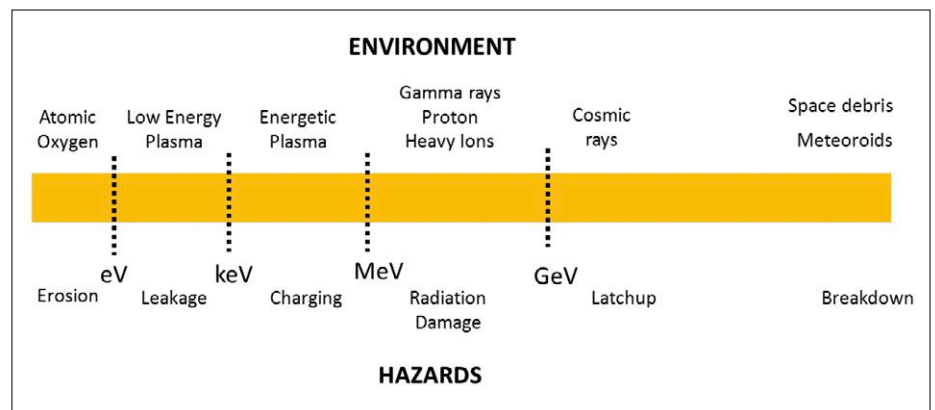


Fig. 1. Les conditions ambiantes dans l'espace et les risques encourus (source : Spectrum Aerospace)

les composants étaient pour la plupart six fois redondants. Une méthode statistique (schéma de sélection) permettait de vérifier la validité des données. L'utilisation de la redondance impliquait aussi l'augmentation du poids et de la consommation ainsi que d'autres exigences dans la conception du matériel et du logiciel. Comme l'électronique militaire n'était pas vraiment adaptée à l'espace et que la sélection n'apportait pas en soi d'amélioration aux composants, dès les années 60 le développement systématique de composants spécifiques pour les vaisseaux spatiaux a commencé avec des procédés de fabrication aux exigences qualitatives bien supérieures. Aux États-Unis, la nouvelle génération de composants s'appuyait sur des composants militaires et des essais complémentaires de qualification. Cette stratégie a permis de réduire les coûts de production des missions spatiales, mais elle visait aussi à atteindre une plus grande robustesse des composants. Cependant, comme la demande est demeurée faible, les prix sont restés élevés.

Aujourd'hui, la qualification selon les normes de l'ESA ou de la NASA représente un investissement d'au moins un million de dollars par composant. La certification prend en gros deux ans [1]. Il va de soi que la complexité des essais varie avec le type de composant. L'essai d'une diode coûte moins cher que l'essai d'un microcontrôleur.

Essais de qualification

L'expérience accumulée dans les nombreuses missions spatiales et les techniques développées au cours de ces soixante dernières années permettent

à des organismes comme l'ESA et la NASA de concevoir des directives et des normes de qualification. Ces directives sous-tendent la pertinence de la qualification des composants électroniques des missions spatiales. Les normes établies par l'ESA peuvent être consultées sur le portail des composants de l'agence spatiale européenne, l'ESCIES (*European Space Components Information Exchange System*) [2]. Une base de données des circuits intégrés qualifiés (rapports d'essai inclus) est disponible sur le site de l'ESCIES [3].

Pour un essai de qualification, il faut préparer les « composants à tester » (*Device Under Test, DUT*) et configurer le système de mesure. Le nombre exact de composants est d'abord déterminé et, selon le type d'essai, une partie ou la totalité des composants sont utilisées. Cela signifie que parfois quelques composants suffisent pour la qualification (par ex. pour les essais d'irradiation) tandis que, pour un test de dégazage par ex., tous les composants sont utilisés. Les essais permettent de caractériser les DUT.

Pour la qualification, il y a une exigence importante : les composants doivent être tous identiques (géométrie, taille, matériaux, etc.) et doivent provenir du même lot.

Le résultat de chaque essai est consigné dans un rapport. Après la qualification, un rapport final consigne l'évolution des caractéristiques pertinentes du composant dans les conditions spatiales.

Une qualification complète qui mobilise beaucoup de ressources s'appelle un *Screening Test* (criblage). C'est un processus qui prend quelques années

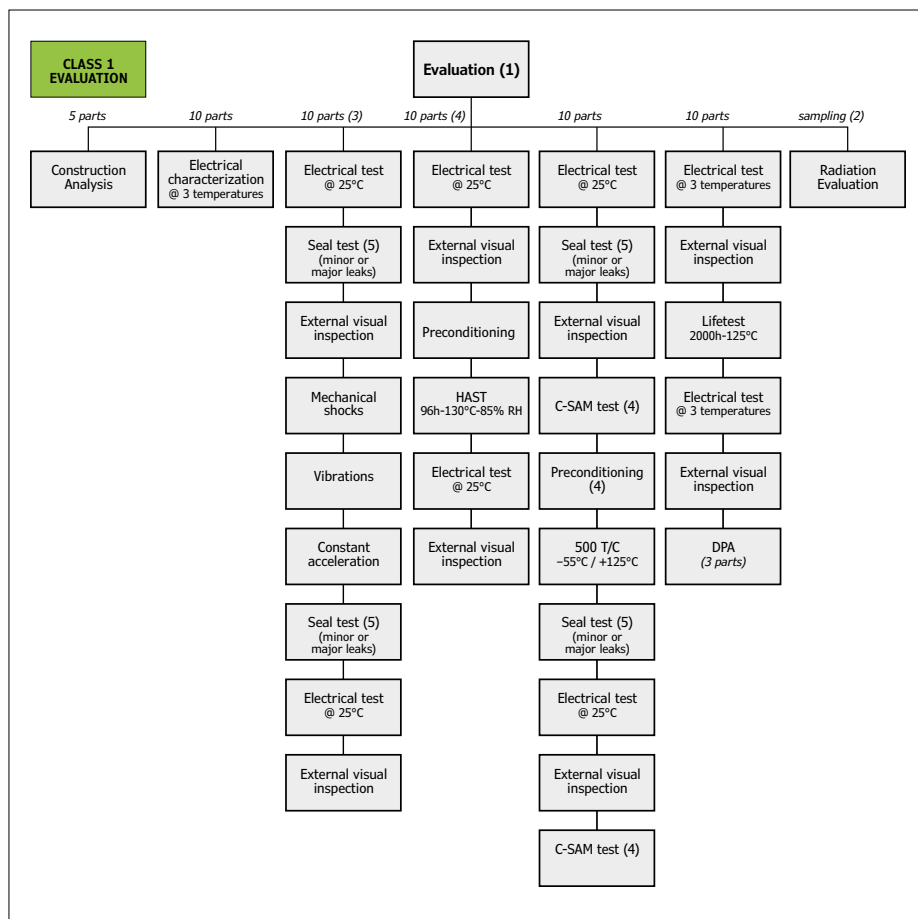


Fig. 2. Déroulement d'un essai de classe 1 (ECSS-Q-ST-60-13C Evaluation tests flow Class 1, source : ESA)

et nécessite un investissement élevé. Dans les criblages, les essais individuels ci-dessous sont réalisés :

- essai électrique
- contrôle d'étanchéité
- inspection visuelle
- choc mécanique
- essai en vibration
- accélération constante
- essai thermique
- essai sous rayonnement
- dégazage
- essai de stress
- choc thermique
- soudabilité

Dans le document ECSS-Q-ST-60-13C assurance qualité des produits spatiaux, l'ESA décrit différentes classes de qualité [4]. La différence entre les classes tient à l'étendue du processus de qualification. La classe 1 concerne une qualification complète (fig. 2), tandis que pour la classe 3 le processus de qualification est court (principalement des essais sous rayonnement).

Caractérisation

Une courte description d'un essai TID (Total Ionizing Dose / dose ionisante totale) subi par un amplificateur opérationnel sans dérive LTC2052 illustre le déroulement de tels processus de qualification. On prépare un châssis équipé de cartes qui portent les composants à essayer (fig. 3). Ici, on considère deux configurations décrites fig. 4 (condition de polarisation). Il s'agit d'un mode *arrêt*, dans lequel toutes les broches sont mises à la masse et d'un mode *marche* dans lequel un point de fonctionnement donné est configuré, sans que le composant ne traite de signaux (une sorte d'état d'attente).

La fig. 5 montre le montage d'essai. Une source au cobalt 60 est chargée de l'irradiation. Les deux configurations d'essai sont irradiées. Pendant cette irradiation, les composants sont évalués périodiquement par un système de mesure (Automatic Test Equipment, ATE). Les résultats de la caractérisation définissent la dégradation des composants quand la

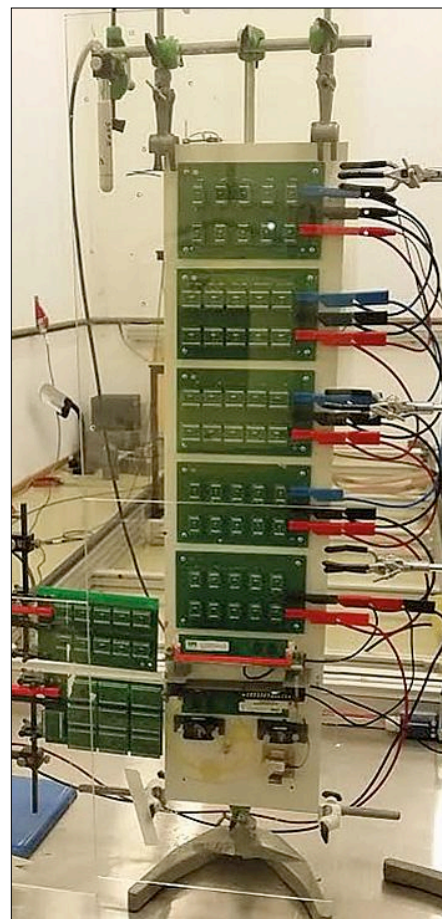


Fig. 3. Le support avec les DUT est prêt à subir l'essai d'irradiation (source : Spectrum Aerospace).

dose reçue augmente. La diminution des performances dépend de la technologie du composant et de la dose de rayonnement cumulée. On observe par ex. une consommation qui croît avec la dose de rayonnement reçue (fig. 6). À la fin de l'essai, on observe la modification de certains paramètres électriques ; dans certains cas, des composants sont totalement détruits.

Marché des petits satellites

Le marché des petits satellites connaît une croissance impressionnante. De nouveaux produits et services sont recherchés pour couvrir des besoins croissants. Pour la fabrication de petits satellites, le choix de composants militaires ITAR (International Traffic in Arms Regulations) n'est pas une bonne solution. Mieux vaut utiliser des composants COTS (Commercial Off-The-Shelf, produits en série) qualifiés.

Les études de marché conduites par la société Northern Sky Research indiquent que dans les dix prochaines années, plus de 2500 petits satellites (jusqu'à

Jaime Estela : questions et réponses

- *Il a été question d'un concours doté d'une place gratuite pour une carte sur un vol StarLab. Qu'en est-il ?*

C'est vrai, et c'est toujours d'actualité. Mais il y a des retards, car l'une des sociétés qui souhaitait organiser le concours et avait une bonne expérience dans ce domaine s'est retirée. Nous sommes en discussion avec d'autres sociétés, mais cela n'a pas encore abouti.

- *Quand le premier vol aura-t-il lieu ?*

Le premier vol devrait avoir lieu en décembre, mais tant que tout ne sera pas parfaitement clarifié, la date ne peut pas être arrêtée. En principe les vols auront lieu trimestriellement.

- *Quel est l'ordre de grandeur du coût ?*

Le prix dépend de la taille de la carte. Pour une petite carte ($5 \times 3 \text{ cm}^2$), ce sera à partir de 3000 € environ.

- *Comment les expériences seront-elles surveillées ?*

L'électronique dans l'espace est surveil-

lée par télémétrie numérique. Des capteurs mesurent certains paramètres et les résultats de ces mesures sont retransmis vers la terre. Ainsi, on sait exactement comment l'électronique travaille dans l'espace. Un développeur doit cependant se concentrer sur sa propre expérience. C'est pourquoi un kit de développement *StarLab* est mis à disposition. Cette électronique auxiliaire décharge le développeur d'électronique afin qu'il puisse rapidement et sans complication mettre son expérience sur pied, l'électronique de télémétrie étant déjà prête.

- *Est-il possible de tester seulement des composants ou également des groupes de composants ?*

Les deux sont possibles. Les composants individuels prennent plus de temps et les coûts sont plus élevés. Les résultats sont cependant plus précis. Pour des cartes complètes, les coûts et le temps d'essai sont plus faibles, mais l'incertitude est plus grande. Le problème avec des groupes de composants ou des cartes complètes est qu'en cas de défaillance, il est plus difficile de localiser précisé-

ment la panne.

Tous les composants n'ont pas la même robustesse. C'est pourquoi il est préférable d'établir la robustesse des composants avant d'exploiter un système dans l'espace. Si on procède à l'inverse, il est possible que la robustesse d'un seul composant compromette celle de l'ensemble. J'ai lancé un nouveau programme intermédiaire entre composant et carte. Il permet de mieux qualifier les cartes complètes.

- *À qui dois-je m'adresser si j'ai des questions ou si je suis intéressé par le projet ?*

Jaime Estela

jaime.estela@gmail.com

jaime.estela@spectrum-aerospace.com

www.linkedin.com/in/

jaime-estela-9045b63b/

Skype : jaimeestela

100 kg) seront lancés dans l'espace [5], une autre étude de la société *Spacework* conclut que 3000 petits satellites de moins de 50 kg seront lancés entre 2014 et 2020 [6]. Ces deux prévisions tiennent compte des futures constellations de satellites (produits en série et par conséquent notablement meilleur marché comme les satellites GPS). Une croissance de la fabrication en série de

satellites signifie un besoin accru en électronique qualifiée.

Composants *space-COTS*

Selon *NewSpace* [7], avec l'émergence de nombreuses jeunes pousses (nord-américaines) dont l'objectif est de proposer des voyages dans l'espace meilleur marché et plus accessibles, les *space-COTS* sont le point d'accès aux

composants commerciaux qualifiés pour l'espace. Le terme de *space-COTS* et son acception résultent de l'intégration d'électroniques commerciales dans les missions de petits satellites et également de la longue pratique des normes de l'ESA pour la qualification de composants électroniques. Les *space-COTS* réalisent un compromis entre des composants n'ayant aucune qualification

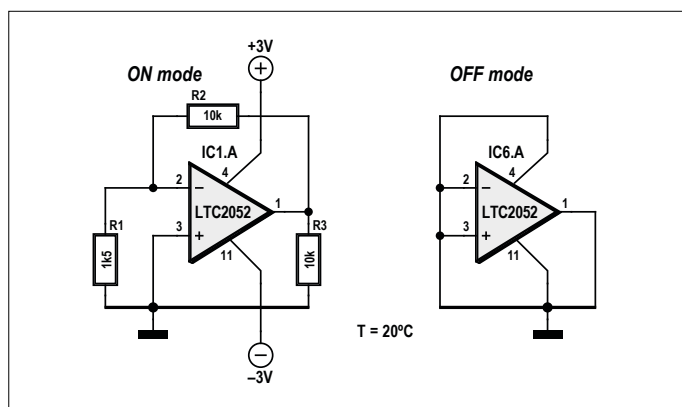


Fig. 4. Les deux configurations du LTC2052 pour les essais TID (source : Spectrum Aerospace).

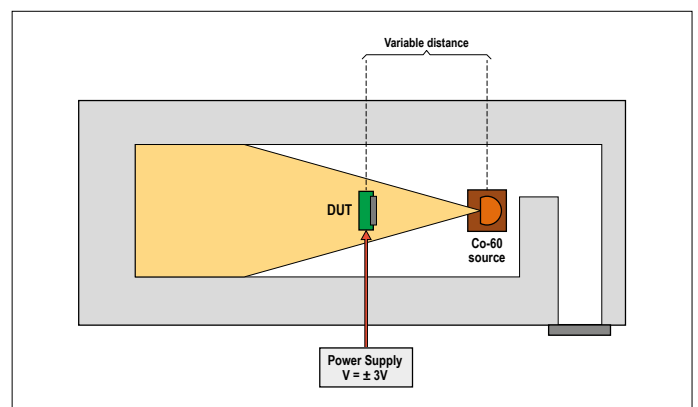


Fig. 5. Montage d'essai avec une source d'irradiation au cobalt 60 (source : Spectrum Aerospace).

(par ex. dans les *CubeSats*, « nanosatellites ») et les composants 100 % qualifiés *ITAR*. L'expérience acquise lors des missions des petits satellites montre que de nombreux composants commerciaux peuvent globalement travailler plusieurs années dans l'espace bien qu'ils ne soient pas conçus pour cela.

Les essais de qualification des *space-COTS* sont progressifs et en fonction de leur étendue on distingue les classes suivantes :

- Classe A :
criblage complet (*full screening*)
- Classe B :
essais *TID* et *SEE*
(*SEE* = *Single Event Effect*)
- Classe C : essai *TID*

Ces classes respectent les directives de l'*ESA* et de la *NASA* ainsi que les nouvelles normes *ISO* (*International Organization for Standardization*).

Projet StarLab

De nombreuses sociétés développent de nouveaux produits pour voyager dans l'espace. Ces produits ne peuvent cependant pas avoir l'expérience des vols spatiaux (*Flight Heritage*). Pour essayer un produit dans l'espace, il faut déboursier au moins 250 000 € (prix du nanosatellite plus vol, sans essai). En outre, une liste d'attente d'au moins 18 mois pour le vol met la patience et le porte-monnaie à rude épreuve. Il existe bien des satellites pour les essais technologiques, mais ces vols ont lieu très rarement et sont réservés à un petit nombre de sociétés privilégiées [8].

Cependant, il existe des plateformes destinées à mettre la station spatiale *ISS* à la portée d'un plus large public. Ces plateformes autorisent l'installation de petites charges utiles privées. L'une d'elles s'appelle le projet *StarLab*. Un conteneur abrite diverses expériences ; chaque expérience dispose d'une alimentation et d'une liaison numérique, les données d'essai sont transmises sur terre rapidement. Le projet *StarLab* avec sa qualification novatrice permet un accès récurrent à l'espace pour M. Tout-le-Monde, et ce à un prix modique. La communauté des créateurs peut ainsi mettre au point de bonnes électroniques. De tels développements pourraient aussi

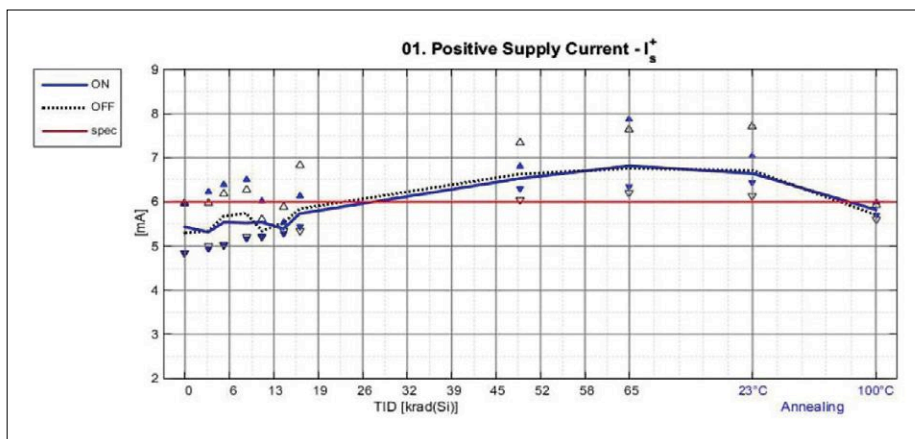


Fig. 6. Résultat d'un essai : la consommation augmente avec la dose de rayonnement reçue (source : Spectrum Aerospace).

s'avérer intéressants pour la navigation spatiale, cependant les cercles de créateurs méconnaissent encore largement ce domaine. Le projet *StarLab* pourrait leur donner des ailes et le moyen de tester ces nouvelles électroniques dans l'espace. Essayer ses propres montages dans l'espace reste un grand défi, mais avec le projet *StarLab*, cet exercice contraignant et coûteux est considérablement plus simple et meilleur marché. Un champ d'investigation tout à fait nouveau s'ouvre aux développeurs et la navigation spatiale pourra profiter de produits novateurs et qualifiés.

Cet article a éveillé votre curiosité ? Vous avez envie de faire un tour dans l'es-

pace ? L'encadré répond aux questions qui vous brûlent les lèvres à propos du projet *Starlab*. La porte de l'espace vous est grande ouverte ! ◀

(160036 – version française : Yves Georges)

Sources et liens internet

- [1] *The history of space quality EEE parts in the United States*, Leon Hamiter, Components Technology Institute : www.cti-us.com/pdf/HistoryEEESpacePartsinUSA.pdf
- [2] European Space Components Information Exchange System : <https://escies.org/webdocument/showArticle?id=167>
- [3] ESCIES Radiation List : <https://escies.org/labreport/radiationList>
- [4] ECSS-Q-ST-60-13C_Space product assurance, ESA Requirements and Standards Division ESTEC, P.O. Box 299, 2200 AG Noordwijk, The Netherlands.
- [5] *Nano- And Microsatellite Markets Report*, Northern Sky Research, LLC, One Mifflin Place, Suite 400, Cambridge, MA 02138.
- [6] *Small Satellite Market Observations 2015 - Forecast*, Dr. John Bradford, Spaceworks Enterprises Inc., 1040 Crown Pointe Parkway, Suite 950, Atlanta, GA 30338 USA.
- [7] NewSpace – Wikipédia : <https://en.wikipedia.org/wiki/NewSpace>
- [8] *DLR IOV/IOD Workshop*, Marc Jochemich, DLR Space Administration, Brussels, 09/2016.

« le menu, s'il vous plaît ! »

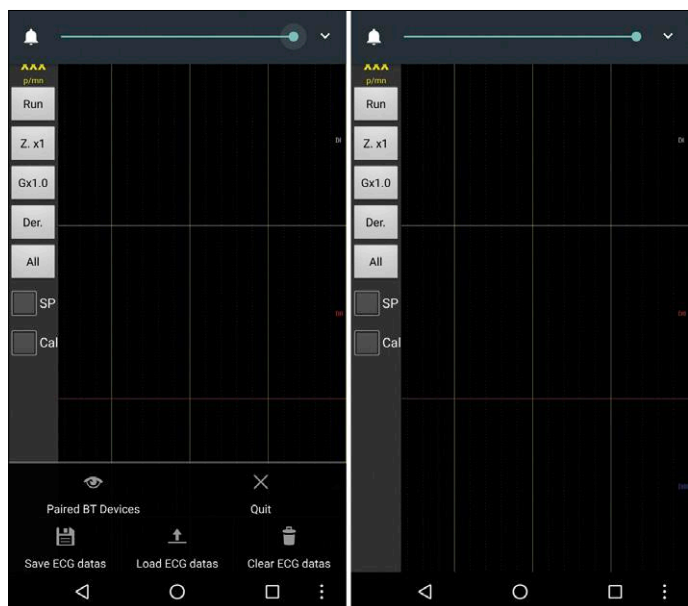


Figure 1. Écrans de l'appli Cardioscope, avec et sans touches.

Luc Lemmens (labo d'Elektor)

Il arrive qu'un programme refuse obstinément de fonctionner ; c'est parfois le cas avec des logiciels anciens qui ne peuvent plus tourner tels quels sous une nouvelle version d'un système d'exploitation. Lorsqu'un lecteur nous a posé une question relative à l'appli de notre Cardioscope Android de 2013, c'est ce à quoi nous avons tout d'abord pensé, mais il s'agissait – heureusement – d'un autre problème.

Ce lecteur, intéressé par le projet, ne parvenait pas à faire apparaître les boutons en bas de l'écran de l'appli. Ces boutons servent à la commande du cardioscope (**figure 1**), et sont donc essentiels pour l'appli. Ce qui manquait à l'appareil du lecteur, ce sont des touches de menu, qui auraient permis de restaurer les boutons.

Les appareils tournant sous Android, tablettes ou téléphones, ont généralement trois touches en bas de l'écran : *Home*, *Back* et *Menu* ; certains ont aussi une touche *Search*.

Sur les premiers appareils, on avait de véritables touches ; aujourd'hui on se contente souvent d'une touche *Home*, flanquée de deux touches capacitives. Le design varie d'un fabricant à l'autre, de même que l'emplacement des touches et les icônes utilisées. La **figure 2** vous montre un échantillon – non exhaustif, loin de là ! – de divers téléphones.

La tendance actuelle est même de ne plus montrer ces touches, au profit d'une vue d'ensemble (*Overview*) ou des boutons permettant à l'utilisateur de passer d'une appli active à une autre (*App Switch*). Notre malheureux lecteur a sans doute un appareil fonctionnant ainsi, mais il fallait quand même une solution. Elle fut trouvée assez vite : si l'on maintient la vue



Figure 2. La diversité des touches de menu Android.

d'ensemble un certain temps, l'écran passe de *App Switch* à *Menu* ; ceci permet de faire apparaître les touches de l'appli Cardioscope.

Nous avons pu donner satisfaction au lecteur assez rapidement, mais si on creuse un peu on se rend compte de l'énorme diversité (voire prolifération) des commandes des appareils Android. Pour terminer sur une touche personnelle, je me suis remémoré mon « combat » avec mon ordiphone flambant neuf l'année dernière : ce n'est qu'après moult essais (avec jurons et appels à l'aide !) que je me suis familiarisé avec les commandes ; encore toutes mes excuses aux collègues et amis que j'ai dérangés... ◀

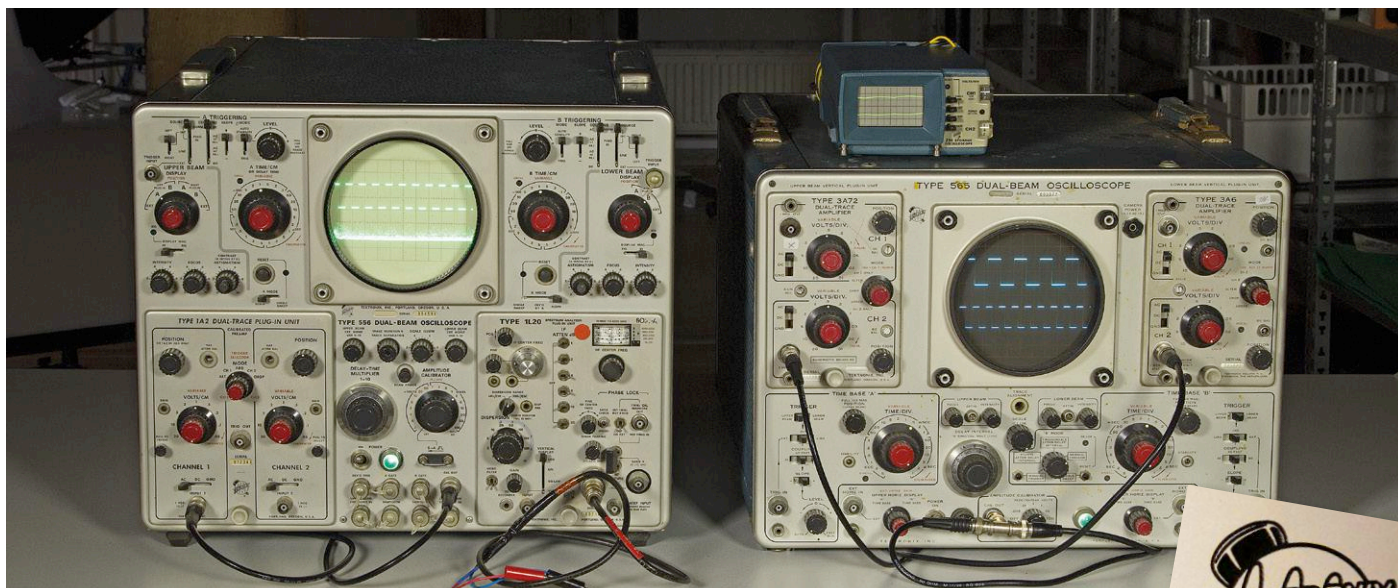
(160349 – version française : Jean-Louis Mehren)



Figure 3. Touches de vue d'ensemble et de basculement entre applis.

oscilloscopes Tektronix 556 et 565 à deux faisceaux

corpulents, lourds et patauds



Jan Buiting, rédacteur en chef de Rétronique

« Quoi ? Et en plus ils se vantent d'avoir deux canaux ? » C'est ce que j'entends lorsque je demande de l'aide pour déplacer mes Tektronix 556 et 565. Ce sont deux des plus gros, des plus lourds et des plus onéreux instruments à tube cathodique, qui servaient entre autres à chauffer les labos d'électroniciens. On a souvent porté au pinacle ces colosses à tubes de la marque dominante du marché, mais qui contenaient quand même pas mal de semi-conducteurs et il existe au moins un Tektro encore plus lourd.

C'est un double FAISCEAU, Monsieur !

Dans leur fierté, les utilisateurs décrivaient inconsidérément les oscilloscopes à tubes comme à 1, 2 ou 4 canaux, traces, voies ou *beam*. J'estime que 98 % de tous les oscilloscopes vendus n'ont qu'un seul faisceau, malicieusement découpé ou alterné si vite dans le tube cathodique que le cerveau en percevait deux ou plusieurs traces distinctes. Cela marche très bien jusqu'à quatre traces de signaux ordinaires, observées dans un labo domestique, pour autant

que la vitesse d'écriture sur l'écran soit suffisante et que le tube cathodique soit de bonne qualité. Ce n'est pas un souci avec la plupart des appareils de marque réputée et la technologie a atteint des sommets.

C'est tout à fait acceptable pour le travail quotidien actuel, mais l'alternance verticale rapide du faisceau électronique dans le tube peut occasionner du papillement ou des artefacts sur des signaux exigeants, d'autant plus s'ils diffèrent fort en récurrence ou en fréquence. Au temps où l'on ignorait encore tout de la

photo numérique, réaliser un instantané (Instamatic Kodak) de l'écran avec des traces multiples tenait de la gageure.

C'est sans doute pour cela que les scientifiques et les instituts de recherche, comme la NASA, ont puisé dans les crédits pour se doter d'oscilloscopes de 15 MHz à deux faisceaux plutôt que d'appareils à deux ou quatre canaux « bien plus rapides et beaucoup moins chers ». Mais s'il vous faut absolument des observations fiables de deux signaux sur le même écran, c'est d'un double faisceau que vous devez disposer. Hélas,



Figure 1. Zone centrale de la face avant de l'oscilloscope Tektronix 556 avec commandes et connecteurs communs aux deux faisceaux.

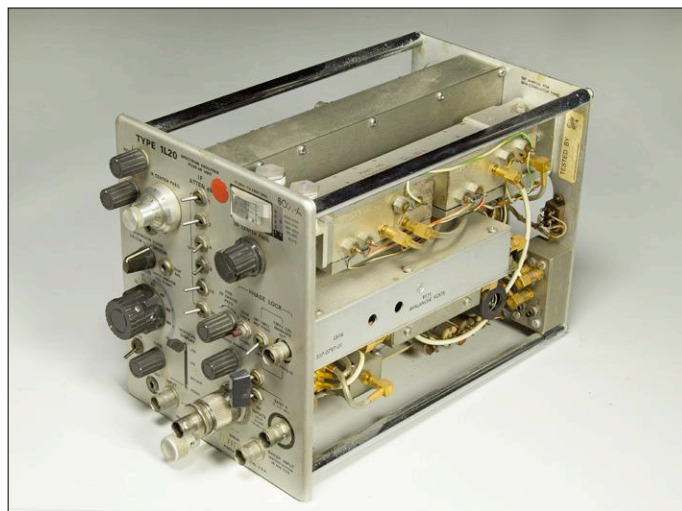


Figure 2. Le tiroir analyseur de spectre de type 1L20, de 10 MHz à 4 GHz, pour les oscilloscopes de la série 500 de Tektro est un oiseau rare.

ces instruments étaient beaucoup plus onéreux en raison des ajouts à incorporer au tube image, doublement des plaques de déflexion, second ampli vertical et des circuits supplémentaires comme pour les parties THT séparées, parfois même une deuxième base de temps.

Il faut le voir pour le croire : l'image d'un oscilloscope à deux faisceaux est extrêmement calme pour les yeux et rien d'étrange ne vient la perturber quand le signal passe d'une voie à l'autre. Et puis, vous pouvez aussi découper le signal ou alterner les faisceaux pour avoir 2×2 ou même 2×4 traces. Ou encore, observer une impulsion dans une période de signal à 50 Hz en haut de l'écran tout en visualisant une dizaine de périodes d'un signal à 50 MHz en bas.

Tektronix a produit un nombre limité d'oscilloscopes en version à deux canons à électrons. Il y a eu les 502, 551, 555, et 565 de la fameuse série 500, le 7844 de la célèbre série 7000 et le 5113 de la série économique 5000.

Je connais au moins un fabricant de scopes à double faisceau, c'est Philips avec la gamme transistorisée PM3230, PM3231, PM3232 et PM3233, ainsi que celui à tubes PM3236. Je serais curieux d'en connaître d'autres. Un oscilloscope à trois ou quatre faisceaux, je n'en ai jamais rencontré et vous ?

Offre spéciale ! Un Tektro 556

Mart, notre photographe attitré de l'équipe Elektor, malgré les tâches spécifiques, laborieuses, mais amusantes, accomplies pendant de nombreuses années pour *Rétronique*, a rechangé à

installer sur sa dispendieuse scène de prise de vues, coutumière des Arduino, RPi et autres précieuses électroniques, les monstres de plus de 45 kg que je lui présentais. Or, il se fait que j'avais une solide table de bureau de 160×80 cm, équipée de roulettes pivotantes et bloquantes, et capables de supporter les monstres 556 et 565 avec câbles, outillage, ainsi qu'un 214. Un « trolley » improvisé pour le transport.

La **photo en tête d'article** montre les deux Goliath et le petit David (le 214) en action, en train de soutirer à peu près 1 600 W d'une seule prise secteur de 230 V dans notre ancien studio photo. Les gros ventilateurs des 556 et 565 y faisaient grand bruit. Le 556 à gauche affiche son signal d'étalonnage à 1 kHz comme trace supérieure et l'autre faisceau représente en bas le spectre RF de la bande Wi-Fi à 2,4 GHz de notre réseau

sans fil interne, simplement capté par deux fiches bananes. La trace du bas montre la difficulté de photographier à distance une image non périodique à cause du travail de notre ordinateur et des friands d'ordiphones !

Alors que le gros 556 est un double faisceau de structure parfaitement symétrique avec de nombreux circuits dédoublés, ses boutons de commande et connecteurs sont communs aux deux canaux. Ils sont commodément disposés dans une zone centrale décrite à la **figure 1**. Ce 556 magnifiquement préservé porte le numéro de série 004584 de l'usine Tektronix de Portland, dans l'Oregon.

Le 556 est un scope configurable à deux travées prévues pour recevoir un grand choix de tiroirs. Vous voyez à gauche de la **figure 2** un ampli vertical à double trace de 50 MHz du type 1A2 et à droite



Figure 3. Vue intérieure du haut du Tektro 556 et son tube cathodique massif à double faisceau. Quel plaisir d'avoir maintenant des écrans LCD et LED sur nos oscilloscopes !

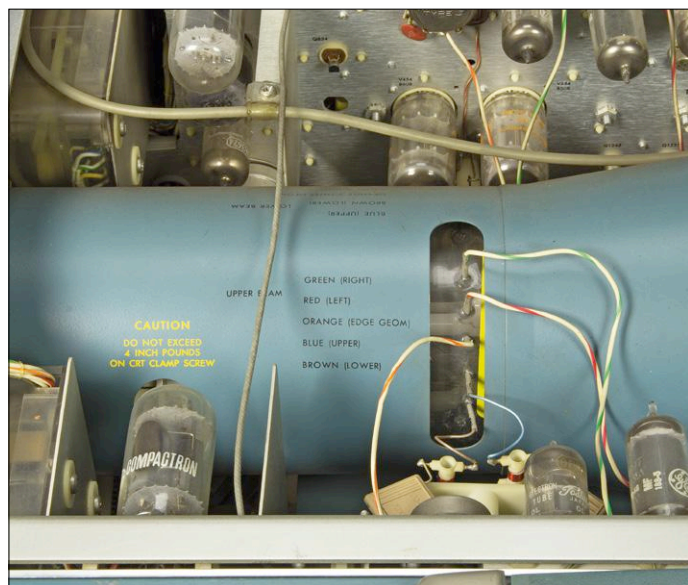


Figure 4. Les connexions au tube image dans le 556. Il n'y a que de simples fils munis de cosse à enficher et des broches à travers le verre.



Figure 5. Le tiroir à double faisceau Tektronix 3A6. Deux d'entre eux dans un oscilloscope 556 vous donnent quatre traces précises de deux faisceaux d'électrons.

un analyseur de spectre entre 10 et 4 200 MHz du type 1L20.

Pour vous donner une idée de la vraie grandeur de ces instruments en reproduction sur une page A4 du magazine, j'ai ouvert le toit du 556 et posé une pile de 9 V sur le couvercle du tube cathodique (figure 3). J'aurais pu en aligner dix. La figure 4 met mieux en évidence

les cinq connexions pour les plaques du faisceau supérieur, il y en a autant de l'autre côté du col du tube pour le faisceau d'en bas, façon de parler pour désigner les plaques de déflexion verticales. On voit aussi les deux extrémités des bobines de compensation pour ces plaques et le fameux tube Compactron ! Il n'a pas été nécessaire de dépanner

ce 556, un petit dépoussiérage a suffi. Il était muni des deux tiroirs représentés, auxquels j'ai eu la chance de pouvoir ajouter plus tard un 1L10 et un 1L30, ce qui me permet de couvrir de 1 MHz à 10,4 GHz.

Des ennuis avec le 565

Légèrement plus petit et moins lourd que le 556, le 565 tranche par des options supplémentaires pour le déclenchement ainsi que pour la séparation, le partage et l'échange des bases de temps. Le 565 n'a que 15 MHz de bande passante contre 50 MHz pour le 556, ce qui fait qu'on se demande pourquoi Tektro a décidé de le produire. La réponse est probablement l'envie d'étendre la série populaire d'oscilloscopes 560 avec un instrument à double faisceau. La série 560 fait usage de tiroirs spéciaux prévus pour piloter directement les plaques V et H du tube cathodique. La figure 5 montre le tiroir amplificateur pour deux faisceaux dans mon 565. Comparé au tiroir comme le 1L20 de la série 500 (cf. fig. 2 et entête), vous pouvez voir qu'il est plus étroit. À l'arrière, il y a deux gros tubes E55L SQ (*special quality*) en commande des plaques V dans l'instrument principal jusqu'à 10 MHz à basse capacité pour éviter pertes et distorsion. La gamme des tiroirs compatibles avec le 560 est aussi impressionnante avec ses unités à 4 ou 8 traces, l'échantillonnage en gigahertz – une prouesse pour l'époque et de nouveau un défi aux

Oscilloscopes à double faisceau de la série 500 de Tektronix			
type	poids (sans tiroir)	caractéristiques	font partie de la collection Rétronique d'Elektor
502(A)	23,6 kg	500 kHz ; haute sensibilité ; bas coût ; compact ; 1 base de temps ; sans ventilateur	✓
551	écran : 23,6 kg alim : 19,2 kg	30 MHz ou moins à haute sensibilité, 2 tiroirs V ; 1 base de temps, pas de retard ; alims séparées ; 1 ventilo	✓
555 « Triple Nickel »	écran : 31 kg alim : 20,5 kg	30 MHz ou moins à haute sensibilité ; 2 tiroirs V ; 2 tiroirs H ; retard ; alims séparées ; 2 ventilateurs	✗
556	40 kg	50 MHz ; 2 tiroirs V (lettre et 1xx série) ; 2 bases de temps ; alim intégrée ; 1 ventilateur	✓
565	31 kg	15 MHz ; 2 tiroirs série 560 (type 2xx/3xx) ; 2 bases de temps, retard ; 1 ventilateur	✓

concepteurs de Hewlett Packard !

Bien que cédé par un autre propriétaire, j'ai déménagé le 565 vers ma nouvelle maison, en même temps que le 556 et une boîte pleine de tiroirs dans ma Suzuki SX4. Avant même d'avoir l'autorisation de l'entrer dans le garage, j'ai dû le nettoyer, parce qu'il avait accumulé la crasse pendant 20 ans dans une cave industrielle avec trente tonnes d'e-déchets. J'ai allumé le 565 alimenté par un transformateur variable et après quelques heures, en approchant de 185 V~ (des 230 V du secteur), j'ai eu la chance de voir apparaître une trace. Au moment où je voulais placer au centre de l'écran la trace bleue magnétisante, j'ai entendu une petite explosion et la trace a disparu. J'ai éteint tout de suite, mais une mauvaise odeur a vite rempli la pièce, puis toute la maison. Je craignais l'expiration d'un condensateur électrolytique dans l'appareil, mais une inspection visuelle n'a rien donné. Aucun fusible fondu, j'ai rallumé l'instrument, tous les tubes ont rougeoyé et les cinq tensions réglées étaient dans une marge de 2 %, rien à redire. En reniflant un peu partout dans l'oscilloscope, j'ai enlevé le couvercle de la section HT et découvert le coupable : un condensateur tout noir avec des signes d'éruption. Comment un si petit objet peut-il schlinguer tellement par un aussi petit trou ? Le condensateur fait partie d'un oscillateur au centre d'un convertisseur HT réglé dans le 565. Quelques minutes après, un condensateur de 0,1 μ F / 630 V a pris sa place (**figure 6**) et après avoir remis en ordre de marche le 565, j'ai vu deux traces précises à l'écran. Le condensateur noir de 1 nF / 1 000 V Sprague au bout de mon crayon est un probable candidat à la retraite, mais il est encore en bon état. J'ai remarqué que les deux traces étaient trop courtes : 7,5 cm au lieu de près de 10 cm. J'ai donc ajusté les potentiomètres de SWEEP WIDTH et de x1 GAIN dans l'instrument, mais il faut encore investiguer et refaire l'étalonnage avec un marqueur temporel (comme le TG501). Les réglages semblaient poussieux et fatigués.

L'écran de mon appareil est garni de phosphore bleu et recouvert d'un graticule à mailles fines qui lui donne l'air d'un trou noir, mais fait voir de splendides traces sans halo ni surbrillance, même à forte intensité (réglable séparément sur chaque faisceau ☺). Notez aussi le connecteur de 6,3 V pour l'ap-

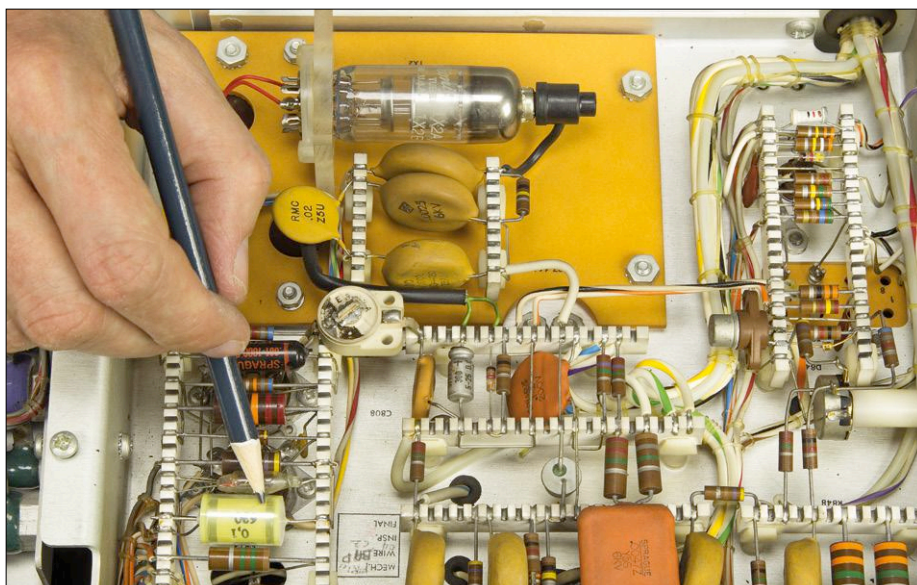


Figure 6. J'ai ranimé la partie haute tension du tube image de mon 565 en lui changeant un condensateur.

pareil photo ! Mart a apprécié le réseau noir mat qui lui a permis de rendre facilement la forme d'onde d'étalonnage interne avec des bases de temps réglées **individuellement** pour les **deux** traces **faisceaux** afin d'obtenir le cliché de la **figure 7**.

Conclusion

Ces appareils ont tant d'options qu'il faut bien passer un quart d'heure à les régler correctement avant de pouvoir prendre une mesure. Pensez aux deux faisceaux (du haut, du bas) qui décrivent chacun deux traces (haut haut et haut bas, bas haut et bas bas), deux bases de temps A et B, retardée et non retardée, appliquées au signal du haut ou au signal du bas, déclenchement sur A ou B ou les signaux permutés... Sans oublier le bruit du ventilateur et la chaleur de 500 à 700 W à supporter, ni leur poids à vous donner un lombago. Mais voilà, leur *look* d'enfer et les sentiments qu'ils suscitent sont au-delà de tout ce que les oscilloscopes actuels en plastique peuvent offrir. Hélas, on ne pourra bientôt plus les mettre en service pendant un bout de temps, puisqu'ils vont partir vers le nouvel entrepôt et les bureaux d'Elektor à Aix-La-Chapelle en Allemagne. S'il vous

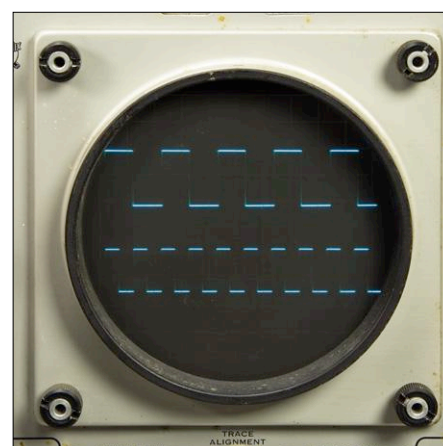


Figure 7. Les oscilloscopes Tektronix sont connus pour leurs traces extrêmement fines et leur excellente réponse en impulsions. Cette image photographiée sur un 565 avec un graticule filtrant à mailles fines permet de bien centrer le point. Notez que l'instrument n'était pas encore étalonné.

plaît, Monsieur le conducteur du chariot élévateur, faites-y bien attention ! ◀

(160402 - version française : Robert Grignard)

EST[®] 2004

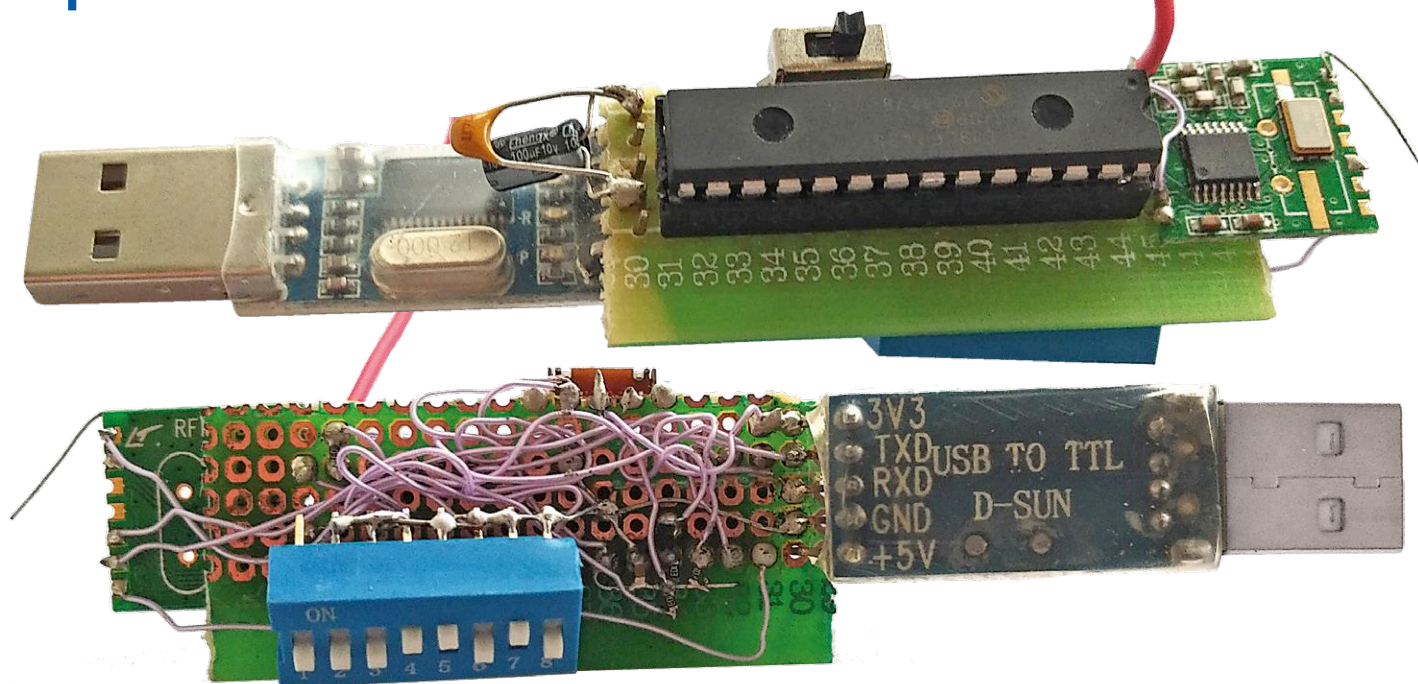
www.elektor.tv



Électronique 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

renifleur avec un module radio RFM12

espionnez l'éther



Laurent Labbe (Chine)

Il y a quelques années, j'ai conçu un thermomètre à affichage Nixie qui utilise des capteurs déportés. Le thermomètre affiche les températures intérieure et extérieure ainsi que l'heure donnée par un module GPS externe. Une application spécifique permet de surveiller une boîte aux lettres éloignée (comptage des passages du facteur). Tous les capteurs émettent (liaison sans fil) des données à destination de l'afficheur toutes les 30 mn. Pour diverses raisons, je ne peux pas connecter de PC au boîtier d'affichage pour surveiller l'arrivée des trames et m'occuper du débogage. C'est pourquoi j'ai eu recours au module radio RFM12 pour espionner mon thermomètre. Maintenant que les routines logicielles tournent, je partage mon expérience avec vous.

Pour être sûr de voir toutes les informations envoyées à l'afficheur du thermomètre, j'ai développé un outil simple : le renifleur RFM12. Il capte tout ce qui passe dans l'air et le transmet via une liaison USB-série à un PC qui affiche ces données. La première version de mon montage s'intéressait aux trames du thermomètre formatées selon un protocole propriétaire (simple), avec un entête et une somme de contrôle. Ici, le principe est encore plus élémentaire : afficher sur le PC tout

ce qui passe dans « l'éther », sans contrôle de la structure des données, ni vérification de somme de contrôle.

Deux ingrédients essentiels : un PIC et un module radio

Pour animer mon montage, j'ai choisi un microcontrôleur PIC 18LF2420 qui supporte le 3,3 V et une fréquence interne maximale de 8 MHz. Vous êtes libre d'utiliser n'importe quel

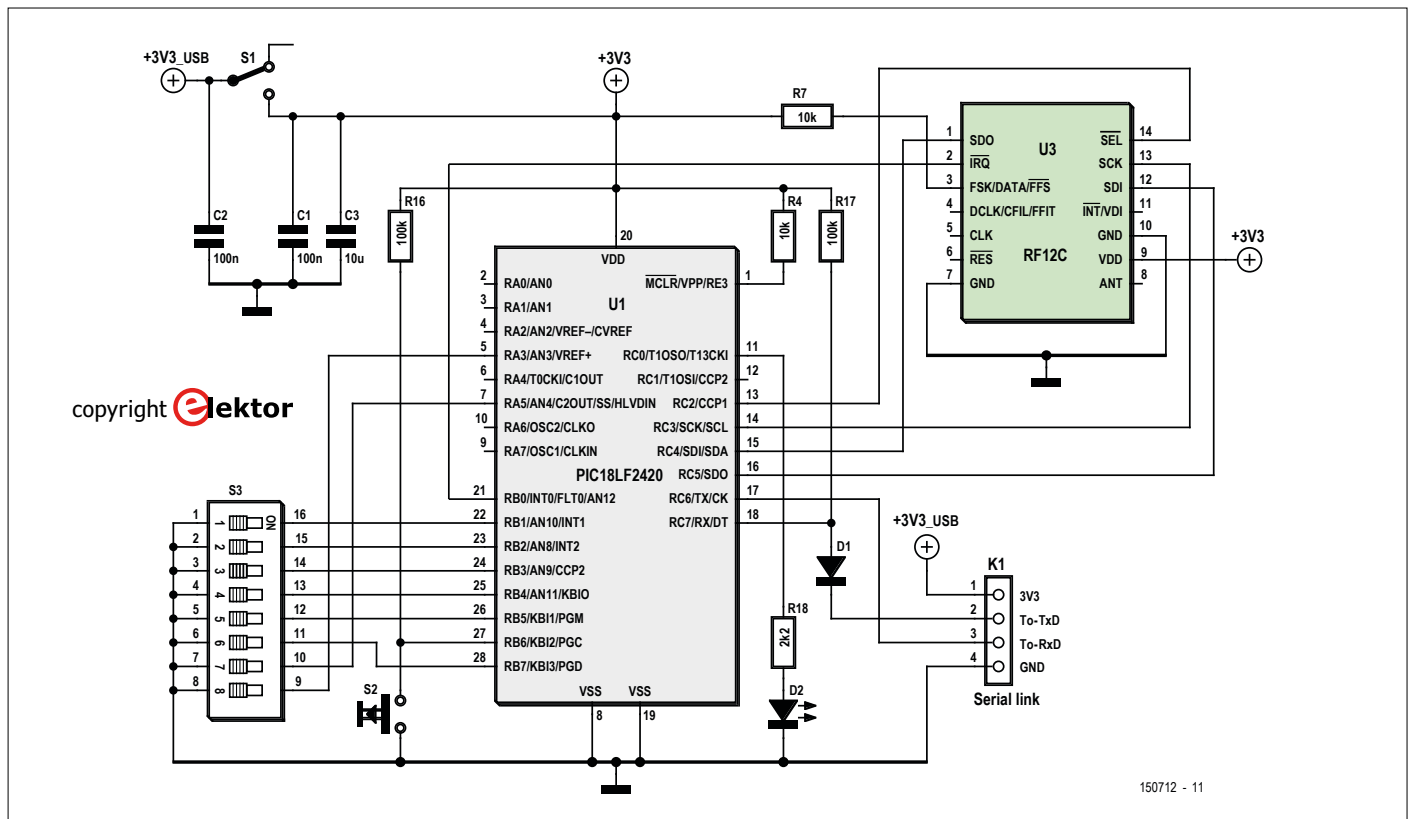


Figure 1. Schéma du renifleur RFM12.

autre microcontrôleur.

Pour l'alimentation, le module USB-série retenu possède deux sorties : 5 V et 3,3 V. Comme le RFM12 nécessite 3,3 V, nous utiliserons cette option après un filtrage correct. Pour information, j'ai fréquemment recours aux modules USB-série de la marque Prolific.

Pour l'adaptation de niveau entre le μC (3,3 V) et l'USB (5 V), la broche Tx (17) du μC est reliée directement au connecteur USB (K1) tandis que la broche Rx (18) est protégée par une diode et munie d'une résistance de rappel vers le haut.

Une rangée de commutateurs DIP (S3) permet de procéder à quelques réglages du circuit. Les deux premiers commutateurs définissent la vitesse de transmission vers le PC (voir **tableau 1**). Si le module RFM12 n'entend plus rien, les commutateurs des positions 3 et 4 fixent le délai à la fin duquel le PC reçoit un retour à la ligne automatique (CR/LF) (voir **tableau 2**).

Attention : on ne peut pas utiliser les positions 7 et 8. En effet ces commutateurs sont reliés au port A sans résistance de rappel vers le haut alors que le port A ne possède pas de résistances internes.

L'interrupteur S1 (marche/arrêt) sert surtout à recharger le micrologiciel sans avoir à retirer le μC . Le bouton-poussoir S2 permet de réinitialiser le logiciel.

Les résistances sont principalement des résistances de polarisation haute : R4 pour la RàZ du μC , R16 pour le bouton-poussoir S2 et R17 pour la liaison série sur le connecteur K1. La rangée de commutateurs DIP a recours aux résistances de rappel internes du port B. La LED D2 clignote lorsque des données arrivent.

Une pincée de code

Le code source est écrit et compilé avec MikroC (à télécharger en [2]). Les procédures RFM12 sont les plus délicates. J'ai adapté des routines trouvées sur l'internet. J'ai ajouté par ex. un chien de garde lorsqu'on attend des données. Ce code est une version allégée de celui de mon thermomètre Nixie. Certaines parties du code sont donc inutiles pour le renifleur présenté ici. La boucle principale se contente d'attendre la réception de données correctes : la broche de sortie /IRQ du module RFM12 annonce l'arrivée de données au μC (entrée PORTB.F0), ensuite

Tableau 1. Vitesse de transmission vers le PC (S3)

Pos. 1	Pos. 2	Pos. 1	Pos. 2	Pos. 1	Pos. 2	Pos. 1	Pos. 2
OFF	OFF	ON	OFF	OFF	ON	ON	ON
1200 bauds		9600 bauds		19200 bauds		19200 bauds	

Tableau 2. Durée du « silence » avant l'envoi d'un retour à la ligne (S3)

Pos. 3	Pos. 4	Pos. 3	Pos. 4	Pos. 3	Pos. 4	Pos. 3	Pos. 4
OFF	OFF	ON	OFF	OFF	ON	ON	ON
pas de CR/LF		1 s		10 s		20 s	

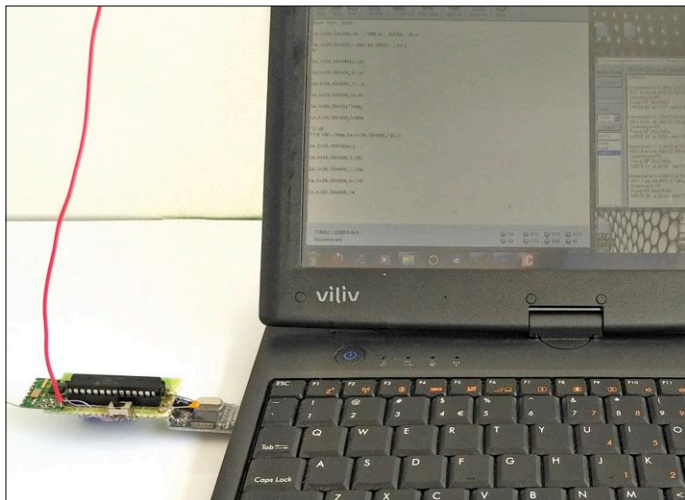


Figure 2. Espion en action : montage relié à un ordinateur portable.

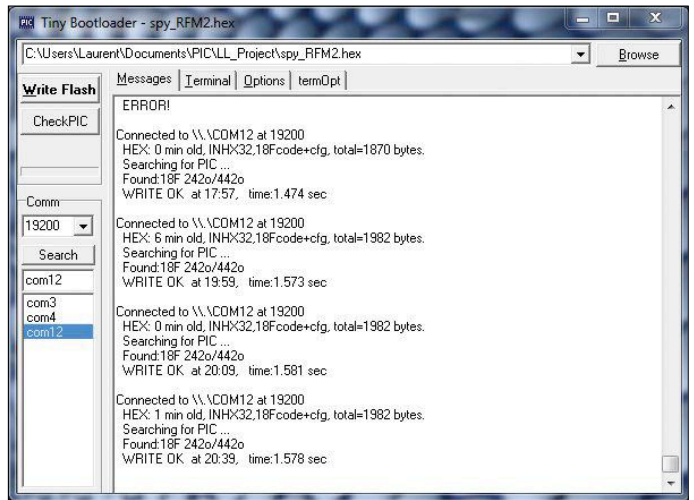


Figure 3. Capture d'écran du chargeur d'amorce.

les données sont envoyées au µC (PORTC.F4) pour être transmises au PC via la liaison série (PORTC.F6).

J'ai remarqué que le RFM12 envoie souvent des données « erronées ». Il faut donc régulièrement remettre à zéro le récepteur et le registre FIFO. Dans le programme original, c'est facile, car chaque message comporte un octet de longueur. Ici, j'utilise les commutateurs DIP 5 et 6 pour programmer une longueur (voir **tableau 3**) ; le récepteur et le registre sont automatiquement réinitialisés lorsque la longueur réglée est atteinte. Sans cela, le RFM12 continue de recevoir des données après chaque trame.

Habituellement, j'utilise un chargeur d'amorce pour toutes mes applications avec le 18LF2420. Pour moi, le plus pratique est *Tiny Pic Bootloader* (voir [3]). Il n'est pas compatible avec tous les µC, mais l'est avec le 18LF2420. Comme le 18LF2420 ne peut pas changer de manière dynamique la fréquence du chien de garde, vous devrez peut-être recompilier le chargeur pour adapter cette fréquence. Le code embarqué occupe environ 100 octets et son fonctionnement est fiable.

J'écoute

L'utilisation du montage n'a rien de sorcier : connectez-le au port USB d'un PC. Il faudra avoir au préalable installé le pilote fourni par le fabricant du module USB-série choisi. Ensuite lancez un outil comme *HyperTerminal* ou *CoolTerm* et réglez la bonne vitesse. Des données devraient s'afficher. Si le *timer* (tableau 2) est activé, les retours à la ligne améliorent la lisibilité. Avec la fonction « enregistrer », vous pouvez laisser le renifleur fonctionner pendant plusieurs jours et analyser les trames capturées a posteriori. Avec *CoolTerm*, l'affichage des caractères réels est de temps en temps figé, mais si vous examinez les données en hexa, tout est là. ◀

(150712)

Liens

- [1] [Projet dans le labo d'Elektor : www.elektormagazine.fr/labs/sniffer-for-rfm12-display-exchanges-over-the-air-on-pc](http://www.elektormagazine.fr/labs/sniffer-for-rfm12-display-exchanges-over-the-air-on-pc)
- [2] [Code source, code compilé, proposition de circuit imprimé : www.elektormagazine.fr/150712](http://www.elektormagazine.fr/150712)
- [3] [Chargeur d'amorce Tiny Pic Bootloader : www.etc.ugal.ro/cchiculita/software/picbootloader.htm](http://www.etc.ugal.ro/cchiculita/software/picbootloader.htm)

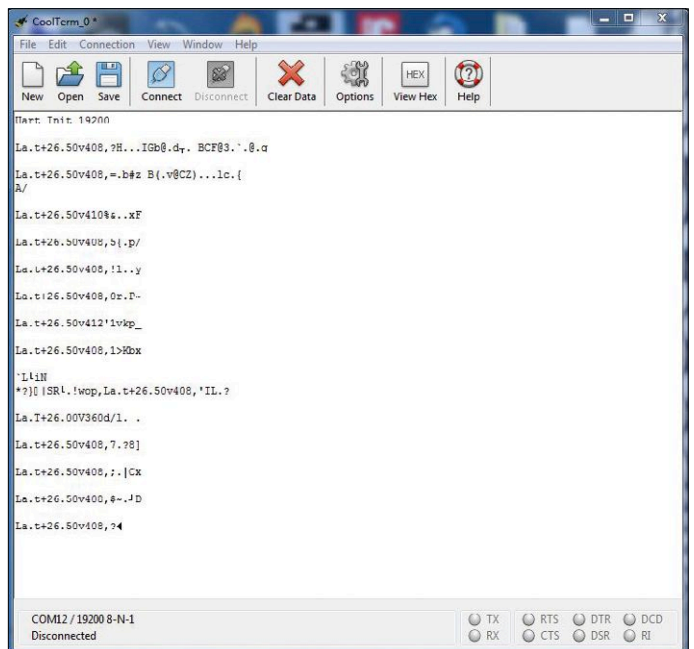


Figure 4. Capture d'écran d'un terminal.

Tableau 3. Longueur de la trame attendue (S3)

Pos. 5	Pos. 6	Pos. 5	Pos. 6	Pos. 5	Pos. 6	Pos. 5	Pos. 6
OFF	OFF	ON	OFF	OFF	ON	ON	ON
10 octets		20 octets		40 octets		100 octets	

Micro-carte

79 € **
au lieu de
~~127,50 €~~

Votre cadeau de bienvenue : Raspberry Pi Zero W***

- ARM11 cadencé à 1 GHz
- 512 Mo de RAM
- Port mini-HDMI
- 2× port micro-USB
- Connecteur à 40 broches, compatible HAT
- Connecteurs pour vidéo composite et RàZ
- Connecteur série pour caméra

- **6x le magazine en version papier**
- **6x le magazine en version PDF**
- **Accès aux archives en ligne d'Elektor, plus de 25 ans de projets**
- **10 % de remise dans l'e-choppe**

www.elektormagazine.fr/offre
votre code promotionnel : ZERO17

*** Offre réservée aux nouveaux abonnés**
(c.-à-d. toute personne ou entité qui n'a pas été membre de la communauté d'Elektor au cours des 24 derniers mois)
**** Prix TTC valable uniquement pour la France métropolitaine.**
Nous consulter pour les tarifs TTC hors France métropolitaine.
***** Dans la limite des stocks disponibles !**

nouvelle formule !

mai / juin 2017 | n° 465
www.electormagazine.fr

elektor

DÉCOUVRIR CRÉER PARTAGER

radiocommande XBee/nRF24

bidirectionnelle à 4 voies

Dans cette édition :
5 projets de labo
6 projets de lecteur
Bluetooth, L-board,
SPI, Li-Ion, PC,
XBee ...

prototypage Bluetooth : module PSoC BLE au format L-board • Phantatron • analyseur de signal par Ethernet : nouveaux logiciels • Sigfox : donner vie aux objets • éclairage extérieur variable (lampes à LED de 12 V) • transformateur de Tesla • simulateur de présence • le bus I²C • chargeur d'accus Li-Ion • utilisation de l'ec-stencil-fix • hors-circuits de R. Lacoste : corrélation numérique • Android sur Raspberry Pi (2) • Rétronique : PET fête ses 40 ans • WaterDeck : analyse de la consommation d'eau • loi sur les robots ? • passerelle IoT et nœuds sans fil (2)

BBC micro:bit pour électroniciens
premiers pas avec mbed

cartes et kits de développement
pour l'Internet des Objets

préamplificateur pour microphones professionnels

ISSN 0181-7450

(B) 16,50 € • (D) 20,00 € • (S) 20,00 € • (S) 20,00 € • (S) 20,00 €
(D) 16,50 € • (D) 20,00 € • (S) 20,00 € • (S) 20,00 € • (S) 20,00 €

des condensateurs qui se rechargent tout seuls !

attention à l'absorption diélectrique

Robert Lacoste (Chaville)

De temps en temps, les composants électroniques peuvent être taquins. Par exemple, il n'y a rien de plus banal qu'un condensateur, non ? Vous savez tous qu'un condensateur est un accumulateur de charges électriques. Si on lui injecte un courant, il se charge ; ensuite si on connecte une résistance à ses bornes, il se décharge. Eh bien ! Détrompez-vous, un condensateur est un peu plus subtil qu'il n'y paraît. Je traiterai ici d'un phénomène assez surprenant, appelé absorption diélectrique. En deux mots, vous allez voir qu'un condensateur peut se recharger tout seul !



Une expérience toute simple

Vous ne me croyez pas ? Commençons par une expérience. Allez dans votre garage, votre grenier, enfin là où vous bricolez, et dégotez les éléments suivants : un gros condensateur électrolytique (par ex. un modèle de 2 200 μ F fera parfaitement l'affaire, de la moins bonne qualité possible), une résistance de faible valeur (allez, disons 100 Ω), une alimentation continue (dans les 10 V, mais ce n'est pas crucial, une batterie ou une pile conviendra aussi), deux interrupteurs et quelques fils. Ah j'oubliais, un petit oscilloscope sera aussi utile. Si vous n'en possédez pas, pas de panique, vous pouvez également utiliser un multimètre numérique, un crayon et du papier, car le phénomène que nous allons étudier est lent. Veillez juste à

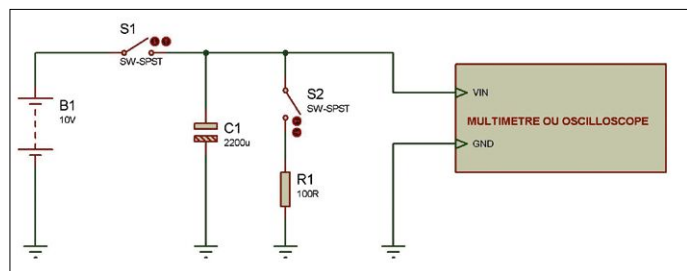


Figure 1. Pour expérimenter l'absorption diélectrique, il suffit d'un condensateur, une résistance, des fils, deux interrupteurs et un instrument de mesure de tension.

ce que le multimètre ait une impédance d'entrée assez élevée (au moins 1 M Ω , le mieux 10 M Ω).

Voici comment brancher tout ça (**figure 1**) : connectez l'oscilloscope ou le multimètre aux bornes du condensateur. Connectez l'alimentation au condensateur via le premier interrupteur (S1) et connectez enfin la résistance de décharge de 100 Ω au condensateur via le deuxième interrupteur (S2). Les deux commutateurs doivent être initialement ouverts. La photo vous montre la *manip* sur ma paillasse (figure 8), rien de bien compliqué.

Maintenant, basculez S1 : le condensateur se charge très vite, la tension à ses bornes devient rapidement égale à la tension d'alimentation, c'est normal. Démarrez l'enregistrement de la tension sur votre oscilloscope, en utilisant une base de temps lente, de 10 s par division ou plus. Si vous utilisez un multimètre, commencez à noter la tension mesurée sur une feuille de papier. Attendez une dizaine de secondes, et débranchez l'alimentation en ouvrant S1. La tension aux bornes du condensateur reste à peu près constante puisque celui-ci est en circuit ouvert, enfin sous réserve que ses pertes soient raisonnablement faibles. Maintenant basculez S2. Cela va provoquer la décharge du condensateur à travers la résistance de 100 Ω , et la tension à ses bornes sera rapidement très faible. La durée requise pour une décharge quasi complète dépend des valeurs du condensateur et de la résistance, mais avec les valeurs proposées de 2 200 μ F et 100 Ω , le calcul montre que cette tension sera inférieure à un millivolt après deux secondes.

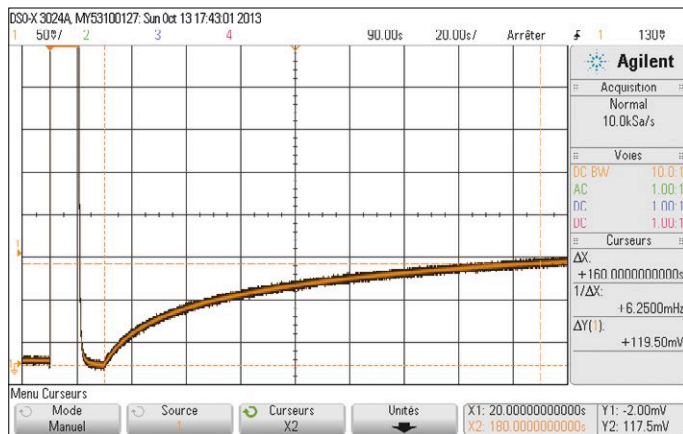


Figure 2. Le résultat d'un essai avec un condensateur de 2 200 μF , et une décharge dans 100 Ω pendant 10 s. Après deux minutes, la tension remonte à 119 mV à cause de l'absorption diélectrique.

Pour être certain que le condensateur est bien déchargé, laissez donc S2 fermé pendant dix secondes. Maintenant, passons à la partie amusante : après ces dix secondes, ouvrez S2, et attendez. Le condensateur a été complètement déchargé et est maintenant en circuit ouvert. Mais devant vos yeux ébahis, vous découvrirez que la tension à ses bornes augmente lentement, selon une courbe exponentielle ! La **figure 2** montre la courbe que j'ai obtenue avec ces valeurs de composants. Avec le condensateur que j'ai utilisé, la tension monte jusqu'à environ 120 mV en deux minutes, comme si le condensateur avait été rechargé par une source de tension. Qu'est-ce qui se passe ici ? Aucun phénomène paranormal, vous venez d'expérimenter l'absorption diélectrique [1] !

Variantes expérimentales

Avant d'expliquer la cause de la chose, quelques autres expériences peuvent être utiles. Vous pouvez penser que la valeur de la résistance était trop élevée pour une décharge complète. Essayez donc avec une autre résistance. J'ai effectué le test pour vous et j'ai remplacé la résistance de 100 Ω par une résistance de 1,5 Ω . Toujours avec la même durée de décharge (10 s), le résultat est illustré sur la **figure 3**. Comme prévu, la décharge est nettement plus rapide en raison de la

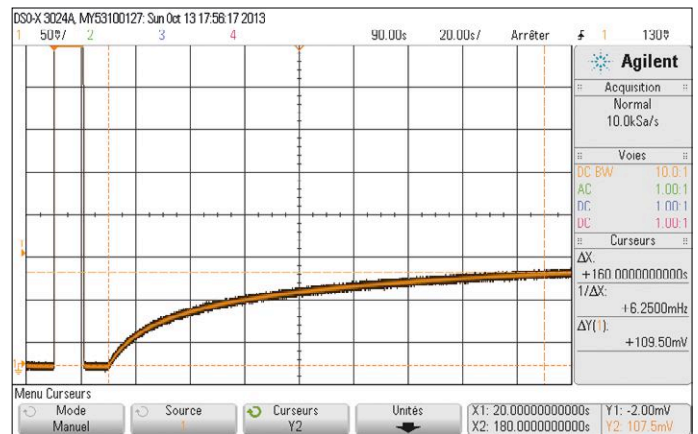


Figure 3. Baisser la résistance de décharge de 100 Ω à 1,5 Ω ne modifie pas significativement le comportement par rapport à la figure 2.

résistance de décharge de plus faible valeur, mais après deux minutes, la tension à travers le condensateur est de 110 mV, très proche des 120 mV obtenus lors du premier test. Donc, la valeur de la résistance de décharge ne semble pas avoir une grande influence.

J'ai donc remis la résistance de 100 Ω , et plutôt que de réduire sa valeur, j'ai augmenté la durée de la décharge. Fait intéressant, si S2 est maintenu fermé pendant 20 s au lieu de 10 s, la tension aux bornes du condensateur après deux minutes ne remonte qu'à 65 mV (**figure 4**), environ la moitié de la tension obtenue dans le test initial. Eh bien, il semble donc qu'il y ait un lien étroit entre la durée de décharge et cette tension fantôme...

Le dernier test intéressant est d'essayer d'autres condensateurs. J'ai testé un tas de condensateurs électrolytiques ou même au tantale. Bien sûr, l'échelle de temps et la valeur de la résistance de décharge doivent être adaptées en fonction de la capacité du condensateur, mais on observe toujours plus ou moins le même comportement. Un exemple intéressant est illustré sur la **figure 5**, avec cette fois un petit condensateur électrolytique de 4,7 μF / 63 V. Après une charge de 5 s et une décharge de 2 s sur 100 Ω , la tension à travers le condensateur a augmenté jusqu'à 83 mV en 12 s, puis a

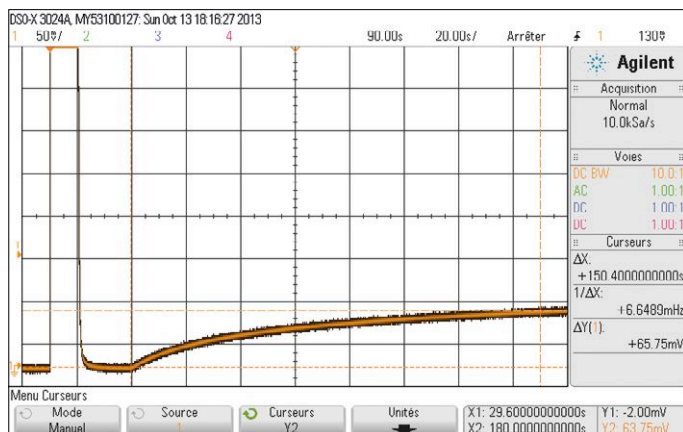


Figure 4. Ce tracé a été enregistré avec une résistance de 100 Ω , mais avec une durée de décharge doublée, 20 s. La tension finale est approximativement divisée par deux par rapport à la configuration initiale.

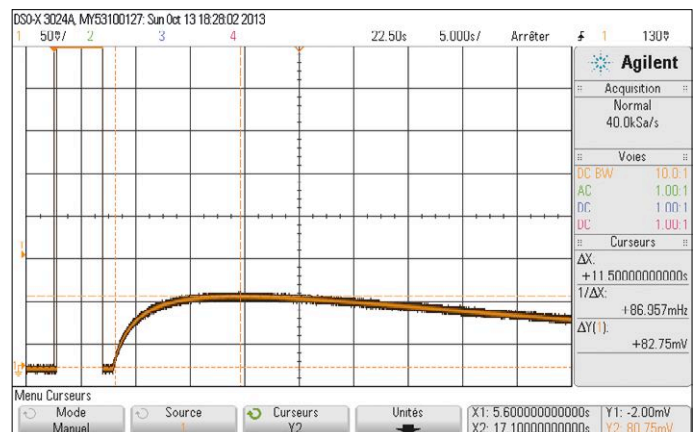


Figure 5. Avec un condensateur électrolytique de 4,7 μF et une décharge de 2 s dans 100 Ω , la tension remonte à 83 mV en 12 s, puis diminue lentement en raison des pertes.

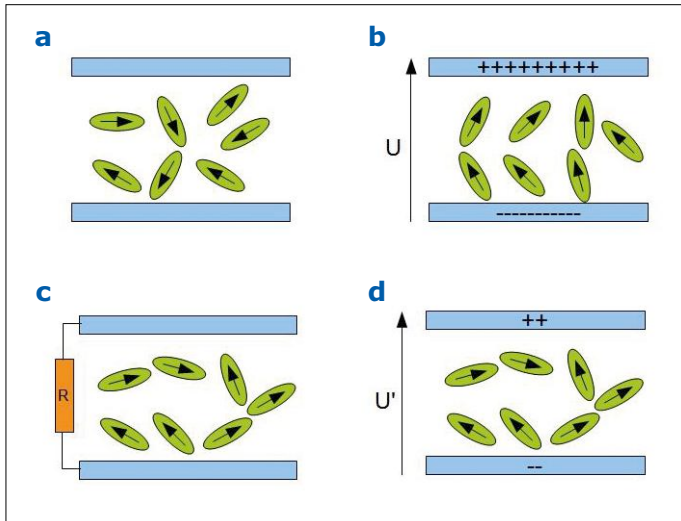


Figure 6. a) Les molécules du diélectrique sont orientées de manière aléatoire. b) Lorsqu'un champ électrique est appliqué entre les plans du condensateur, les molécules s'orientent dans la direction du champ. c) La décharge du condensateur libère l'orientation des molécules, mais cela prend du temps. d) Si la décharge s'arrête trop tôt alors que les molécules sont encore orientées, cette orientation se propage vers les plans, ce qui recrée une tension entre les bornes du condensateur.

diminué lentement. Cette décroissance était évidemment due à la décharge interne de ce condensateur, combinée au faible courant d'entrée de l'oscilloscope. Ensuite, j'ai remplacé le condensateur électrolytique par un condensateur céramique de même valeur (4,7 μF). L'augmentation de la tension fantôme a été considérablement plus faible, moins de 20 mV.

Résumons nos découvertes : avec certains types de condensateurs, principalement les condensateurs dits « chimiques » (électrolytiques ou au tantale), le condensateur ne semble pas complètement déchargé après un cycle de décharge rapide. Plus précisément, il semble se recharger de manière

autonome, mais avec une échelle de temps lente (plusieurs secondes, voire des minutes). La tension finale dépend du type du condensateur, de la tension initiale et de la durée de décharge, mais peu de la valeur de la résistance de décharge. Comme ordre de grandeur, cette tension finale était dans mes tests de l'ordre de 1 ou 2% de la tension de charge initiale.

Des explications !

Pour comprendre d'où vient cette absorption diélectrique, il faut passer de l'électronique à la chimie. Je suis sûr que vous savez qu'un condensateur est constitué de deux plans conducteurs parallèles, entre lesquels est présent un champ électrique. Entre ces deux plans, un matériau diélectrique assure l'isolation, et ce diélectrique est à l'origine du phénomène. En deux mots, ce diélectrique se compose de molécules « collées » entre elles par des liaisons chimiques. La grande majorité des molécules présentent une polarisation électrique, ce qui signifie que leurs électrons sont situés plus près d'une extrémité de la molécule que de l'autre. Lorsqu'un champ électrique est appliqué entre les plaques du condensateur, les molécules du diélectrique tendent à s'aligner sur ce champ électrique. Cependant, ce mouvement est ralenti par les liens entre les molécules et leurs voisines. Cela a deux conséquences : premièrement, il y a de l'énergie dissipée dans le diélectrique pour aligner ces molécules, ce qui entraîne la perte dite diélectrique du condensateur. Deuxièmement, cet alignement prendra du temps à s'installer (**figure 6**). Lorsque le condensateur est déchargé, le champ électrique est annulé, de sorte que les molécules se relaxent et retournent à leur état non orienté, mais cela prendra aussi du temps. Si le condensateur est déchargé rapidement, une grande partie des molécules sera encore polarisée à la fin de la décharge, et leur polarisation se propagera ensuite lentement de proche en proche jusqu'aux plaques du condensateur. C'est ce que nous venons de voir expérimentalement !

Ceci permet de comprendre pourquoi l'autorecharge mesurée dépend de la durée de la décharge et, bien sûr, du type de diélectrique. Évidemment, un condensateur à air, c'est-à-dire sans diélectrique, ne souffre pas de ce défaut.

Mon but ici est juste de vous donner un aperçu de la chose, si cela vous intéresse vous trouverez une analyse très complète de ce phénomène dans un superbe article de Bob Pease disponible en ligne (voir [2]). Cet article explique en particulier que le modèle équivalent d'un condensateur avec effet d'absorption diélectrique est un grand condensateur en parallèle avec plusieurs petits condensateurs, chacun en série avec des résistances d'assez forte valeur. J'ai procédé pour vous à une petite simulation d'un tel circuit. Regardez le résultat sur la **figure 7**, c'est assez proche des résultats expérimentaux, non ? J'ai réalisé cette simulation sous Proteus (Labcenter), mais vous pouvez bien sûr utiliser n'importe quel simulateur de type Spice si vous souhaitez tester vous-même.

Pour ceux qui ne le connaissent pas (mais est-ce possible ???), Bob Pease, décédé en 2011, était l'un des fondateurs et gourous de *National Semiconductors*, maintenant *Texas Instruments*. Il a laissé un tas de publications fantastiques et passionnantes. Texas a eu la bonne idée de publier tous ses articles sur le web, je vous encourage très sincèrement à y fureter, c'est une extraordinaire source d'inspiration !

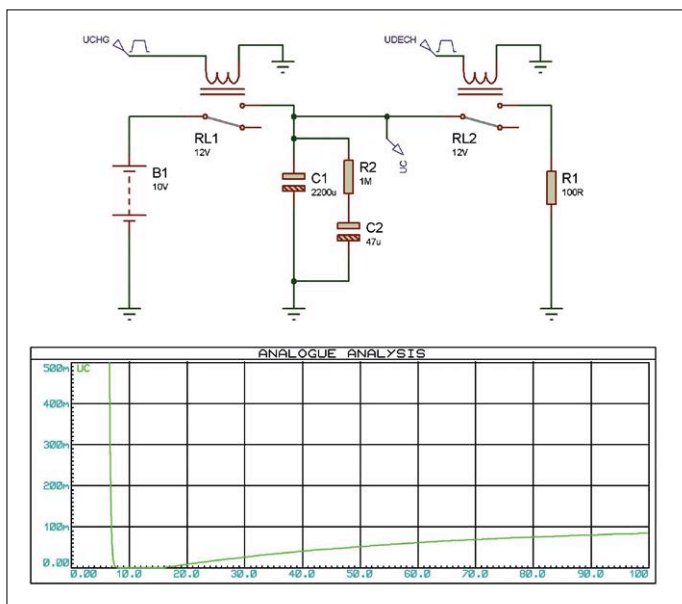


Figure 7. L'absorption diélectrique peut être modélisée par un petit condensateur en série avec une grande résistance, connectée sur le condensateur principal. Cette petite simulation sous Proteus [6] montre que le comportement ainsi modélisé est très proche de l'expérience.

Revenons à l'absorption diélectrique. Le problème est suffisamment important pour qu'il existe une méthode

standardisée pour le mesurer. Lancez votre moteur de recherche favori et recherchez la norme militaire MIL-PRF-19978 [4]. C'est grosso modo la même méthode que celle que j'ai utilisée dans mes exemples, mais avec des valeurs standardisées. Le condensateur doit d'abord être chargé pendant une heure à sa tension nominale, puis déchargé pendant dix secondes via une résistance de 5 Ω . La tension est à nouveau mesurée au cours des quinze minutes suivantes, avec un millivoltmètre à très haute impédance (plus de 10 000 M Ω ...). Le coefficient d'absorption diélectrique du condensateur est le rapport entre la tension maximale retrouvée et la tension initiale, exprimé en pourcentage. Les fabricants de condensateurs communiquent ces coefficients d'absorption diélectrique pour chaque technologie. Le **tableau 1**, extrait de Wikipédia, vous donne quelques valeurs typiques.

Pour conclure

Ce phénomène est-il réellement un souci ? Imaginez que vous devez réparer un téléviseur cathodique ou tout autre bidule plein de très hautes tensions. Vous le débranchez avec précaution du secteur et vous l'ouvrez. Bien sûr, vous savez que vous devez décharger tous les condensateurs avant de mettre vos mains pleines de doigts partout. Donc, vous prenez une résistance de décharge, et vous l'appliquez sur ce grand condensateur de 10 kV / 1 μ F pendant 5 ou 10 s pour être en tranquille. Ensuite vous mettez le doigt dessus. Grosse erreur ! La tension peut avoir remonté à quelques centaines de volts grâce à l'absorption diélectrique. Même des condensateurs d'alimentation secteur chargés sous 400 V peuvent récupérer une tension suffisamment élevée pour être dangereuse ou, plus fréquemment, pour détruire un circuit sensible.

L'absorption diélectrique est aussi une grosse source d'ennuis si vous devez concevoir des circuits analogiques de précision, comme ce qu'on appelle un échantillonneur-bloqueur. Dans un tel circuit, la tension d'entrée est rapidement appliquée à un condensateur de stockage, qui est ensuite mis en circuit ouvert. À cause de l'absorption diélectrique, le condensateur récupérera inévitablement une partie de sa tension précédente. Cela peut fausser la mesure ou, pire, rendre la tension dépendante de l'ensemble des valeurs précédentes. Pour de telles applications, des condensateurs spécifiques à faible absorption diélectrique sont donc indispensables. Dans le passé, on utilisait pour ces applications des condensateurs au polystyrène, au polypropylène ou même au Teflon. Aujourd'hui il est vrai qu'un échantillonneur-bloqueur est en général proposé sous la forme d'un circuit intégré et que le condensateur de stockage est interne à la puce, mais celui-ci est bien sûr optimisé pour cette tâche par son fabricant.

Un mot enfin pour nos amis audiophiles. Si j'avoue me moquer de temps en temps de ceux qui dépensent plusieurs centaines d'euros pour un câble secteur ou une fibre optique « spécial audio », car cela s'approche pour moi du spiritisme, il est indubitable que le choix des condensateurs doit faire l'objet d'une grande attention pour des circuits audio de qualité. Vous en connaissez maintenant l'une des raisons : l'absorption diélectrique. Ce phénomène peut en effet introduire une distorsion, qui plus est dépendante de la fréquence et de l'amplitude du signal. C'est pourquoi deux condensateurs de 1 μ F ne se valent pas pour ce type d'application. Encore une fois, reportez-vous à l'article de Bob Pease pour plus d'information sur ce sujet.

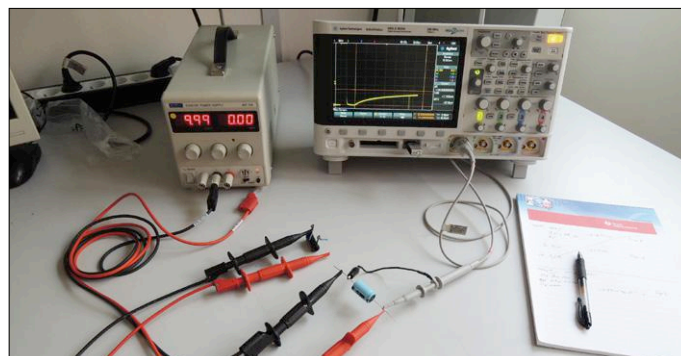


Figure 8. Voici ma configuration de test. Rien d'extraordinaire, sauf peut-être un bel oscilloscope Keysight DSO-X 3024A [4], beaucoup trop puissant pour une telle expérience !

Type de condensateur	Absorption diélectrique
Air	non mesurable
Céramique classe 1 (NP0)	0,6%
Céramique classe 2 (X7R)	2,5%
Polypropylène	0,05 à 0,1%
Film polyester	0,2 à 0,5%
Électrochimique, tantale	2 à 3%
Électrochimique, aluminium	10 à 15%

Tableau 1. Ce tableau, extrait de Wikipédia, donne les facteurs d'absorption diélectrique des familles de condensateurs classiques. Pour les électrolytiques, l'autorecharge peut aller jusqu'à 10-15% de la tension initiale !

Nous y voilà. J'espère que cet article vous a intéressé, car c'est un sujet souvent oublié ou mal compris. En tout cas, n'hésitez pas à essayer par vous-même, surtout que pour une fois c'est vraiment une expérience à la portée de tout le monde ! Attendez-vous à d'autres surprises, car je vais continuer sur la même lignée dans mes prochains articles. Nous verrons qu'un condensateur peut avoir une capacité variant de 1 à 10 selon la tension appliquée. Nous verrons aussi qu'il peut très bien ne plus être un condensateur, mais une inductance, selon la fréquence de travail. Décidément, les composants simples ne sont plus ce qu'ils étaient ! ◀

(160500)

Cet article a été publié dans la revue Circuit Cellar (n°283, février 2014).

Liens et ressources

- [1] Absorption diélectrique : http://en.wikipedia.org/wiki/Dielectric_absorption
- [2] Article de Bob Pease : <http://electronicdesign.com/analog/whats-all-soakage-stuff-anyhow>
- [3] Site de Bob Pease : www.ti.com/ww/en/bobpease/assets/www-national-com_rap.pdf
- [4] MIL-PRF-19978 : www.dscc.dla.mil/Downloads/MilSpec/Docs/MIL-PRF-19978/prf19978.pdf
- [5] Oscilloscope Keysight DSO-X 3024A : www.keysight.com/en/pdx-x201839-pn-DSOX3024A/oscilloscope-200-mhz-4-channels?cc=FR&lc=fre
- [6] Simulateur Proteus VSM : www.labcenter.co.uk

MC sur SUPRA 2.0

tout petit signal, mais grande qualité

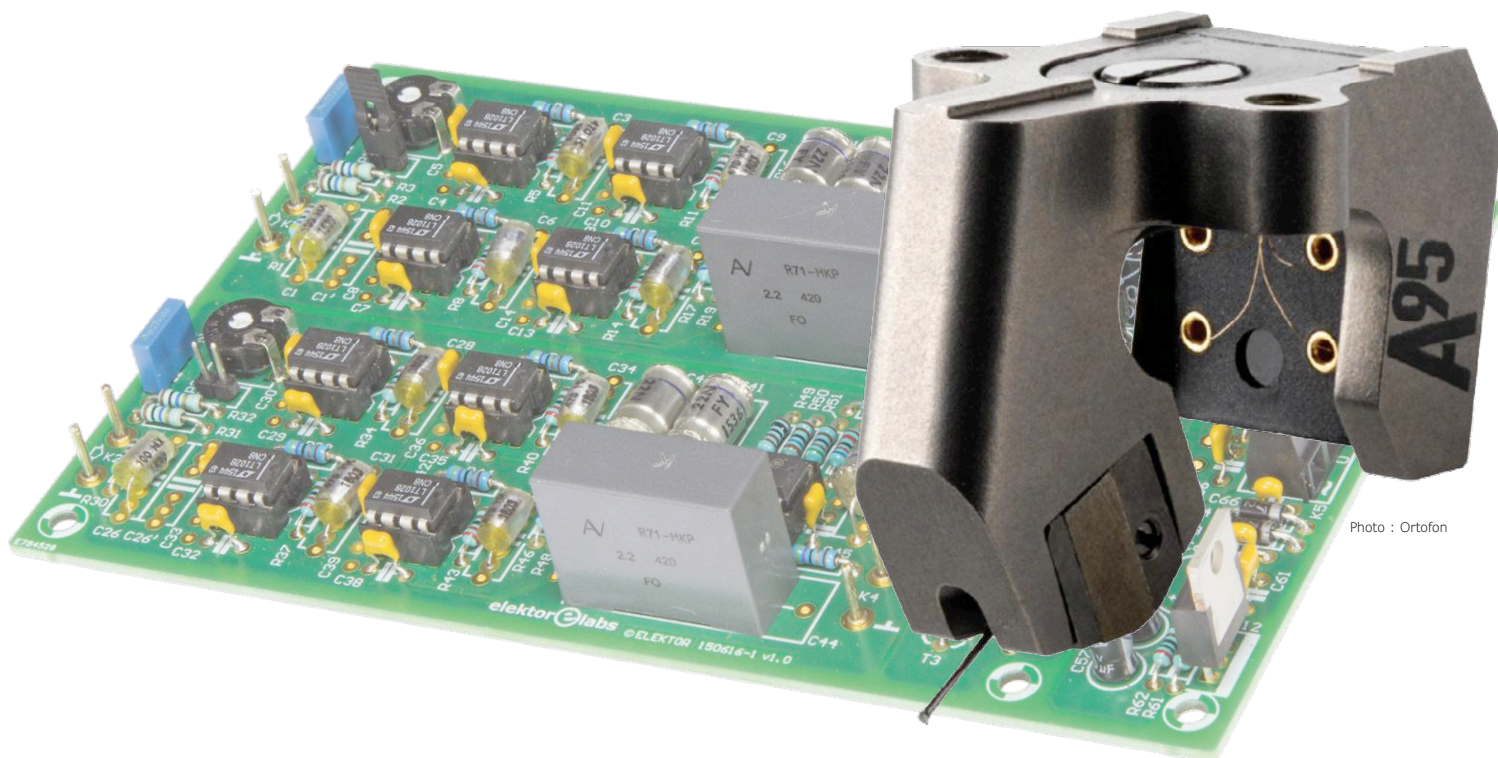


Photo : Ortofon

Une cellule MC est un transducteur à bobines mobiles d'une grande fidélité, mais à très bas niveau de tension. Le préampli phono SUPRA 2.0 à bruit minimal, avec ses quatre amplis op en parallèle, publié en juin 2016, est tout désigné pour traiter ce genre de signaux et de nombreux audiophiles nous ont demandé comment l'adapter à la cellule MC. C'est très simple, nous vous montrons comment faire.

Ton Giesberts (labo d'Elektor)

SUPRA 2.0, préamplificateur phono MM/MD [1] a donné la puce à l'oreille de nombreux fervents du son « vinyle », ils ont tenté l'aventure de construire ce préampli d'exception. Nombre d'entre eux ont même acheté le kit malgré le prix sérieux des composants de haute qualité qu'il contient.

Beaucoup d'audiophiles attendaient impatiemment la possibilité de raccorder une cellule « *moving coil* » au SUPRA 2.0. Ces cellules MC sont généralement supérieures à leurs homologues MM/MD, mais

nécessitent un préamplificateur spécial à plus haut gain, puisque leur tension de sortie se situe entre 0,2 et 0,5 mV. Au départ, le SUPRA était conçu pour ce genre de cellule aussi, mais à l'approche de la date de publication du nouveau SUPRA 2.0, nous avons concentré notre attention sur son usage sur MM/MD. Depuis lors, nous avons modifié un exemplaire au labo pour augmenter son coefficient d'amplification. Et puis, nous avons effectué les mesures nécessaires.

Les modifications

Les changements à opérer sont effectivement très simples. Sur un SUPRA 2.0 déjà assemblé, il n'y a par canal que neuf résistances concernées. La **figure 1**

montre la section du schéma où des modifications sont nécessaires. La plupart des cellules MC briguent une impédance de bouclage nominale d'environ 100 Ω . Pour les satisfaire, il faut échanger les résistances R1 et R30 qui fixent l'impédance d'entrée du préampli contre des 100 Ω . Mais si votre cellule a une impédance de 47 Ω par exemple, donnez-leur la même valeur.

Pour la capacité d'entrée du préampli, ce sont les condensateurs C1 et C26 qui sont déterminants. Pour les cellules MM/MD, vous pouvez avec ces capacités influencer la réponse aux hautes fréquences, mais avec une MC, ils n'ont que peu d'effet. Elle a en réalité une

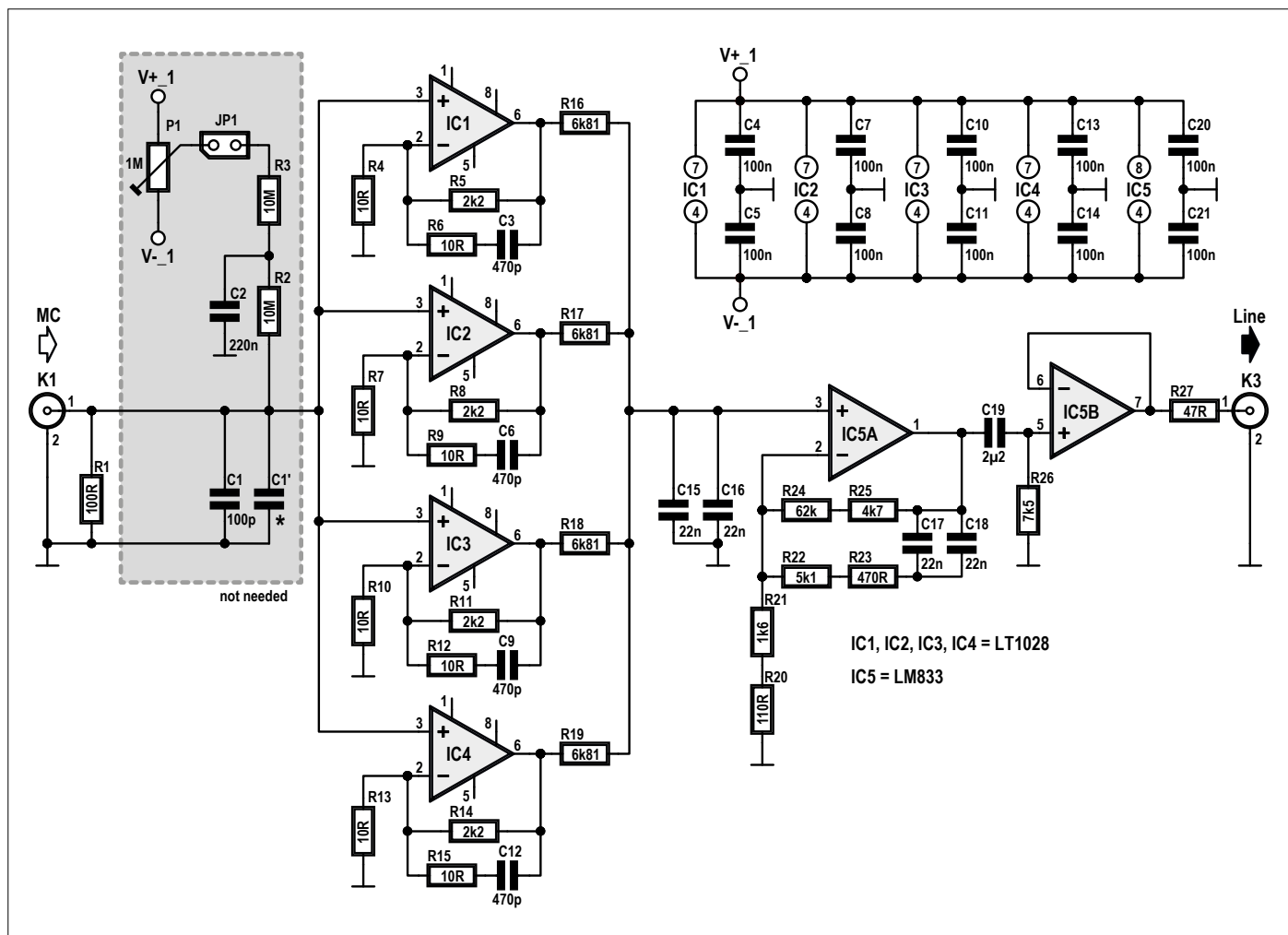


Figure 1. Schéma d'un canal de SUPRA 2.0 avec les valeurs modifiées des résistances pour l'utiliser avec une cellule MC.

résistance interne de quelques ohms et une inductance négligeable à cause du faible nombre de spires. Vous pouvez donc ôter ces condensateurs du circuit ou évidemment ne pas les installer lors de la construction, mais s'ils y sont, vous pouvez aussi les laisser.

Ensuite, les seize résistances de 47 Ω dans la rétroaction de IC1 à IC4 et IC6 à IC9, à savoir R4, R6, R7, R9, R10, R12, R13, R15, R33, R35, R36, R38, R39, R41, R42 et R44 doivent être remplacées par des 10 Ω . Du coup, l'amplification totale du préampli grimpe à 900 fois, dont les quatre amplificateurs opérationnels en parallèle, avec 221 fois, se taillent la part du lion.

Pour déterminer le gain nécessaire, nous avons pris la tension de sortie moyenne de vingt-six cellules MC, avec comme résultat un peu plus de 0,3 mV. Comme l'entrée de ligne de la plupart des amplificateurs audio a une sensibilité de 100 ou 200 mV, il nous a paru raisonnable de considérer un niveau d'entrée qui ne

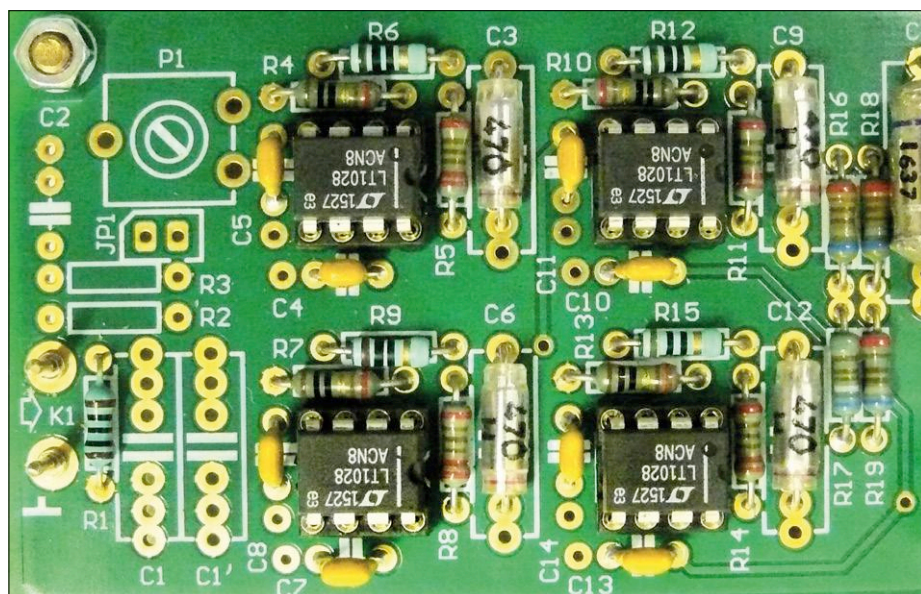
s'en éloigne pas trop, soit 250 mV pour ne pas pousser le gain inutilement fort. La diminution de la tension de sortie nominale par rapport à l'ancienne version (elle était de 468 mV pour 2,5 mV d'entrée) sert aussi à obtenir sur les quatre amplis op en parallèle une largeur de

bande suffisante malgré le grand facteur d'amplification de la version MC. La largeur de bande en boucle ouverte du LT1028 est de 70 MHz. Avec un gain de 221, la bande passante atteint un peu plus de 300 kHz. Le condensateur de 470 pF dans la rétroaction la réduit

Caractéristiques techniques et mesures

Mesures prises avec un signal d'entrée de 0,3 mV (impédance de source 5 Ω)
Valeurs mesurées avec alimentation de labo sous tension de ± 14 V.

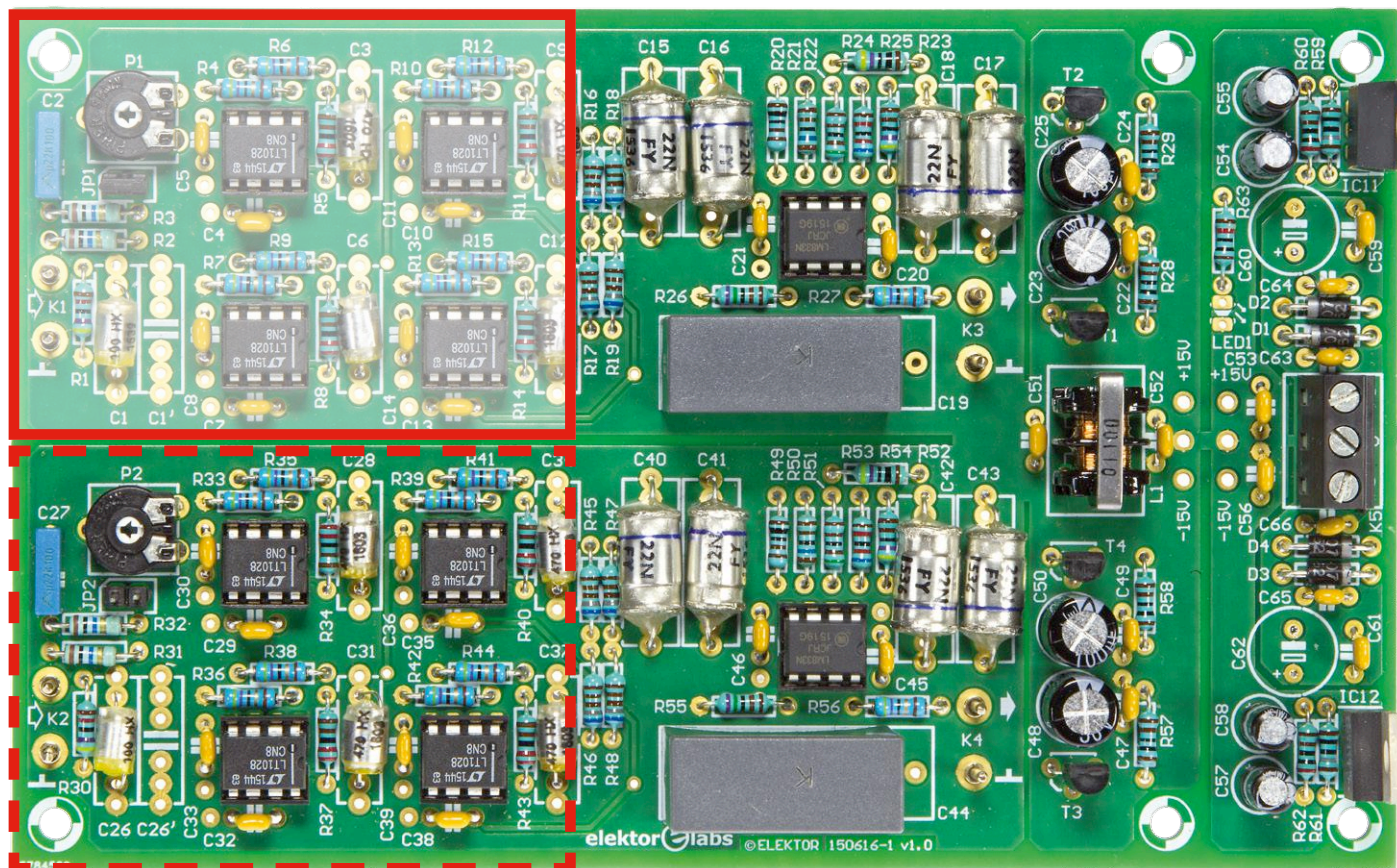
- Impédance d'entrée : 100 Ω
- Niveau de sortie pour 0,3 mV en entrée : 270 mV
- Rapport signal/bruit
 - linéaire : > 73 dB (B = 22 Hz à 22 kHz)
 - pondération A : > 79 dBA (B = 22 Hz à 22 kHz)
- DHT+b (à 1 kHz) : < 0,02 % (bruit uniquement, B = 80 kHz)
- DHT à 1 kHz : < 0,0007 % (par mesure FFT pondérée)
- DHT à 10 kHz : < 0,0002 % (par mesure FFT pondérée)
- Écart avec la courbe RIAA :
 - < 0,1 dB (50 Hz à 10 kHz)
 - < 1 dB (20 Hz)
 - < 0,15 dB (20 kHz)



Résumons toutes les modifications

- R1, R30 = 100 Ω /1 % (selon la cellule MC utilisée)
- R4, R6, R7, R9, R10, R12, R13, R15, R33, R35, R36, R38, R39, R41, R42, R44 = 10 Ω /1 %
- C1, C26 = inutiles, mais ne nuisent pas si déjà en place
- P1/R2/R3/C2/JP1 et P2/R31/R32/C27/JP2 = inutiles, mais ne nuisent pas si déjà en place.

Figure 2. La région du circuit imprimé où effectuer les modifications.



à 150 kHz dans l'étage d'entrée, ce qui donne une largeur de bande globale d'environ 115 kHz.

Si nous augmentons encore le gain de la section d'entrée, l'amplitude à 20 kHz en serait notablement influencée. Si l'on veut un plus haut gain, c'est plutôt sur la deuxième section qu'il faut jouer, mais là,

les choses se compliquent en raison de la correction RIAA qu'elle opère. Il faudrait alors y changer toutes les résistances.

Il y a encore quelques réseaux de compensation sur le circuit imprimé de chaque canal du SUPRA 2.0 pour pouvoir adapter le courant de polarisation d'en-

trée des quatre amplis op. Les réseaux P1/R2/R3/C2/JP1 et P2/R31/R32/C27/JP2 ne sont pas nécessaires sur la version MC en raison de sa très basse impédance d'entrée. Mais il n'y a pas de mal à les y laisser s'ils sont déjà installés. Pour la version MC de SUPRA 2.0, nous avons utilisé la version A du LT1028. Elle

est un peu plus chère que l'ordinaire, qui n'est déjà pas bon marché, mais elle offre de plus faibles valeurs de décalage d'entrée et de courant de polarisation. Les plus grands décalages mesurés avec les amplis op A sur notre prototype avec un grand facteur d'amplification de 221 ne dépassaient pas 7 mV !

N'oubliez pourtant pas que la somme des quatre décalages est multipliée par 40 dans le second étage. Si la tension de décalage à la sortie de IC5A/IC10A dépasse quelques centaines de millivolts avec des LT1028 ordinaires, envisagez de les remplacer par la version A. C'est assez cher, il vous en faut quand même 8, mais pensez qu'une plus grande tension de décalage à la sortie de IC5A/IC10A va occasionner plus de

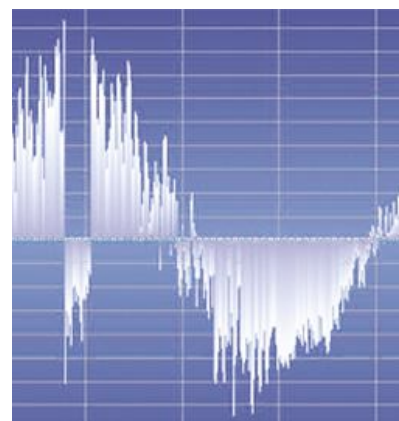
courant de bruit dans la rétroaction. Sur notre prototype, le décalage de sortie d'IC5A/IC10A est de 100 mV environ. Que votre collection de vinyles sur votre table de lecture équipée d'une cellule MC magnifiée par SUPRA 2.0 vous apporte tout le plaisir d'écoute possible ! ◀

(160263 – version française : Robert Grignard)

Liens

- [1] www.elektormagazine.fr/150616
www.elektormagazine.com/labs/supra-20-high-end-preamp-for-record-player-150616-i
- [2] www.elektormagazine.fr/labs/mc-modification-for-supra-20

À propos du bruit



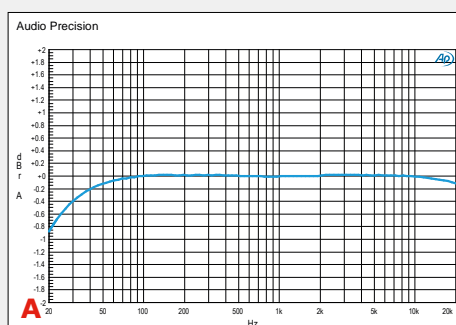
Le bruit thermique des quatre résistances de sommation de 6,81 kΩ à la sortie du LT1028 n'a presque pas d'influence sur le rapport signal/bruit des quatre amplis op. Le bruit dans une branche de rétroaction (2,2 kΩ en parallèle sur 10 Ω) vaut 0,41 nV/√Hz. L'entrée d'un LT1028 produit 1 nV/√Hz. Avec une cellule MC d'une résistance interne de 5 Ω, on arrive à 1,04 nV/√Hz. En conséquence, le bruit total d'entrée d'un ampli op est de 1,12 nV/√Hz (1,08 nV/√Hz avec l'entrée court-circuitée).

À la sortie de l'ampli op, le bruit sera 221 fois plus grand, 247 nV/√Hz. Ajoutons-y le bruit d'une résistance de 6,81 kΩ, le bruit en sortie monte à 247,2 nV/√Hz. Le bruit total derrière le nœud de sommation vaut la moitié de cette valeur, soit 123,6 nV/√Hz. Nous n'avons pas pris en considération dans ce calcul le bruit de courant.

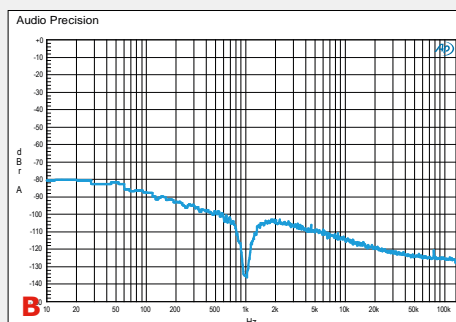
Le bruit de l'étage d'entrée MC est 3,7 fois supérieur à celui de l'entrée MD (en court-circuit). Le gain est lui, 4,62 fois plus grand. Si nous recalculons le bruit sur l'étage MD avec l'entrée en court-circuit (88 dB) pour 270 mV en sortie, le rapport signal/bruit est plus bas de 4,84 dB, donc 83,15 dB. La différence de bruit calculée entre MC et MD est de 11,4 dB avec les entrées en court-circuit.

Ces calculs nous apprennent que le rapport signal/bruit de la version MC de SUPRA 2.0 avoisine 72 dB, ce qui est très proche des valeurs mesurées (cf. Caractéristiques techniques et mesures) !

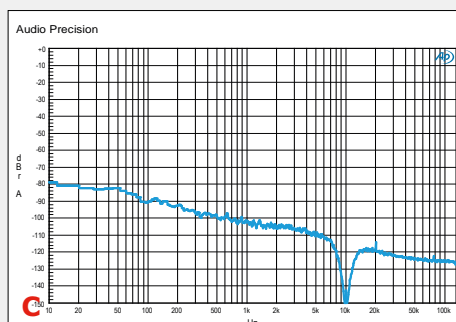
Ici aussi, quelques courbes de mesures en appui des résultats obtenus.



Le **graphe A** montre l'amplitude de sortie pour un signal sinusoïdal corrigé selon la norme RIAA appliqué à l'entrée. À 20 Hz, l'écart est d'un peu moins de -0,9 dB. À 20 kHz, il est de -0,1 dB. À 138 Hz et 3,3 kHz, on remarque un léger relèvement d'amplitude, mais +0,022 dB est une valeur négligeable. C'est exceptionnellement bon quand on pense que tous les composants responsables de la correction RIAA affichent une tolérance de 1 % !



Le **graphe B** représente le tracé d'une FFT pour un signal d'entrée de 1 kHz. Le spectre affiché va de 10 Hz à 130 kHz, avec réjection de la fréquence fondamentale de 1 kHz. Pour clarifier l'image, on a calculé la moyenne de seize mesures. Comme il n'y dans cette image aucun harmonique visible, nous pouvons en déduire que la distorsion harmonique totale est inférieure à 0,0007%.



Le **graphe C** représente le tracé d'une FFT similaire à celle de B, mais avec une fréquence d'entrée de 10 kHz. Ici aussi, la DHT est très basse, on ne voit que le deuxième harmonique à un niveau de -114 dB, ce qui correspond à 0,0002%.

commande de feux tricolores pour le modélisme ferroviaire

Piet Kralt (Pays-Bas)

Outre des voies ferrées, un réseau ferroviaire miniature comporte aussi des routes. Habiller une intersection routière de feux tricolores (surtout s'ils fonctionnent !) accroît le réalisme de la maquette. Avec cet article, découvrez comment commander les feux miniatures comme les grands.

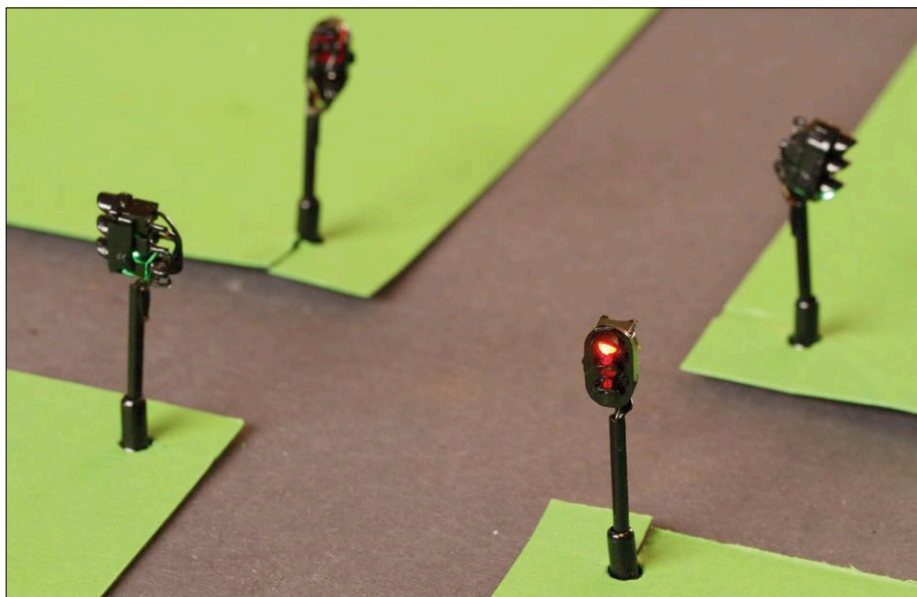


Figure 1. Dans le cas de réseaux à échelle N, chacune des LED a une ligne de cathode ; elles partagent une anode commune.

Construire ses propres feux tricolores est un véritable défi, mais heureusement on en trouve dans le commerce. La photo de la **figure 1** montre des « totems » pour des voies à l'échelle N. Avec ce modèle, les trois LED ont chacune une ligne de cathode, mais une anode commune. La ligne d'anode contient une résistance de limitation du courant de 1 k Ω . Les feux sont alimentés avec une tension comprise entre 12 et 15 V. Un petit calcul ($I = U/R$) montre qu'elles consomment de l'ordre de 10 à 13 mA.

Le matériel

Il existe une multitude de circuits électroniques pour commander ces feux tricolores, mais pour faire simple – au moins en ce qui concerne le matériel – j'ai opté pour une commande par logiciel avec un PIC, un 16F526. Comme une sortie peut débiter jusqu'à 25 mA, elle peut attaquer deux LED, mais cela à une

tension de 5 V seulement. Il faut donc modifier les feux tricolores : suppression de la résistance de 1 k Ω puis ajout d'un exemplaire de 270 ou 330 Ω dans chaque ligne de cathode (cf. schéma de la **figure 2**). Cela permet de commander les quatre feux de circulation d'un croisement classique ; les feux opposés de même couleur sont pilotés en parallèle, ils sont reliés à la même sortie du μC , via les résistances de limitation du courant.

En règle générale, un réseau ferroviaire miniature dispose d'une alimentation de 12 V, mais pas nécessairement d'une tension de 5 V, il faut donc un régulateur pour abaisser la tension d'alimentation comprise entre 8 et 16 V, appliquée sur K1. Quelques condensateurs et le 7085 (IC1) en font du 5 V.

Le connecteur SV1 permet de programmer *in situ* le PIC. On peut s'en pas-

ser si l'on utilise un programmeur. Les feux tricolores sont raccordés aux connecteurs SV3 et SV4. Chaque LED possède sa propre ligne. Le cavalier et le bouton-poussoir S1 peuvent servir à d'éventuelles nouvelles fonctions.

Le logiciel

Le logiciel sera d'autant plus compliqué que le matériel est simple. Toutefois nous ne trouvons pas que ce soit un inconvénient. Il est bien plus facile d'adapter un programme que de modifier un circuit électronique.

Pour la commande, j'ai commencé par écrire un petit programme simple en assembleur. Il est téléchargeable depuis [1]. La chronologie est fixe, mais on pourra bien évidemment la modifier dans le fichier .asm. Le connecteur SV2 permet de choisir entre le fonctionnement classique (rouge \rightarrow vert \rightarrow orange \rightarrow rouge) ou orange (jaune) clignotant.

Comme le circuit est simple, on a vite fait de le monter sur une platine d'expérimentation pour le tester.

Circuit imprimé du prototype

En [1], vous trouverez aussi le fichier 150710-11.zip qui contient les fichiers de CAO du circuit imprimé (Eagle). Nous avons délibérément créé un circuit compact. Il est possible de faire plus petit en utilisant des CMS, mais le gain en taille n'en vaut pas la chandelle, la réduction obtenue ne fait pas le poids face aux aspects délicats de l'utilisation de CMS. On voit en **figure 3** le prototype terminé. Dans la toute dernière version, les composants ont quelque peu bougé, surtout à cause du connecteur HE10 mâle. Une barrette ordinaire à 2x5 contacts est plus simple et meilleur marché, mais présente un risque évident : enfichage erroné du connecteur femelle. Nous le disons : chaque feu a sa propre ligne de connexion. Le **tableau 1** en donne le « brochage ». La connexion des « totems » 2 et 4 à SV4 est également conforme à ce tableau. Sur le prototype, les fils sont soudés directement au circuit imprimé, la plupart sur le dessous car cela simplifiait les choses. Au départ, nous avons envisagé d'utiliser de petits borniers à vis, encartables, mais les fils sont si fins que cela ne marche pas.

Conclusion

Ce projet a permis de réaliser un prototype fonctionnel qui respecte le cahier des charges. Les modifications et améliorations possibles sont nombreuses :

- une version qui utilise des circuits de commande plus costauds pour attaquer des LED plus puissantes, voire de petites ampoules ;
- réaliser la séquence rouge → rouge/orange → vert → orange → rouge que connaît un certain nombre de pays ;
- paramétrer différentes durées du

Tableau 1. Branchement des LED		
broche SV3	feu	connexion/couleur
1	1	common 1
3	1	vert
5	1	jaune
7	1	rouge
2	3	common 3
4	3	vert
6	3	jaune
8	3	rouge

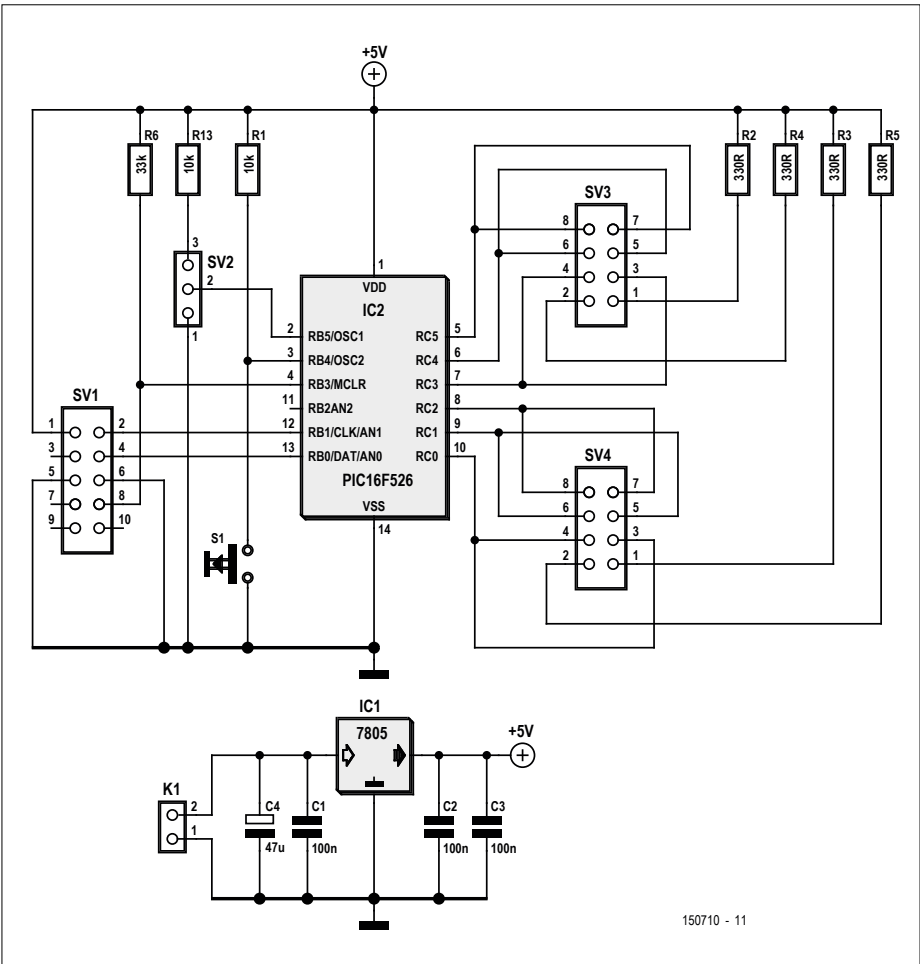


Figure 2. L'électronique du circuit se résume à bien peu de choses.

vert et de l'orange. C'est la raison de la présence du bouton-poussoir.

En quelques mots : donnez libre cours à votre imagination, adaptez le logiciel et le matériel.

(150710 – version française : Guy Raedersdorf)
Photos : Dirk Jan Kralt.

Liens

- [1] Téléchargements logiciel et CAO : www.elektormagazine.fr/150710
- [2] Page du projet : www.elektormagazine.fr/labs/stop-lichtautomaat-voor-de-modelbaan

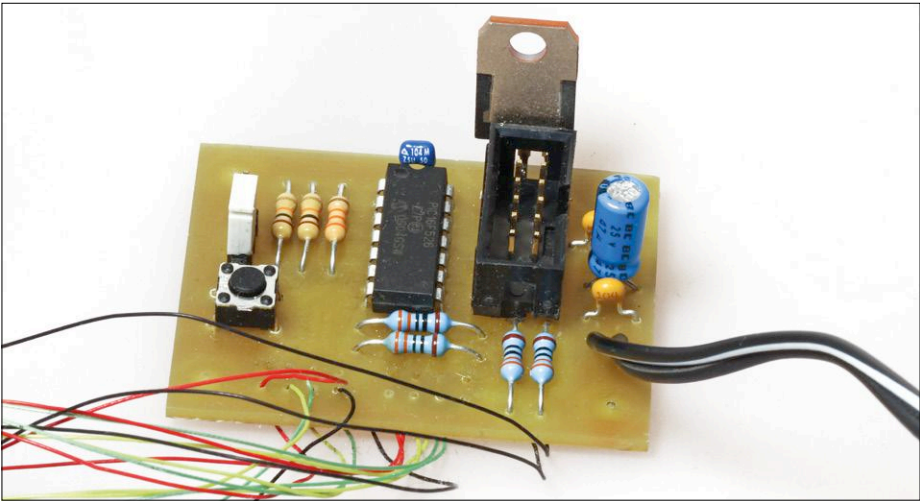


Figure 3. Sur le prototype, les fils de connexion sont soudés directement sur le circuit imprimé.

ukulé-LED

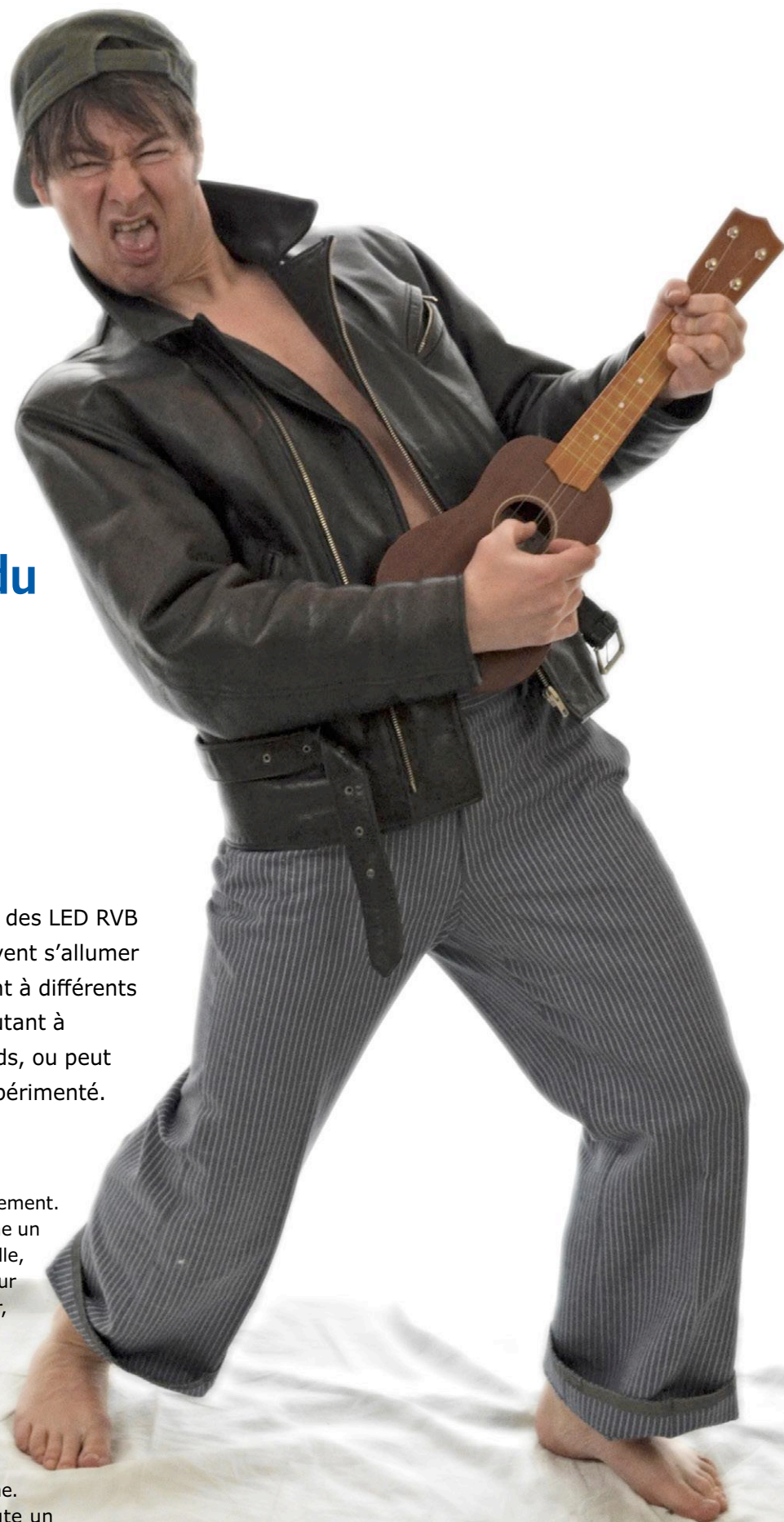
des LED adressables en série pour apprendre à jouer du ukulélé

Raghav Ayurkudi Subramaniam (États-Unis)

Le ukulé-LED est un ukulélé modifié avec des LED RVB adressables sur le manche. Ces LED peuvent s'allumer dans des configurations qui correspondent à différents accords, ce qui peut aider un joueur débutant à apprendre comment jouer certains accords, ou peut servir de guide pour un musicien plus expérimenté.

Le ukulé-LED possède deux modes de fonctionnement. Pour jouer (mode *play*), il faut fournir au système un fichier contenant le tempo, la signature temporelle, et une liste chronologique des accords à jouer. Sur le ukulélé, les LED indiqueront l'accord à jouer, au bon moment. Pour s'exercer (mode *practice*), l'utilisateur choisit un seul accord, qui reste allumé indéfiniment.

Nous utiliserons seize LED RVB, encastrées dans les quatre premières cases du manche sous chaque corde. Ces LED sont reliées à un microcontrôleur Atmel fixé sur le ukulélé lui-même. Le ukulélé peut être relié à un PC, qui exécute un



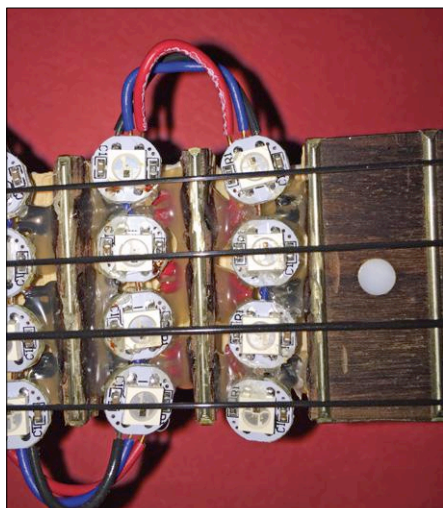


Figure 1. Gros plan sur le manche du ukulé-LED. Les seize LED RVB sont reliées en série et connectées à une seule broche d'E/S du microcontrôleur.

Caractéristiques

- Système d'apprentissage de la musique par LED RVB
- Modes *Exercice* et *Jeu*
- Format de fichier facile à comprendre

programme associé, doté d'une interface de commande en ligne.

Le matériel

Nous avons décidé d'utiliser un microcontrôleur Atmel ATmega1284P, car il est doté des deux temporisateurs à 16 bits nécessaires à ce projet. L'un est utilisé pour déclencher des routines, l'autre pour fixer le rythme. Nous avons également utilisé le circuit imprimé pour ATmega1284P de Bruce Land [2], doté entre autres d'un quartz, une LED et un régulateur de tension. Cela a facilité l'interfaçage du microcontrôleur. Il est également possible d'employer la carte Platino d'Elektor avec un ATmega1284P. Pour modifier le ukulé, nous avons utilisé un outil Dremel pour creuser dans le manche les rainures qui recevront les LEDs. Le point critique fut d'épargner les frettes et de garder les LEDs suffisamment basses pour que le ukulé reste jouable. Sinon, ce sont les LEDs et non les frettes qui auraient bloqué les cordes, décalant ainsi la note jouée. Nous avons fixé dans les rainures quatre jeux de quatre LEDs avec de la colle à chaud, soit une LED sous chaque corde. La **figure 1** montre le manche ainsi modifié.

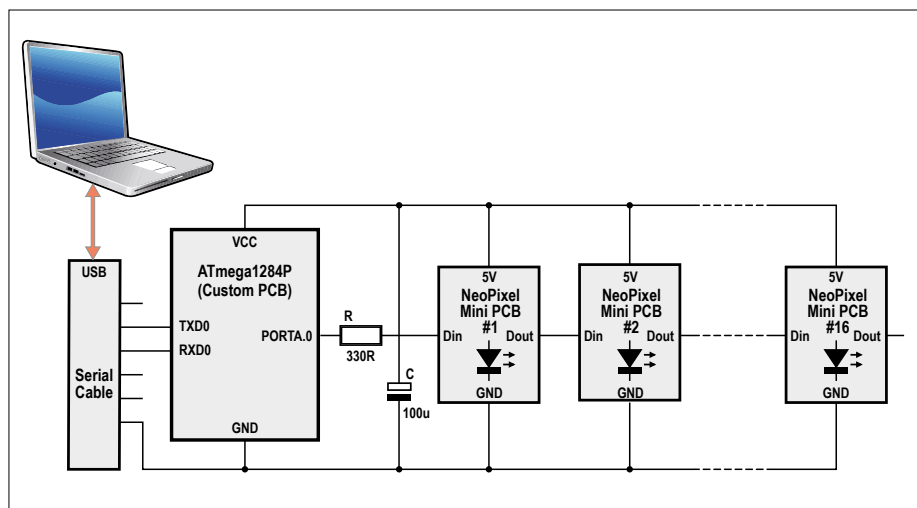


Figure 2. Synoptique de la partie matérielle qui comprend un microcontrôleur, des LED RVB et la liaison série vers le PC de l'utilisateur.

Nous avons utilisé des mini-circuits imprimés NeoPixel d'Adafruit, chacun doté d'une LED RVB WS2812B. Toutes les LEDs sont pilotées par une seule broche d'entrée/sortie du µC. Les WS2812B sont commandées en série, par un protocole 1-Wire au chronogramme très précis. La sortie de données de la première LED est reliée à l'entrée de données de la suivante, et ainsi de suite ; les broches d'alimentation des LEDs sont toutes reliées à la même ligne ; idem pour la masse. L'entrée de données de la première LED est reliée à la broche d'E/S du µC via une résistance de 330 Ω, afin de la protéger de toute pointe de tension. Précaution supplémentaire : un condensateur de 100 µF entre masse et alimentation évite aux grosses pointes de courant de détruire la première LED (conformément à la documentation d'Adafruit sur le

circuit imprimé NeoPixel [3]).

Le plus long dans la construction du ukulé-LED a été de câbler les LEDs RVB. Chacun des seize circuits NeoPixel devait être câblé individuellement. Nous avons soudé des fils très courts sur les pastilles des NeoPixel, de façon à avoir un espace de 1 à 2 mm entre les quatre LEDs d'une même rainure. Nous avons ensuite relié ces quatre groupes avec des fils plus longs. Finalement, nous avons raccordé les seize circuits NeoPixel à la carte principale avec trois longs fils.

Le travail suivant a été d'alimenter correctement les LEDs. Sachant que nous n'alimentons que quatre LEDs simultanément, en deçà de l'intensité maximale, nous pouvions utiliser la broche VCC du microcontrôleur plutôt que de recourir à une alimentation

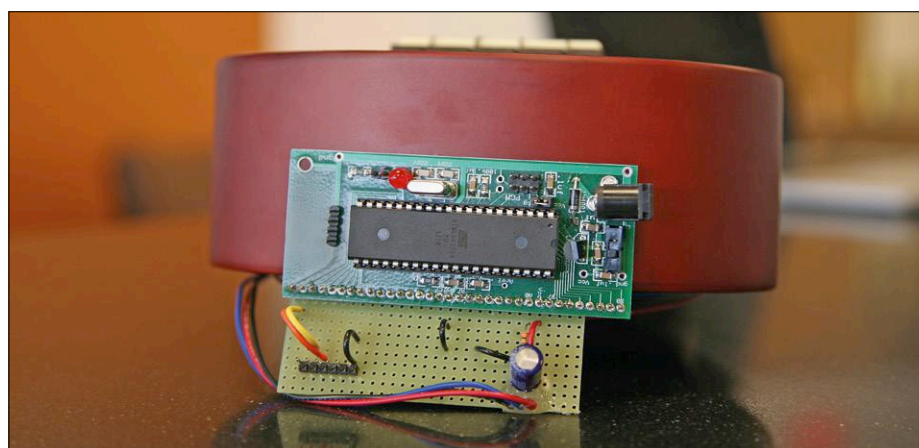


Figure 3. Carte fixée sur le ukulé. Elle supporte tous les composants annexes nécessaires au pilotage des LEDs et à la liaison série.

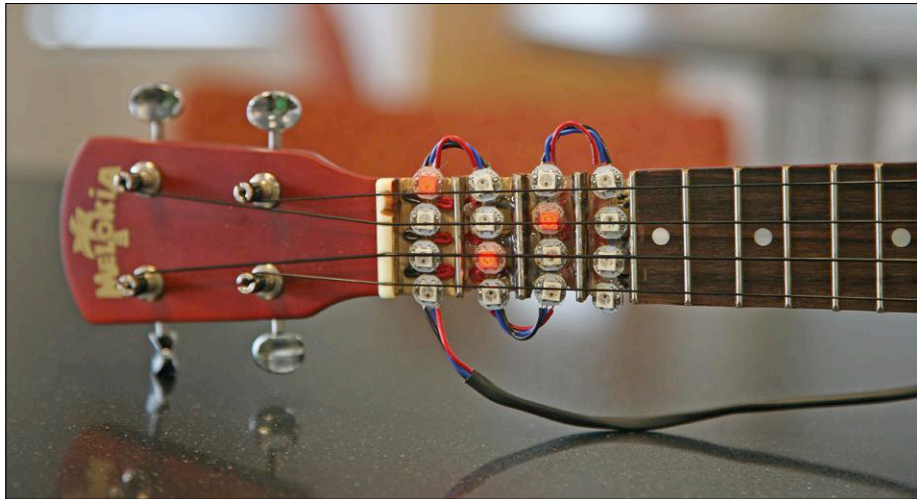


Figure 4. Un accord en Sol mineur, donc affiché en rouge.



Figure 5. Un accord en Sol septième, donc en bleu.

séparée. Enfin, nous n'avons eu besoin que d'une pile de 9 V pour alimenter le tout. La **figure 2** représente le circuit électronique du ukulé-LED.

Finalement, nous avons connecté le tout – microcontrôleur, circuit des LED, alimentation et câblage de la liaison USB – à une petite plaque à trous que nous avons collée à la base du ukulé, comme le montre la **figure 3**.

Interface en ligne de commande

Pour interagir avec le µC, notre projet nécessite un ordinateur qui exécute un interpréteur de ligne de commande fondé sur Python (pyCLI). Nous avons utilisé docopt [4], un analyseur d'arguments de ligne de commande Python. Il existe deux formats de commande pour entamer une session ukulé-LED :

```
ukule_led practice "chord" port
"portname"
```

```
ukule_led play "file" port
"portname"
```

Le premier indique au µC d'entrer en mode *practice* (exercice). Pour ce mode, le premier argument est *chord*, c'est-à-dire l'accord à jouer, par ex. A, C#m, ou Gb7 (soit La, Do dièse mineur, ou Sol bémol 7^e) ; le second argument est *portname*, à savoir le port série auquel est relié le µC via un câble convertisseur USB-série. Le second type de commande indique au µC d'entrer en mode *play* (jeu). Les arguments sont *file* qui spécifie le chemin du fichier à jouer sur l'ordinateur de l'utilisateur, et *portname* comme précédemment.

Nous avons créé notre propre format de fichier pour le ukulé-LED, les standards et protocoles existants ont été jugés trop sophistiqués. Voici un exemple de morceau :

```
240 4
|A B C D|E F G -|A - - B|
|A - Bbm A7|Bm X A#7 -|A - X D|
```

La première ligne du fichier contient deux nombres, le tempo du morceau en battements par minute, et la signature temporelle en battements par mesure. Un seul accord peut être joué par battement. Le reste du fichier contient les mesures du morceau, délimitées par le caractère « | ».

Le format accepte les accords majeurs, mineurs, et de septième. Nous avons fait en sorte que les accords majeurs soient indiqués en vert, les mineurs en rouge, et les septièmes en bleu. Des exemples sont visibles en figures 4 et 5. Une base d'accord peut être n'importe quelle note, un « # » signale un dièse et un « b » un bémol. Enfin, un tiret signifie le maintien de l'accord pour le battement suivant, et un « X » aucun accord durant ce battement.

Quand l'interpréteur pyCLI reçoit une commande, il utilise *docopt* pour en analyser les arguments. La commande est alors convertie en une chaîne envoyée sur la liaison série. Nous avons imaginé un codage des accords qui assigne à chaque nom d'accord un numéro à partir de 1. De plus, le 0 a été utilisé pour transcrire un « X » du fichier à jouer. « 1/17 » est un exemple de chaîne en mode *Exercice* et « 2/200/4/14/13/02/30/-1 » en est un du mode *Jeu*. Le premier nombre de la chaîne est un drapeau, 1 pour le mode *Exercice* et 2 pour *Jeu*. En mode *Exercice*, le nombre qui suit le 1 correspond à l'accord à jouer. En mode *Jeu*, le 2 est suivi du tempo, puis du nombre de battements par mesure, soit respectivement 200 et 4 ici. Les nombres suivants sont les accords du morceau dans l'ordre chronologique. Leur quantité n'est pas limitée ; la liste se termine par -1.

Le programme ouvre alors le port série demandé en ligne de commande, en utilisant le paquetage *pySerial* de Chris Liechti [5], et transmet la chaîne sur ce port.

Le micrologiciel

Le microcontrôleur exécute deux tâches principales. La première est de réceptionner les commandes envoyées par pyCLI sur le port série. Le µC contient un tableau dont les indices correspondent aux numéros des accords et les valeurs à la configuration des LED à allumer. Chaque centième de seconde, une seconde tâche recherche la valeur pointée dans ce tableau d'accords, et met à jour l'allumage des LED. En mode *Exercice*, ce pointeur est invariable. En mode *Jeu* par contre, un temporisateur déclenché selon une fréquence correspondant au tempo met à jour le pointeur du tableau d'accords selon le morceau à jouer (**figure 6**).

Nous avons utilisé le noyau `TinyRealTime` (TRT), développé par Dan Henriksson et Anton Cervin, pour cadencer ces deux tâches. Nous avons également fait appel au pilote de LED WS2812 d'Alan Burlison [7]. Ce pilote est écrit en assembleur et tient compte des contraintes du chronogramme de commande des LED RVB.

Le micrologiciel contient une fonction `main` qui met en route le pilote des LED, en spécifiant le port d'entrée/sortie auquel les LED sont connectées. Ensuite, il configure TRT conformément à sa documentation. TRT utilise le temporisateur 1 (`timer1`), l'un des deux temporisateurs à 16 bits de l'ATmega1284P. La fonction `main` règle ensuite l'autre temporisateur à 16 bits (`timer3`), qui gère les changements d'accords.

La première tâche en temps réel, réalisée par la fonction `serialComm`, attend que les données de l'utilisateur arrivent par le port série et les range à mesure qu'elles arrivent. Elle octroie un tampon de commande `cmd` de 1 000 octets, suffisant pour la majorité des chansons, et appelle la fonction TRT `gets(cmd)`, qui attend le sémaphore signalant que TRT reçoit des données du port série. Lorsque la fonction sort de l'attente, une commande (un ensemble de nombres extraits de la chaîne transmise par pyCLI sur la liaison série) sera stockée dans `cmd`.

Si la commande est en mode *Exercice*, `serialComm` charge dans la variable `chord_idx` le numéro de l'accord à jouer par les LED. `chord_idx` est le pointeur du tableau d'accords. Cette valeur reste fixe jusqu'à la réception d'une nouvelle valeur. Si la commande est en mode *Jeu*,

Remerciements

Je remercie pour leur aide le Dr Bruce Land et mon maître de stage, Eileen Liu. Le ukulé-LED a pu naître grâce aux logiciels libres de Dan Henriksson, Anton Cervin, Jeorg Wunsch, Alan Burlison, Chris Liechti, et aux développeurs de *docopt*. Enfin, ce projet n'aurait jamais vu le jour sans mon partenaire, Jeff Tian.

`serialComm` lit le second nombre de la commande, qui représente le tempo en battements par minute, et convertit cette valeur pour régler la période du temporisateur 3 d'après la formule suivante :

$$\text{période} = (62500 \times 60) / \text{tempo} - 1$$

Après réglage de `timer3`, la fonction extrait le troisième argument de la commande, le nombre de battements par mesure. Enfin, `serialComm` appelle un assistant qui parcourt le tableau `cmd`, convertit les nombres au format « caractère » en « entier », et les insère dans un tableau global appelé `song`. On utilise l'interruption du temporisateur 3 pour rythmer le morceau. Chaque fois que l'interruption est déclenchée, on exécute `chord_idx = song[song_idx]`, où `song_idx` est l'index actuel dans `song`, puis on incrémente `song_idx`.

La seconde tâche en temps réel examine simplement la valeur de `chord_idx` fixée par la routine d'interruption, en déduit la zone de `lookup_table` correspondante, et passe les données au pilote de LED.

Conclusions

Mon but avec ukulé-LED était de bâtir un système pratique et extensible pour apprendre à jouer du ukulé. Mon instrument est de même poids qu'un ukulé normal et le ressenti

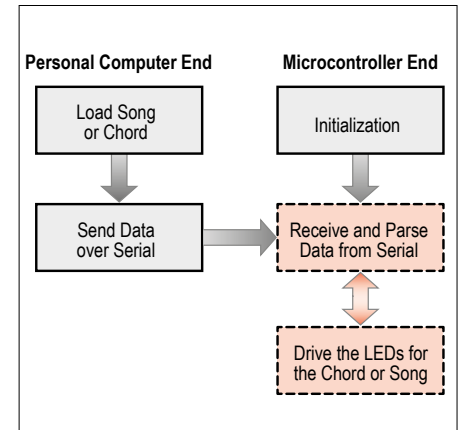


Figure 6. Ce diagramme montre les différentes tâches logicielles réalisées par le ukulé-LED, dont deux tâches en temps réel pour le microcontrôleur et une interface de commande en ligne qui tourne sur le PC de l'utilisateur.

est semblable. L'interface logicielle est simple et facile à utiliser, grâce à *docopt*. N'hésitez pas à modifier les fichiers `chord_map.py` et `chord_lookup_table.h`, qui déterminent quels accords sont pris en charge et avec quelles couleurs ils sont représentés. Une extension possible serait, en complétant ces deux fichiers, d'accepter tous les types d'accords. ◀

(160439 – version française : Guy Matt)

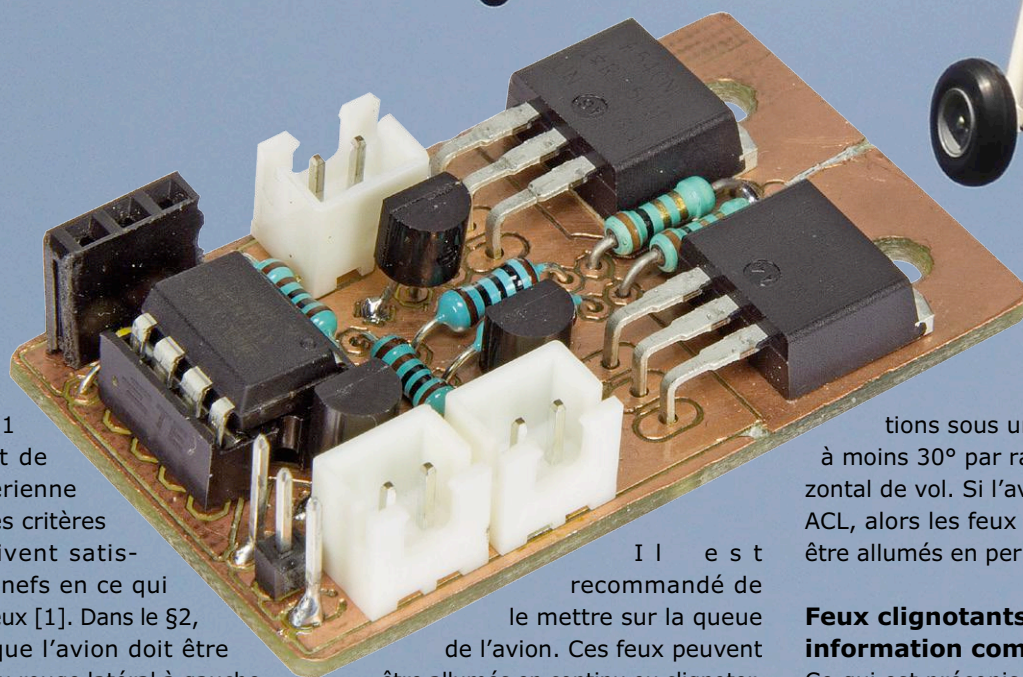
Liens

- [1] www.elektormagazine.fr/160439
- [2] <http://people.ece.cornell.edu/land/PROJECTS/ProtoBoard476/>
- [3] <https://learn.adafruit.com/adafruit-neopixel-uberguide>
- [4] <http://docopt.org/>
- [5] <http://pyserial.sourceforge.net/>
- [6] www.control.lth.se/~anton/tinyrealtime/
- [7] <https://sourceforge.net/projects/abavr/>

feux de position multifonctions pour tout modèle réduit volant

En aéromodélisme, le modèle réduit doit être fidèle dans les moindres détails à son original grandeur nature. Ses dispositifs d'éclairage n'y font pas exception. Mais ceci ne doit pas nous empêcher de doter ces feux d'une petite fonction supplémentaire : ils nous transmettent également des informations sur l'état de charge des accumulateurs de la maquette.

Udo Hunzelmann, DK8VT@DARC.de



Dans l'annexe 1 du règlement de circulation aérienne sont définis les critères auxquels doivent satisfaire les aéronefs en ce qui concerne les feux [1]. Dans le §2, on peut lire que l'avion doit être équipé d'un feu rouge latéral à gauche et d'un vert à droite ; les feux latéraux présentent un angle d'ouverture de 110°. Il y a en plus un feu blanc à l'arrière visible avec un angle d'ouverture de 140°.

Il est recommandé de le mettre sur la queue de l'avion. Ces feux peuvent être allumés en continu ou clignoter. Outre ces feux de position, le §3 recommande d'avoir des feux clignotants anti-collision (*Anti Collision Light*, ACL) qui peuvent être vus de toutes les direc-

tions sous un angle de plus 30° à moins 30° par rapport au plan horizontal de vol. Si l'avion est équipé d'un ACL, alors les feux de position doivent être allumés en permanence.

Feux clignotants avec information complémentaire

Ce qui est préconisé pour un avion (ou un hélicoptère) devrait aussi l'être pour un modèle réduit. Mais de fait, en tant que pilotes de modèle réduit, nous ne sommes pas tenus à respecter les pres-

criptions légales des « grands », et nous pouvons ajouter aux fonctions initiales des feux externes une petite fonction supplémentaire bien utile : une indication visuelle de l'état de charge des accumulateurs !

En effet, pendant le vol d'un modèle réduit subsiste toujours le problème de n'avoir aucune visibilité sur l'état de charge ou de décharge de ses batteries. Afin de pallier ce problème, j'ai conçu un petit circuit clignotant qui influence le clignotement des feux en fonction de l'état de charge des accumulateurs. On peut aussi y raccorder un avertisseur sonore qui permet de trouver, grâce à un signal sonore, l'endroit où se trouve la maquette.

Le circuit en **figure 1** est connecté par K1

L'ATtiny45 dispose d'une fonction interne extraordinaire : il peut mesurer la tension sur sa broche VCC ! Je l'ai évidemment utilisée et j'ai conçu une sorte de comparateur à fenêtre pour VCC/2, sans ajouter des composants (uniquement par logiciel) :

- $VCC/2 > 3,7\text{ V}$ Batterie chargée : 3× clignotement
- $3,4\text{ V} \leq VCC/2 \leq 3,7\text{ V}$ Batterie semi-chargée : 2× clignotement
- $VCC/2 < 3,4\text{ V}$ Batterie déchargée : 1× clignotement

Ce circuit comporte néanmoins un défaut, car on ne mesure que la première cellule qui représente toutes les autres. Le prochain circuit devra mesurer la tension totale et par là toutes les cellules.

à la sortie servo du contrôleur du modèle réduit ; il contient un petit microcontrôleur ATtiny45 avec seulement huit pattes qui pilote trois sorties à commutation au travers d'un MOSFET à canal N. Le vibreur connecté à K2 tire un courant de faible intensité (à cause de la petite tension d'alimentation, de l'ordre de 5 mA, selon le modèle) et on peut se contenter d'un transistor de commutation pour signaux faibles BS170. Il en va autrement pour les LED de forte puissance des clignotants connectés à K3 et K4. Les MOSFET de puissance IRF540 en boîtier TO220 gèrent sans peine les forts courants des LED. Les sous-ensembles constitués des shunts R3 et R6 et des transistors bipolaires BC547C limitent les courants et veillent à ce que les étages à MOSFET soient immédiatement désactivés en cas de court-circuit (des LED). Le circuit est alimenté à l'aide du bloc d'accumulateurs (au moins deux cellules LiPo) et du câble d'alimentation avec équilibreur [2]. Il faut raccorder cette alimentation aux trois broches d'embase mâle (K5) que l'on voit sur la **figure 2**. Alors que les LED de forte puissance sont alimentées avec une tension de 7,4 V, le contrôleur et le vibreur le sont par la moitié (VCC/2), soit 3,7 V.

INFOS SUR LE PROJET

aéromodélisme

feux de position

ACL

➔ débutant

connaisseur
expert

Environ une heure

Soudure,
programmeur ATtiny45,
EDI Arduino

Environ 20 €

L'assemblage du circuit sur le circuit imprimé ne prend qu'une heure. Ce

circuit imprimé est disponible dans l'e-choppe et son tracé est téléchargeable,

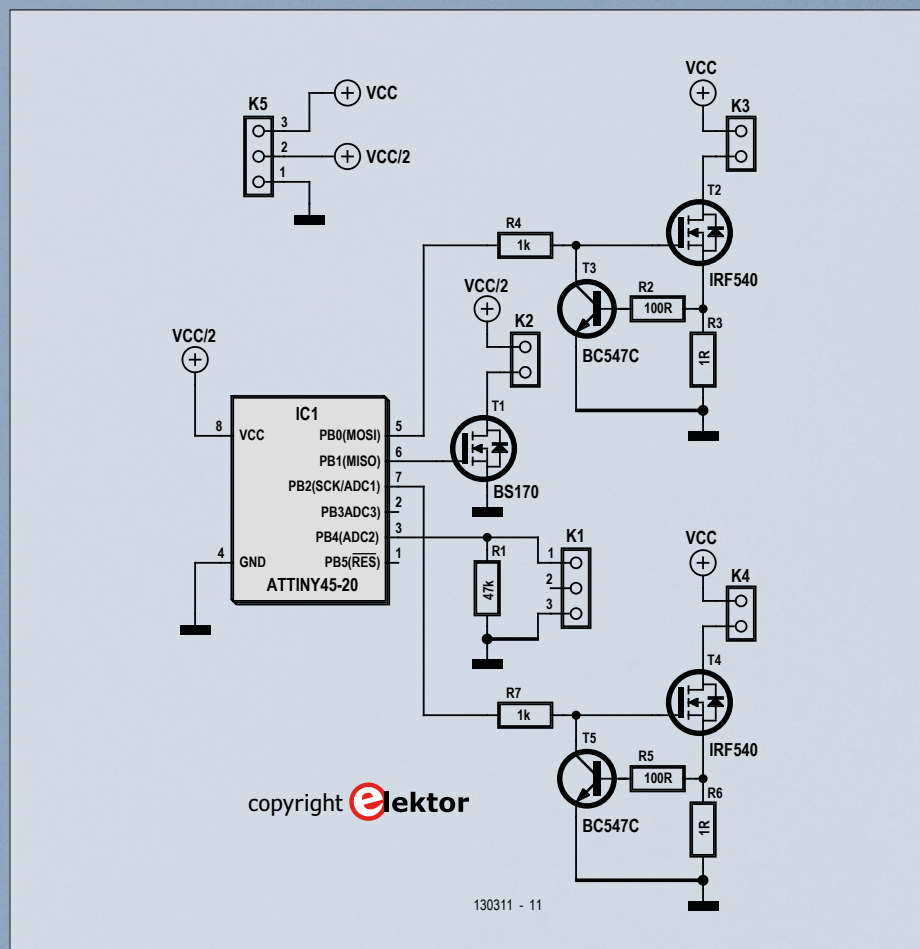


Figure 1. Circuit du clignotant pour modèles réduits.

voir [3]. Il n'y a pas de difficulté majeure puisque tous les composants sont traversants, pas de CMS.

Trois modes de fonctionnement par logiciel

Le logiciel du circuit clignotant permet de configurer trois modes ; un interrupteur à 3 positions sur la télécommande permet de changer de mode aussi bien en plein vol qu'après l'atterrissage.

- Interrupteur vers l'arrière : clignotants et vibreur éteints.
- Interrupteur au milieu : clignotants allumés et vibreur éteint.
- Interrupteur vers l'avant : clignotants allumés et vibreur actif.

La position médiane est prévue pour le vol, la troisième pour après. Si le modèle réduit s'écrase (ce qu'on ne souhaite à personne, mais cela peut arriver aux meilleurs...), on peut le localiser grâce au signal sonore sur les terrains accidentés. Le signal sonore est configuré à 4 kHz dans le logiciel, mais vous pouvez facilement modifier la tonalité selon votre envie.

Le logiciel [3] a initialement été écrit en BASCOM et porté ultérieurement en C dans l'EDI Arduino. Le choix du mode se fait immédiatement au début de la boucle principale, et dépend de la longueur des impulsions de servo (< 1,25 ms, 1,25 à 1,75 ms, > 1,75 ms). En mode 2, quand le clignotement des LED doit renseigner sur l'état de la charge restante des accumulateurs, l'ATtiny45 mesure la

tension de fonctionnement et la quantifie. La simplicité de fonctionnement se retrouve dans le **listage 1** (extrait).

État des accumulateurs en un coup d'œil

À l'automne dernier, j'ai construit et intégré deux exemplaires du clignotant dans deux de mes planeurs électriques. J'ai pu constater que les éclairs des LED, par grand soleil, bonne visibilité et à une altitude de vol du modèle réduit estimée à 100 m étaient bien reconnaissables, même si le modèle était éloigné de près de 500 m. Je ne pense pas surestimer cette appréciation parce que d'autres pilotes me l'ont confirmé.

Quand le signal sonore était activé en vol, on pouvait l'entendre jusqu'à 50/60 m lors d'un vol à 30 m. Ceci dépend évi-

Listage 1. Mesure du VCC par l'ATtiny45 (extrait du programme).

```
void measureBattery() {
    batteryVoltage = readVcc();
    if (batteryVoltage >= 3800) {           //more than 3.7V: battery full
        ledBlink(3);
    }
    if (batteryVoltage >= 3500 && batteryVoltage < 3800) {
        //between 3.3V and 3.7V, apparently there is a small offset
        ledBlink(2);
    }
    if (batteryVoltage < 3500) {           //less than 3.4V: battery empty
        ledBlink(1);
    }
}

long readVcc() {
    // Read 1.1V reference against AVcc
    // first set the reference to Vcc and the measurement to the internal 1.1V reference
    ...
    #elif defined (__AVR_ATtiny25__) || defined(__AVR_ATtiny45__) || defined(__AVR_ATtiny85__)
        ADMUX = _BV(MUX3) | _BV(MUX2);
    ...
    delay(2);                             // Wait for Vref to settle

    ADCSRA |= _BV(ADSC);                  // Start conversion
    while (bit_is_set(ADCSRA, ADSC));      // measuring

    uint8_t low  = ADCL;                   // must read ADCL first - it then locks ADCH
    uint8_t high = ADCH;                   // unlocks both

    long result = (high << 8) | low;

    result = 1125300L / result;            // Calculate Vcc (in mV); 1125300 = 1.1*1023*1000
    return result;                         // Vcc in millivolts
}
```




LISTE DES COMPOSANTS

Résistances :

R1 = 47 kΩ
R2, R5 = 100 Ω
R3, R6 = 1 Ω
R4, R7 = 1 kΩ

Semi-conducteurs :

T1 = BS170
T2, T4 = IRF540
T3, T5 = BC547C
IC1 = ATtiny45-20PU (DIP 8), 130111-41

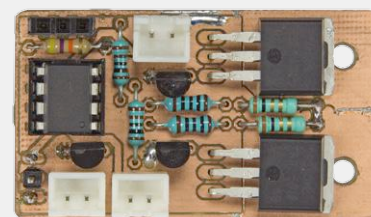
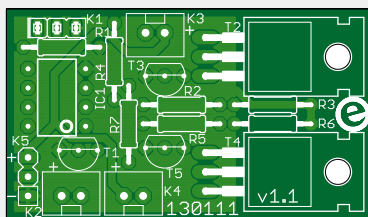


Figure 2. Implémentation des composants facile : aucun CMS.

Divers :

K1 = embase femelle à 3 contacts,
pas de 2,54 mm
K2, K3, K4 = embase renforcée XH (JST chez

RS Components 820-1554) et boîtier adapté
(RS Components 820-1611) avec cosses à
sertir (RS Components 820-1529)
K5 = connecteur à 3 broches, pas de 2,54 mm
Circuit imprimé réf. 130111-1

demment aussi du type de vibreur et de l'endroit où il est monté sur le modèle. Il existe des vibreurs qui, sous une tension d'alimentation basse de 3 V murmure avec 72 dB, et d'autres qui sous de telles conditions arrivent à 105 dB ! On peut donc, selon le choix, optimiser la portée acoustique.

Évolution

Il y a quelques jours j'ai construit un clignotant plus grand pour un modèle réduit d'un collègue. Ce circuit permet de piloter des LED supplémentaires, par exemple des rouges et vertes en continu en complément du clignotement ACL. Deux LED blanches peuvent également être commandées et un vibreur est disponible. Les fonctions de l'interrupteur de la télécommande ont été très peu modifiées. En position 1, le clignotement est désactivé, en position médiane toutes les LED travaillent comme décrit (et le vibreur est éteint) et en position 3 les LED clignotantes sont désactivées et le vibreur activé.

De bonnes perspectives

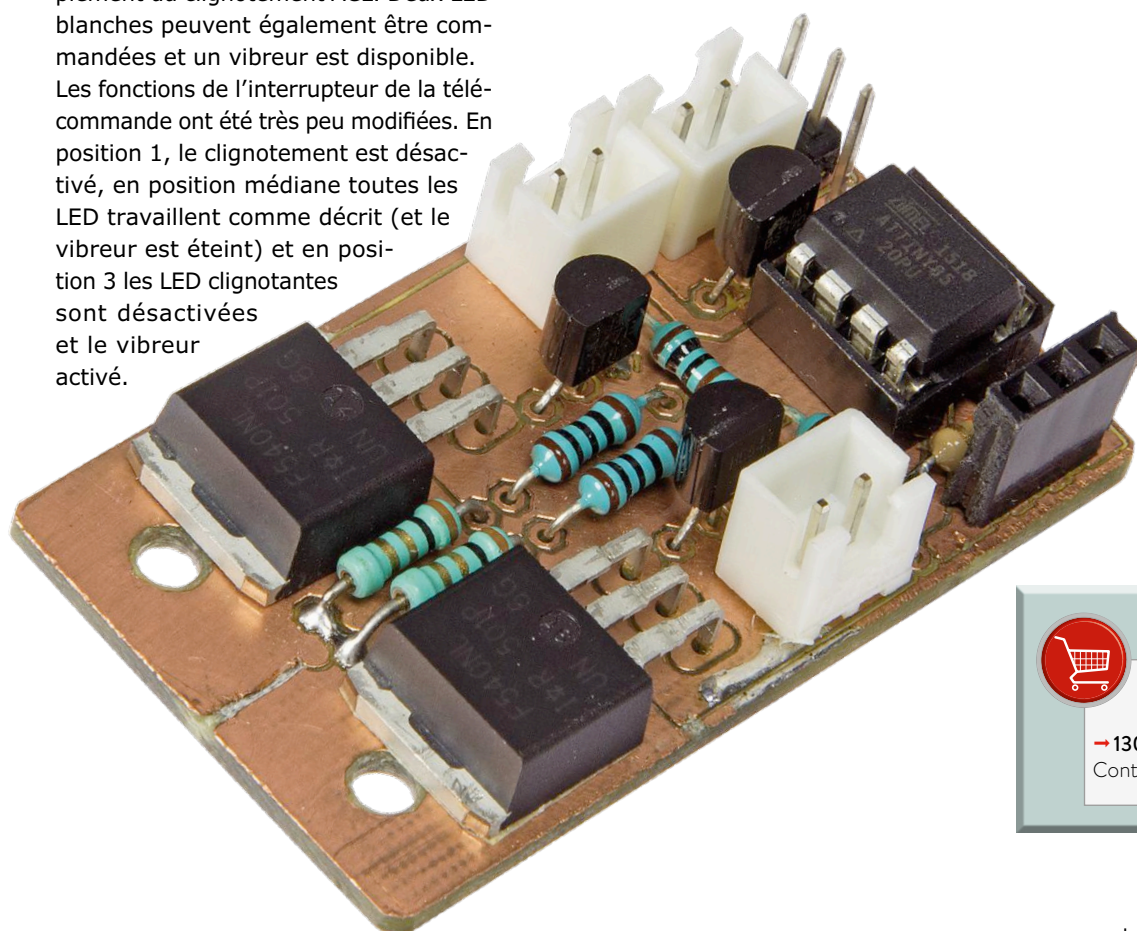
En ce moment, un collègue aéromodéliste intègre ce clignotant étendu dans son planeur électrique d'une envergure de 4,5 m. Le premier vol de test du planeur avec le clignotant pourra bientôt avoir lieu. Je suis également affairé à une extension du clignotant à six fonctions. Pour ce faire, le clignotant nécessite deux connexions au servo du récepteur. On peut alors, pour commander le clignotant et les autres fonctions, utiliser sur l'émetteur l'interrupteur à 3 positions et un curseur.

Je serais ravi de savoir si vous avez construit le clignotant et l'avez intégré à vos différents modèles réduits. Tous les commentaires et toutes les propositions d'amélioration sont les bienvenus. ◀

(130111 – version française : Patrick Bechler)

Liens

- [1] www.buzer.de/s1.htm?g=luftvoanl_1&a=1-6
- [2] http://cache.freescale.com/files/32bit/doc/app_note/AN4428.pdf
- [3] www.elektormagazine.fr/130111



DANS L'E-CHOPPE

→ 130111-1

Circuit imprimé nu

→ 130111-41

Contrôleur programmé



QUESTIONS D'ÉTHIQUE

appareils intelligents qui invitez-vous chez vous ?

Tessel Renzenbrink (Pays-Bas)

« La porte refusa de s'ouvrir et déclara :
- *Cinq cents, s'il vous plaît.*
À nouveau, il chercha dans ses poches. Plus de pièces ; plus rien.
- *Je vous paierai demain*, dit-il à la porte. (Il essaya une fois de plus d'actionner le verrou, mais celui-ci demeura fermé).
Les pièces que je vous donne constituent un pourboire, je ne suis pas obligé de vous payer.
- *Je ne suis pas de cet avis*, dit la porte. *Regardez dans le contrat que vous avez signé en emménageant dans ce conapt.*
Il trouva le contrat dans le tiroir de son bureau : depuis que le document avait été établi, il avait eu besoin maintes et maintes fois de s'y référer. La porte avait raison ; le paiement pour son ouverture et sa fermeture faisait partie des charges et n'avait rien de facultatif.
- *Vous avez pu voir que je ne me trompais pas*, dit la porte avec une certaine suffisance.
Joe Chip sortit un couteau en acier inoxydable du tiroir à côté de l'évier ; il s'en munit et entreprit systématiquement de démonter le verrou de sa porte insatiable.
- *Je vous poursuivrai en justice*, dit la porte tandis que tombait la première vis.
- *Je n'ai jamais été poursuivi en justice par une porte, mais je ne pense pas que j'en mourrai.* »

Philip K. Dick, *Ubik*, 1966-1969 ; traduction d'Alain Dorémieux, 1970

Qui invitons-nous chez nous lorsque nous achetons un appareil intelligent ? À qui ou quoi ouvrons-nous la porte lorsque nous remplaçons nos appareils dépassés et « stupides » par d'autres, connectés à l'Internet des Objets (IdO – *Internet of Things*, IoT) ? Joe Chip, le personnage principal d'*Ubik*, roman de science-fiction écrit par Philip K. Dick en 1966, est constamment confronté à des objets qui lui réclament leur dû avant qu'il puisse les utiliser. L'ouvrage prêtait à sourire lorsqu'il fut publié en 1969 ; aujourd'hui, en 2017, on le considère comme visionnaire.

Un fabricant rancunier

La querelle, en avril dernier, entre le fabricant de *Garadget* – un appareil connecté qui permet d'ouvrir et fermer une porte de garage via le *nuage* (cloud) – et un utilisateur est un bon exemple actuel de l'univers d'*Ubik*. L'utilisateur, un certain R. Martin, ne parvenait pas à faire fonctionner l'appareil, et a demandé l'aide de la communauté *Garadget* ; mécontent de ne pas recevoir de réponse assez rapidement, il a posté un

commentaire très négatif sur *Amazon.com*, qualifiant l'appareil de camelote (*piece of junk*).

Le fabricant n'a pas apprécié du tout, et il a répliqué en mettant l'appareil hors service, acte facilité par le fait que la commande passe par un serveur qu'il contrôle. Il a en outre répondu à R. Martin sur le forum de *Garadget* : « l'appareil dont le numéro d'identification est 2f0036... n'a plus accès au serveur. Votre seule option est désormais de retourner l'appareil à *Amazon* pour vous faire rembourser ».

Un tissu de relations

L'essor des appareils connectés a complètement changé la donne des relations fabricant-consommateur. Auparavant, un achat était une transaction ponctuelle : vous achetez un frigo, on vous le livre à la maison, vous le branchez, et si tout va bien vous n'avez plus besoin du vendeur ou du fabricant pendant au moins dix ans.

Avec les produits connectés, ce n'est plus le cas. La relation est de longue durée, car vous acquérez en fait un paquet de services. Un de ces services est par ex. l'entretien du serveur distant qui accompagne votre frigo ; vous devez être assuré que le fabricant maintiendra ce serveur dans le *nuage* pendant toute la durée de vie estimée de l'appareil. D'autres parties sont aussi souvent impliquées, ce qui crée un tissu de relations (économiques et informatiques) ; le fabricant est entre autres dépendant de l'entreprise qui héberge les serveurs. Bien entendu, tous ces acteurs ne peuvent pas tomber en faillite...

Le microcontrôleur qui gère le frigo va recevoir de temps à autre des mises à jour du logiciel, sans lesquelles l'appareil pourrait devenir dangereux ou même cesser de fonctionner. Cela arrive fréquemment avec des téléphones, qui du jour au lendemain sont obsolètes : le matériel est en parfait état de marche, mais le système d'exploitation n'est plus soutenu par le fabricant.

Un autre problème est celui de l'usage fait des données collectées par les appareils. Récemment, *Vizio*, un fabricant américain d'appareils électroniques, s'est vu infliger une amende salée, parce qu'il avait revendu les données du comportement de quelque 11 millions de téléspectateurs à des tiers. Ce n'est pas la vente elle-même qui était ainsi condamnée, mais bien le fait de ne pas en avoir informé les téléspectateurs au préalable.

Des tracteurs immobiles et une récolte foutue

Les exemples de consommateurs pris en otage par leurs appareils connectés ou les fabricants de ceux-ci sont légion. La plus grosse perte de l'utilisateur de *Garadget* a été son après-midi passée à monter l'appareil, mais les conséquences peuvent être bien plus sérieuses.

Les tracteurs du fabricant américain *John Deere* ne sont plus de simples machines agricoles, mais bien des ordinateurs sur

roues, avec système d'exploitation et moult programmes. Lors de l'acquisition, les fermiers doivent signer des conditions d'utilisation ; celles-ci stipulent que l'utilisateur ne peut plus « bricoler » son engin, mais doit faire appel à un technicien qualifié. Ces derniers sont non seulement plus chers, mais aussi parfois stationnés à une distance considérable ; les temps d'intervention – et d'immobilisation du tracteur – s'allongent... En période de récolte, une machine en panne, c'est beaucoup d'argent perdu ! Le temps où les agriculteurs réparaient eux-mêmes leurs machines, parfois avec l'aide d'un mécanicien auto local, est bien révolu ; du moins, s'ils veulent rester dans la légalité. Le journaliste Jason Koebler s'est entretenu pour le compte du site *Motherboard* avec des agriculteurs, qui consultent les forums de *hacking* pour y trouver des logiciels piratés [1]. Ils peuvent ainsi télécharger un programme de diagnostic pour le matériel et le logiciel ; ces agriculteurs estiment par ailleurs que ce programme devrait être fourni gratuitement par *John Deere*. Avec ces programmes illicites, les utilisateurs retrouvent un peu d'autonomie et de voix au chapitre sur leur propre matériel. Cela a cependant un prix, dit John Koebler : la crainte constante qu'ils soient poursuivis en justice pour violation des conditions d'utilisation, et pire, que *John Deere* bloque purement et simplement le tracteur à distance.

Tout comme la porte suffisante de Joe Chip, les appareils connectés sont certes intelligents, mais la véritable question est bien de savoir à qui sert cette intelligence ! ◀

(160393 – version française :
Jean-Louis Mehren)

Lien

[1] is.gd/mpeZUY



Marionnettiste (photo : domaine public).



bienvenue dans votre e-choppe

la rédaction recommande

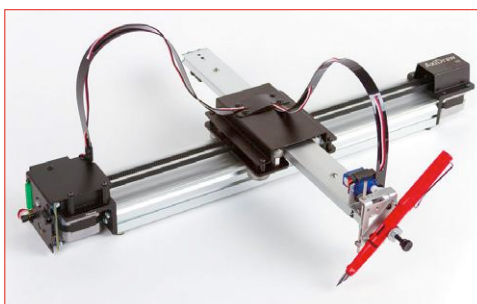


AxiDraw

AxiDraw est une table traçante (*plotter*) selon la terminologie classique, ou un robot dessinateur, capable de reproduire n'importe quel dessin vectoriel ou vectorisé sur une surface plane. Pas seulement du papier, mais aussi du carton, un tableau blanc, des grands panneaux (aussi à la verticale !), de la tôle, etc. Par rapport à une table traçante conventionnelle, sa grande originalité est donc de pouvoir dessiner ou écrire sur des surfaces plus grandes que la machine elle-même. AxiDraw n'est pas un kit, il est fourni (presque) prêt à l'emploi. Ce n'est pas un jouet, plutôt un outil

d'expérimentation et de (re)production. Sa construction en coulisses XY orthogonales est robuste et rigide (alu extrudé) ; toutes les pièces rigides sont métalliques, même le porte-stylo et les capots. L'impression de grande précision est confirmée par le poids (2,2 kg). AxiDraw est un traceur très polyvalent conçu pour répondre aux besoins les plus divers en matière de dessin et d'écriture !

Denis Meyer (labo d'Elektor)



www.elektor.fr/axidraw

Vos favoris :

1. LabNation SmartScope
www.elektor.fr/smartscope



2. Microscope USB Andonstar
www.elektor.fr/andonstar-usb-microscope

3. Raspberry Pi 3
www.elektor.fr/raspberry-pi-3

4. Elektor Uno R4
www.elektor.fr/elektor-uno-r4

5. Carte de commande LEGO pour Raspberry Pi
www.elektor.fr/lego-rpi-board

6. DVD Elektor 2016
www.elektor.fr/dvd-2016

Maîtrisez les microcontrôleurs à l'aide d'Arduino



3^e édition augmentée avec AVR Playground & Elektor Uno R4

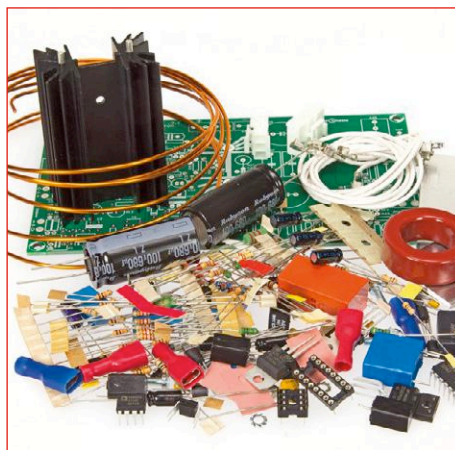
L'ambition de cet ouvrage est de vous faire entrer dans le monde Arduino puis de vous en faire ressortir victorieux pour vous emmener plus loin dans l'apprentissage de la programmation des microcontrôleurs. Il met la théorie en pratique sur une carte Arduino avec l'environnement de programmation Arduino.



Prix (membres) : 32,85 €

www.elektor.fr/maitrisez-les-microcontroleurs-3

Déwattreur – kit de l'ampli audio de classe D

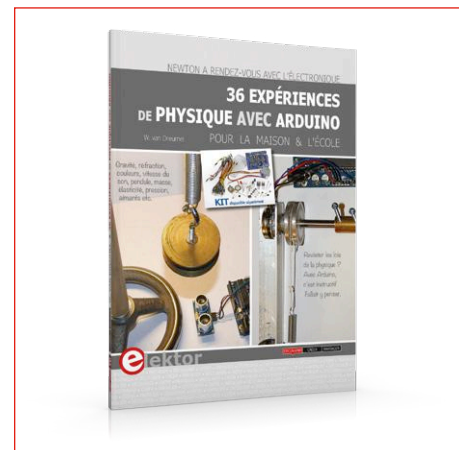


Le déwattreur, ce nouvel ampli de puissance en classe D, est conçu à partir d'une puce audionumérique spéciale, l'IRS2092. Cet ampli est puissant (200 W sur 8 Ω), mais ne dissipe pratiquement pas de chaleur. Finis les gros radiateurs ! Attention : il faut assembler (solder) le kit proposé ici, toutefois il n'y a aucun CMS, que des composants traversants. Le boîtier et l'alimentation ne sont pas inclus, il faut les commander séparément.

Prix (membres) : 98,96 €

www.elektor.fr/d-watter

36 Expériences de Physique avec Arduino

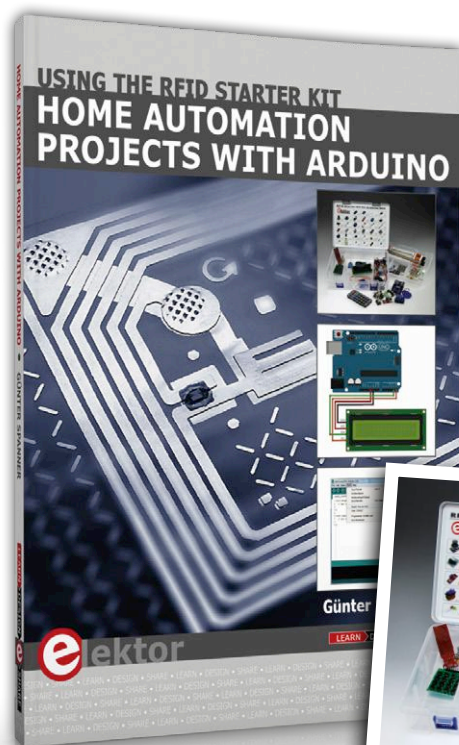


Ce livre n'est pas un manuel de physique : pas d'équations différentielles ni de courbes abstraites, mais des phénomènes physiques de la vie quotidienne. C'est une approche nouvelle et créative des leçons de physique grâce aux techniques modernes de mesure et de traitement des données. L'électronique utilisée (Arduino) est simple. Ajoutez-y le logiciel gratuit CoolTerm pour enregistrer les mesures et les retravailler ensuite sous Excel.



Prix (membres) : 22,41 €

www.elektor.fr/arduino-36-experiences



Home Automation Projects with Arduino

Offre groupée : livre (en anglais) + kit d'initiation RFID pour Arduino



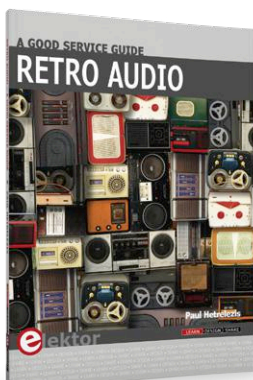
Pour réaliser des projets vraiment novateurs, il faut souvent farfouiller dans les entrailles des composants, ce qui peut poser de gros problèmes aux débutants. C'est là que ce livre (en anglais) peut les aider. Il explique comment réaliser nombre de projets pratiques à l'aide d'un ensemble de composants et d'une carte Arduino. Ces composants sont rassemblés dans un kit baptisé « kit d'initiation RFID pour Arduino ». Toutefois ce kit ne se limite pas aux applications RFID, ses 30 composants et modules couvrent tous les domaines de l'électronique (capteurs de température/son/IR, clavier, commutateurs, joystick, module RVB, pilote de moteur, horloge en temps réel...).



Prix (membres) : 69,66 €

www.elektor.fr/offre-rfid

Retro Audio (livre en anglais)



Aujourd'hui il y a un regain d'intérêt pour les disques vinyle et les tourne-disques, ce qui a donné naissance à un marché parallèle de la réparation. Le but de ce livre (en anglais) est d'apporter au lecteur, électronicien amateur ou confirmé, des connaissances ainsi que des solutions en matière d'audio d'antan. Cet ouvrage contient des tableaux avec des données utiles, des trucs pour l'entretien ainsi que des informations pas si communes.



Prix (membres) : 26,96 €

www.elektor.fr/retro-audio

Microscope USB modèle V160 d'Andonstar



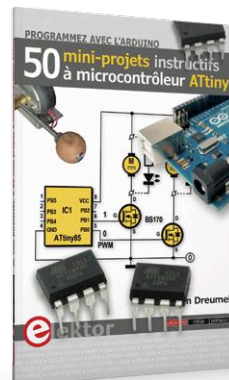
Le microscope USB modèle V160 d'Andonstar fournit une image stable et nette grâce à son support massif. En outre les réglages y sont précis et faciles. Le champ de vision dépend de la molette de réglage de la platine. La mise au point est confiée à une bague tournante placée sur le boîtier de la caméra. La qualité de son capteur CMOS de deux mégapixels (sans interpolation) à 30 images/s vous surprendra, surtout pour un si petit prix !



Prix (membres) : 80,96 €

www.elektor.fr/microscope-usb-andonstar

50 mini-projets instructifs à µC ATTiny



Pour mettre au point des montages à microcontrôleur, la carte Arduino est parfaite. Ensuite, on peut obtenir le même résultat avec un autre µC, plus petit et moins cher, comme l'ATTiny. Ce livre montre comment faire à l'aide de 50 applications originales, voire déjantées. Dans ces pages, ni théorie ni mathématiques, que de la pratique ! ATTiny programmé avec une carte Arduino ; croquis disponibles sur cette plateforme transposables pour l'ATTiny.



Prix (membres) : 22,41 €

www.elektor.fr/50-mini-projets-attiny-arduino

Hexazener casse-tête pour elektorniciens

Pour cet été, notre traditionnelle grille hexadoku est remplacée par une grille « hexazener » (le contour de cette grille dessine une zener).

Cette grille est proposée par un fidèle lecteur, Claude Ghyselen. Les règles appliquées pour remplir la grille ainsi que les caractères autorisés restent les mêmes que pour la grille hexadoku normale.

Rappel : une grille hexadoku est composée de chiffres du système hexadécimal, de 0 à F. Il faut la remplir de telle façon que tous les chiffres hexadécimaux de 0 à F (0 à 9 et A à F) n'apparaissent qu'une seule et unique fois dans chaque rangée, colonne et bloc (les blocs sont délimités par un filet gras).

L'hexazener se distingue de l'hexadoku par les points suivants :

- La grille comporte 16 blocs, mais 18 lignes et 20 colonnes.
- Quatre zones restent vides (en blanc).

- Les blocs ne sont pas de forme carrée ou rectangulaire, ils ressemblent plutôt à des pièces de puzzle (on parle d'*hexadoku jigsaw*).
- Le nombre de cases de chaque bloc dépend de sa position et de sa forme, mais n'excède pas 16. Cette grille peut donc toujours être qualifiée d'hexadoku.
- Le nombre total de cases actives est de 186, contre 256 pour un hexadoku.

Comme certains blocs contiennent moins de 16 cases, il n'est pas possible d'y inscrire tous les caractères hexadécimaux. C'est une petite difficulté supplémentaire dont il faut tenir compte lors de la résolution. Certains chiffres, déjà placés dans la grille, définissent la situation de départ. Pour participer, inutile de nous envoyer toute la grille, il suffit de nous envoyer la série de cinq chiffres sur fond grisé.

Participez et gagnez !

Nous tirons au sort **cinq** des réponses internationales correctes reçues dans les délais ; leurs auteurs recevront chacun un bon d'achat Elektor d'une valeur de **50 €**.

À vos crayons !

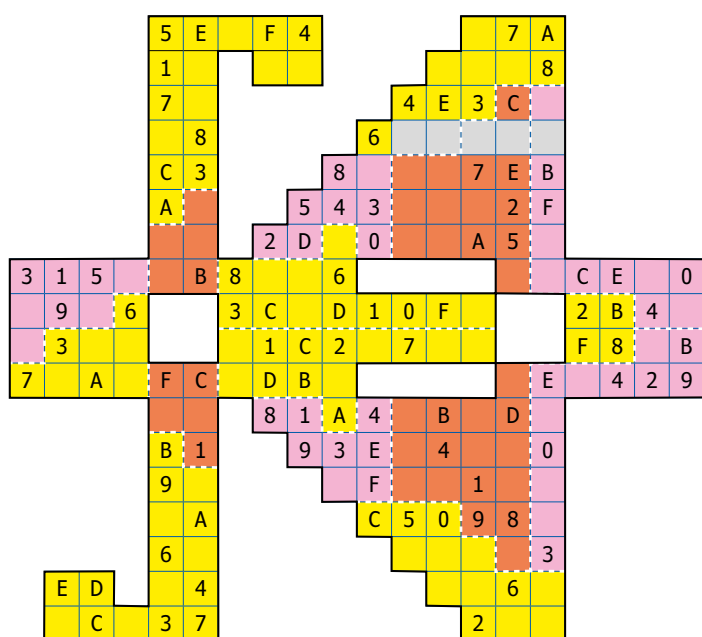
Envoyez votre réponse (les chiffres sur fond grisé) avec vos coordonnées par courriel, avant le **20 juillet 2017** à l'adresse **hexadoku@elektor.fr**

Les gagnants

La solution de la grille du numéro de mai/juin est **1E709**

Les cinq bons Elektor d'une valeur de 50 € vont à : Nikos **Chararas** (Autriche), Ariaan **Hoed** (Belgique), Vincent **Pierdominici** (États-Unis), Guy **Savage** (Royaume-Uni) et Jean **Thevenoux** (France).

Bravo à tous les participants et félicitations aux gagnants !



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

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.

enova

PARIS

LE SALON
DES **TECHNOLOGIES**
POUR LES **INNOVATIONS**
DE DEMAIN

ÉLECTRONIQUE / EMBARQUÉ / IOT / MESURE / VISION / OPTIQUE / BIG DATA

19 AU 21 SEPT
2017

Paris expo Porte de Versailles - Hall 4

NOUVEAU

+ d'exposants

+ de conférences

+ de services

+ village Prototypage / Impression 3D

DONNEZ VIE À VOS PROJETS

AÉRONAUTIQUE | MILITAIRE | AGROALIMENTAIRE | AGRICOLE | AUTOMOBILE | TRANSPORT
SMART CITIES | SMART BUILDING | MÉDICAL | INDUSTRIE 4.0 | RECHERCHE ACADÉMIQUE

Made by



BADGE GRATUIT : www.enova-event.com

MÊME LIEU, MÊMES DATES

18^e CONGRÈS INTERNATIONAL DE MÉTROLOGIE

CIM2017



ALIMENTATION ÉLECTRIQUE AUTONOME IDÉAL POUR LES DÉPLACEMENTS

(Intenso)[®]

Intenso Slim-Powerbank, 10.000 mAh

**Notre astuce de prix, convient idéalement
pour les smartphones & les tablettes !**

- Capacité des cellules : 10.000 mAh, ions lithium
- Prise Output: USB A et câble micro USB
- Output : 5 V DC, max. 2,1 A
- Éléments de protection : protection contre les surtensions, protection contre la décharge, protection contre les surcharges, fusible de protection contre les court-circuits
- Dimensions : 130 x 70 x 16 mm

N° de commande :
au lieu de 18,45 INTENSO 7332530

**Économisez
41% 10,88**



AIDE POUR CHOISIR



Smartphone
pour powerbanks avec 1–1,5 A
et 2.200–6.000 mAh



Pc Tablettes
pour powerbanks avec plus
de 1,5 A et 6.000–20.000 mAh

Bloc d'alimentation universel stabilisé, 12–24 V DC

**Alimente avec fiabilité en énergie toute
fraîche les netbooks et les notebooks dans la voiture !**

gobay[®]

- Tension d'entrée : 12–24 V DC
- Sortie : 15–20 V DC/max. 4.730 mA
- Avec fiche mâle de raccordement 10 DC pour différents appareils
- Protection contre les court-circuits, les surcharges, les sur-intensité et la surchauffe
- Port de charge USB intégré, 5 V DC



N° de commande :
NTS 90 USB

29,08

Logilink Slim-Powerbank, 20.000 mAh

**Le géant de l'énergie parmi
les powerbanks**

- Sortie : 5 V DC, max. 2,1 A
- Affichage de niveau de charge LED (4 segments)
- Fonctions de protection : fusible de protection contre la surcharge, fusible de protection contre la décharge profonde, protection contre les court-circuits
- 2 prises de charge USB
- Dimensions (longueur x largeur x hauteur) : 160 x 82 x 23 mm

**ENCORE UN
PEU PLUS ?!**

N° de commande :
LOGILINK PA0086B

25,17



**NOTRE HOTLINE EXISTE DÉSORMAIS
ÉGALEMENT DANS VOTRE LANGUE**

Nous attendons votre appel :
+33 97 518 03 04



Prix du jour! Prix à la date du: 29. 5. 2017

Prix en € hors T.V.A., frais de port en sus - reichelt elektronik, Elektronikring 1, 26452 Sande (Germany)

Les langues de notre boutique:

MODES DE PAIEMENT INTERNATIONAUX:

SOFORT
BANKING

VISA

MasterCard

amazon pay

PayPal

ACHETER EN LIGNE TOUT SIMPLEMENT !



www.reichelt.fr

ASSISTANCE TÉLÉPHONIQUE: +33 97 518 03 04