



elektor

DÉCOUVRIR

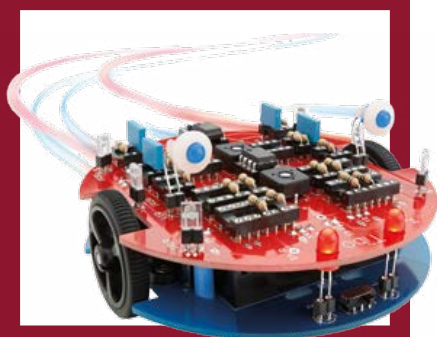
CRÉER

PARTAGER



The invisible touch !

µC ARM pour
néophytes – 8^e partie



robot analogique
de l'idée au kit



drôles de drones
quelques modèles
surprenants

générateur de fonctions par DDS

sinus, carrés et dents de scie
jusqu'à 10 MHz

Dans cette édition :
4 projets de labo
1 projet de lecteur
2 cours
Q&R, trucs et astuces et
bien d'avantage...



convertisseur USB-série

compact (puce FT232H) • VFD

ou afficheur fluorescent • carte d'E/S pour

Android (3) : enregistreur de données • convertisseur *buck* tête • zootrope
numérique • FX3S et USB 3.0 (solution anti-goulot d'étranglement) • T-Board bien
ARMée avec la puissance du Cortex-M0+ à 32 bits • préamplificateur Preco •
ampli de classe A, B, C, D, E, F, G, H : quesako ? • trucs & astuces pour Eagle (2)



KCS TraceME

2G 3G 4G LBS

LoRa™ BLE M2M

Iridium Sensor



Bluetooth®

iBeacon™

SMS

Glomass GPRS

RF GPS

Internet of Things



L'Internet des Objets avec LoRa™

KCS enrichit sa gamme de produits TraceME avec un puissant module conçu pour la mobilité planétaire et l'Internet des Objets. Dans sa dernière version, ce module de suivi et de traçage par GPS/GPRS allie géolocalisation RF et technologie LoRa™. Cette combinaison ajoute de l'intelligence aux objets communicants grâce aux longues portées et faibles consommations que permet la technologie LoRa™ dans les applications de communication entre machines. La prise en charge des protocoles GPRS/SMS et celle, optionnelle, des standards 3G, Wi-Fi, Bluetooth LE, ANT/ANT+ et iBeacon™ facilite l'intégration du module dans les réseaux sans fil et les applications mobiles. Nous le déclinons bientôt en versions haut de gamme et gamme intermédiaire.

Module ANTIVOL à RF

Avec sa gamme de produits TraceME, KCS offre une solution de géolocalisation intelligente pour les applications antivol et à usage en intérieur. Elle utilise la RF et un algorithme intelligent pour mesurer le temps de propagation des signaux transmis (selon un protocole propriétaire). Notre module présente des caractéristiques uniques: petitesse (46x21x6,5 mm), légèreté (7 g pour une carte équipée) et longévité (durée de vie supérieure à 10 ans en mode veille). Notre algorithme « Écouter avant de communiquer » rend pratiquement impossible la localisation du module et sécurise ainsi le véhicule ou le bien à protéger. L'intégration dans les réseaux sans fil et les applications mobiles est facilitée par la prise en charge des protocoles GPRS/SMS et celle, optionnelle, des standards 3G, Wi-Fi, Bluetooth LE, ANT/ANT+ et iBeacon™.

www.Trace.ME

Toutes les marques mentionnées ici sont la propriété de leurs propriétaires respectifs.

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 :

Fabio Romagnoli +32 485 65 40 90

fabio.romagnoli@eimworld.com

DROITS D'AUTEUR :

© 2015 Elektor International Media B.V.

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

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

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



« Où sont les boutons de l'autoradio ? »

Cette question déroutante (c'est le cas de le dire), je me la pose tous les jours depuis que, sur l'écran tactile du tableau de bord de ma nouvelle voiture, je dois tâtonner pour changer de station de radio : impossible d'appuyer à coup sûr sur le bon bouton (virtuel) du pavé tactile sans forcément quitter la route des yeux. Une manœuvre risquée et même dangereuse !

À certains modèles d'interface tactile, les fabricants d'écrans ont ajouté un retour de force, pour reproduire au toucher la résistance et le mouvement des objets manipulés virtuellement. C'est le cas des manettes de jeux par exemple, qui vibrent lorsqu'on les touche. Un jour, les plus perfectionnés de ces systèmes dits *haptiques* simuleront des formes et des textures différentes. Ce jour-là, s'il peut se fier au seul titillement sur ses doigts, sans regarder, le conducteur pourra enfin rester concentré sur la route. Pensez-vous que les sensations seront aussi explicites qu'avec de vrais boutons, poussoirs ou rotatifs ?

J'en doute. En plus, dans les menus, les fonctions d'un tableau de bord multimédia sont nombreuses. Les concepteurs d'interfaces tactiles pour l'automobile devront beaucoup réfléchir aussi aux séquences de manipulation et aux schémas de navigation. Pour l'instant, pas question de les atteindre sans regarder. J'ai tâtonné longtemps pour simplement annuler un itinéraire sur le GPS. Qu'est-ce que c'est compliqué ! Il reste beaucoup à améliorer *pour que l'automobile soit toujours un plaisir**.

Lorsque vous développerez un projet basé sur la carte d'E/S pour Android d'Elektor (page 32 de ce numéro), mettez-vous dans la peau de l'utilisateur de votre appli tactile !

Bonne lecture.

Mariline Thiebaut-Brodier

* Slogan d'un constructeur français

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 :	Thierry Destinobles, Eric Dusart, Robert Grignard, Jean-Louis Mehren, Denis Meyer, Hervé Moreau, Kévin Petit, Guy Raedersdorf
Service de la clientèle :	Cindy Tijssen
Graphistes :	Giel Dols, Mart Schroijen, Patrick Wielders
Elektor en ligne :	Daniëlle Mertens

- 5 bientôt dans Elektor
- 6 Elektor : guide de connexion(s)
- 26 le labo d'Elektor
- 66 l'e-choppe d'Elektor
- 81 des nouvelles du monde d'Elektor
Est-ce e-éthique ?
- 82 Hexadoku
casse-tête pour elektorniciens
- 83 Agenda nov./déc. 2015

DÉCOUVRIR CRÉER PARTAGER

- 8 bienvenue dans la section DÉCOUVRIR
- 9 trucs et astuces
Jamais assez de lignes de port ?
- 10 µC ARM pour néophytes
pour passer de 8 bits à 32 bits - 8^e partie :
« The invisible touch ! »
- 19 VFD ou afficheur fluorescent
drôle de composant n°20
- 20 trucs & astuces pour Eagle (2)
créer une nomenclature avec ULP
- 22 sac à malices
aux Olympiades de Sciences de l'Ingénieur
- 24 FX3S et USB 3.0
la solution anti-goulot d'étranglement

DÉCOUVRIR CRÉER PARTAGER

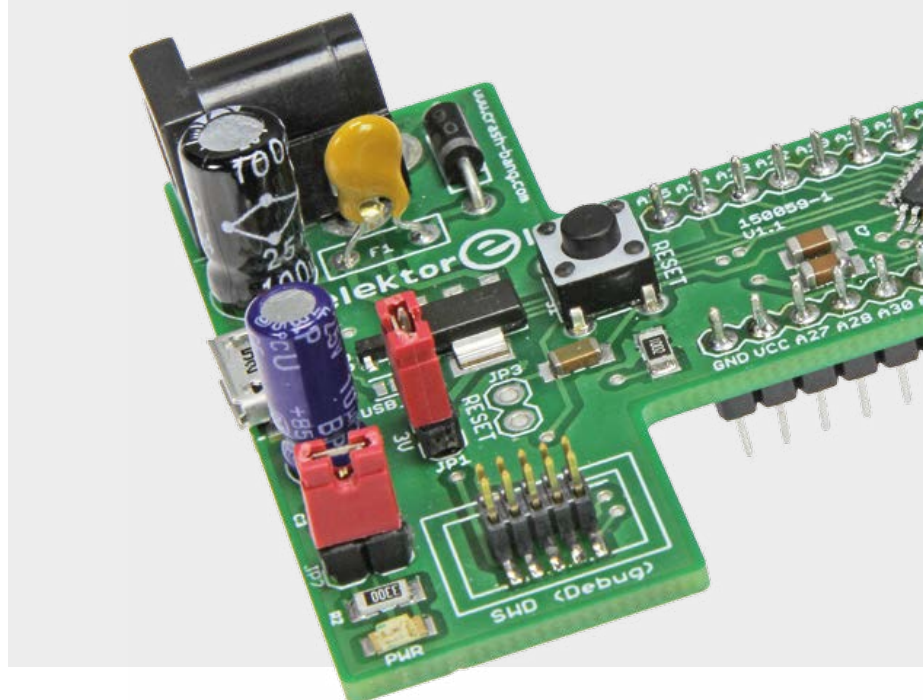
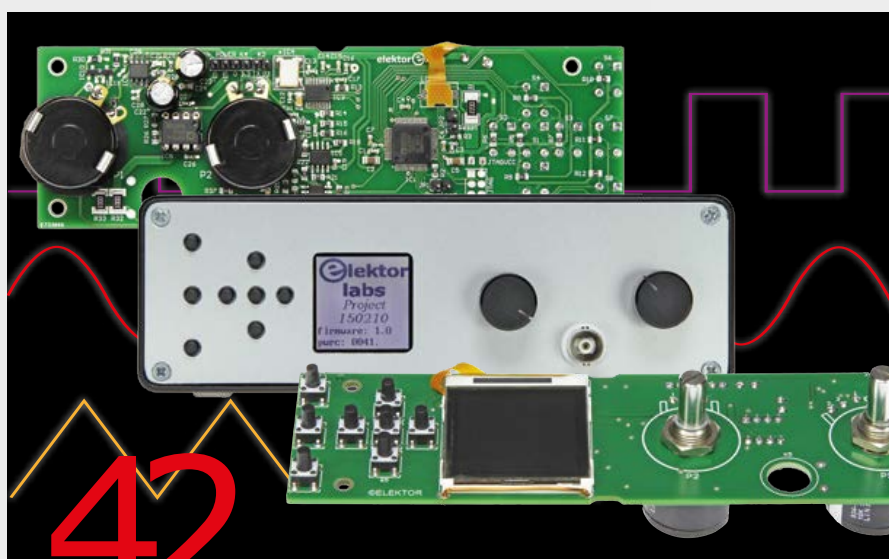
- 28 bienvenue dans la section CRÉER
- 29 convertisseur USB-série compact
avec la puce FT232H
- 32 carte d'E/S pour Android (3)
l'enregistreur de données
- 39 zootrope numérique
ou comment faire du neuf avec du vieux
- 42 générateur de fonctions par DDS
sinus, carrés et dents de scie jusqu'à 10 MHz
- 50 robot analogique
de l'idée au kit

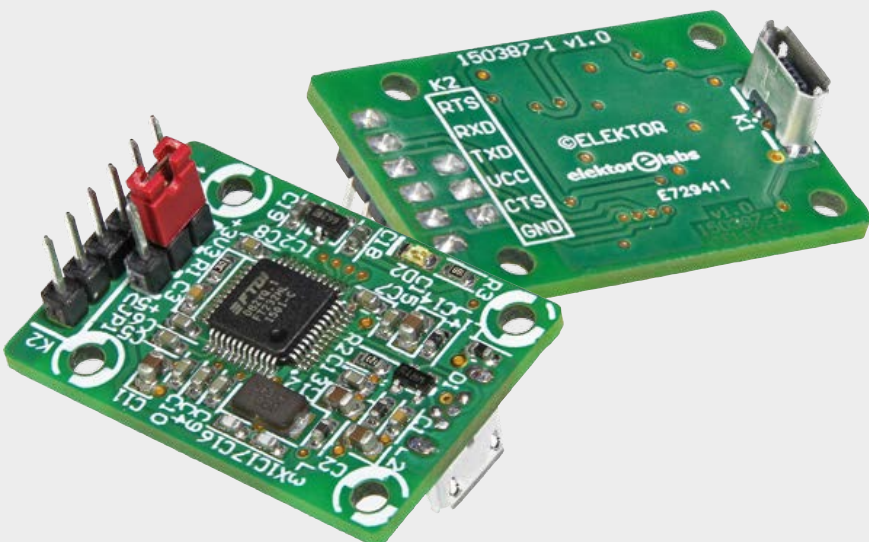
29

convertisseur USB-série compact

avec la puce FT232H

Le port série, bien que démodé, est toujours très utilisé comme interface entre PC et électronique. Voici un module compact sur la base du circuit intégré FT232H de FTDI, qui pourra être connecté « à chaud » (*plug-and-play*) au bus USB 2.0, après installation des pilotes (*drivers*) ad hoc, pour des liaisons à haute vitesse (*high speed*).

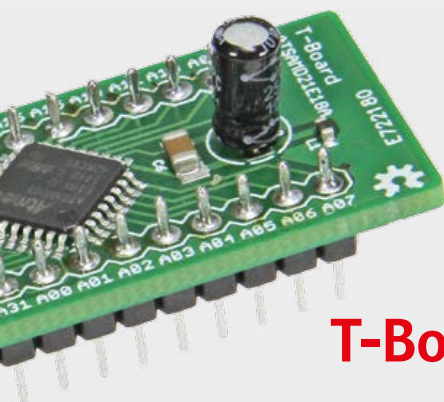




générateur de fonctions par DDS

sinus, carrés et dents de scie jusqu'à 10 MHz

Les puces DDS (*Direct Digital Synthesis*) simplifient grandement la construction de la partie analogique des générateurs de fonctions. Ajoutez à votre puce DDS une circuiterie de sortie, un microcontrôleur, une interface utilisateur, et voilà. Simple comme un jeu de briques. Du moins en théorie. En pratique, ce n'est pas si facile. À moins de choisir la bonne puce !



54

T-Board bien ARMée

avec la puissance du Cortex-M0+ à 32 bits

Créés il y a un peu plus d'an, les moTules T-Board sont en passe de devenir une institution chez Elektor. Après les premières versions à AVR à 8 bits, la T-Board sans fil et la T-Board « ampli audio », voici la dernière (et la plus puissante) de la famille : la T-Board à ARM à 32 bits.

54 T-Board bien ARMée

avec la puissance du Cortex-M0+ à 32 bits

62 hors-circuits avec R. Lacoste

ampli de classe A, B, C, D, E, F, G, H : quesako ?

DÉCOUVRIR

CRÉER

PARTAGER

71 du changement dans la section PARTAGER

72 convertisseur buck tête

au comportement erratique

74 préamplificateur Preco

39 ans en mode Hi-déFi (et comment j'ai réduit son bruit au silence)

78 drôles de drones

quelques modèles surprenants

 bientôt sur ces pages

Filtre d'aiguillage actif

Ce filtre à trois voies d'usage universel permet, entre autres, de réaliser un système d'enceintes acoustiques actives.

Alimentation modulaire

Une alimentation à conversion CC/CC et séparation galvanique, que l'on peut configurer de différentes manières pour l'adapter au mieux à ses besoins, par exemple pour faire 3,3 V / 0,5 A ou ± 15 V / 0,33 A. Comme source, un adaptateur secteur de portable de 19 V fait l'affaire.

Conditionneur de thermostat pour frigo sur carte d'E/S pour Android

N'importe quel lecteur d'Elektor est capable de remplacer le thermostat de son réfrigérateur. Mais trouver le modèle adéquat, c'est une autre histoire. Procurez-vous un thermostat universel à bon compte et, muni de votre Android avec l'extension d'E/S, vous le mettrez au point (*de congélation*).

Sous réserve de modification.

Le numéro de novembre paraîtra le 20 octobre.

Elektor : votre guide

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

57

pays

246833

membres actifs

10

experts &



elektor.post

L'hebdo d'Elektor

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

www.elektor.com/newsletter



communauté Elektor

Devenez membre, Green ou Gold

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

www.elektor.com/memberships



elektor.TV

Pour y voir plus clair

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

www.youtube.com/user/ElektorIM



Elektor PCB Service

Des cartes à la carte

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

www.elektorpcbservice.com



elektor.labs

découvrir, créer & partager

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

www.elektor-labs.com



elektor.academy

À cheval sur la courbe d'apprentissage

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

www.elektor-academy.com

de connexion(s)

31

479

233628

14:17

OCTOBER 06 2015

auteurs

publications

visiteurs (mois)

date de référence



elektor.magazine

Plus de 500 pages d'électronique inédite chaque année

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

www.elektormagazine.fr



e-choppe Elektor en ligne

Votre panier d'achats pour l'électronique

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

www.elektor.fr



livres et DVD Elektor

La puissance de l'information

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

www.elektor.fr

Formule GREEN

92,50 € par an

- ✳ 10 x magazine imprimé
- ✓ 10 x magazine numérique
- ✓ l'accès à l'archive d'Elektor
- ✓ 10 % de remise dans l'e-choppe
- ✓ l'accès à Elektor.labs
- ✓ un DVD annuel
- ✓ des offres exclusives
- ✓ 26 nouveaux projets inédits

www.elektor.fr/formule-greencard

Formule GOLD

127,50 € par an

- ✓ 10 x magazine imprimé
- ✓ 10 x magazine numérique
- ✓ l'accès à l'archive d'Elektor
- ✓ 10 % de remise dans l'e-choppe
- ✓ l'accès à Elektor.labs
- ✓ un DVD annuel
- ✓ des offres exclusives
- ✓ 26 nouveaux projets inédits

www.elektor.fr/formule-goldcard

Formule gratuite !

- ✳ 10 x magazine imprimé
- ✳ 10 x magazine numérique
- ✳ l'accès à l'archive d'Elektor
- ✳ 10 % de remise dans l'e-choppe
- ✳ l'accès à Elektor.labs
- ✳ un DVD annuel
- ✓ des offres exclusives
- ✓ 26 nouveaux projets inédits

www.elektor.fr/inscription

3 formules pour rester connecté avec Elektor !

bienvenue dans la section **DÉCOUVRIR****Jens Nickel**

La Foire internationale de l'Électronique grand public (IFA) de Berlin a déjà été largement couverte en long et en large par de nombreuses autres publications. J'avais gardé le souvenir de ma dernière visite (il y a quelques années déjà) d'immenses halles avec d'immenses téléviseurs. De très jeunes hommes et femmes se tenaient devant eux, leurs connaissances techniques étaient plutôt limitées. Du point de vue purement rédactionnel, 2015 n'allait probablement pas apporter grand-chose, en tout cas, c'est ce que je pensais.

J'ai été agréablement surpris : les halles immenses sont toujours là, mais elles sont « peuplées » de machines à café, aspirateurs et barbecues (naturellement, tous intelligents). J'ai été encore plus ébahi par la halle 11.1 : une journée ne suffisait pas pour découvrir toutes les nouveautés étonnantes du monde de l'électronique proposées par de petites et très petites entreprises.

Une nouvelle fois, le voyage à Berlin en valait la peine ! La startup HZO parvient à imperméabiliser complètement des circuits électroniques grâce à une fine couche d'enduit en plastique. Sur leur stand, une carte RPi plongée dans un aquarium et un ordiphone dans une baignoire ont démontré l'efficacité du produit. L'appareil modifié reste fonctionnel : il est possible d'utiliser un iPhone comme caméra sous-marine sans problème. L'astuce : le circuit électronique est extrait du boîtier, recouvert d'enduit puis remis en place. Ce type de service est (malheureusement) trop cher pour avoir de vraies chances sur le marché. HZO recherche plutôt des entreprises partenaires qui recouvriraient leurs circuits électroniques en masse en usine (www.hzo.com).



Panono est le nom d'une balle recouverte de 36 caméras, dont nous avons déjà parlé dans notre lettre d'information hebdomadaire Elektor.Post. La jeune pousse berlinoise s'apprête à entrer sur le marché de masse. Les photos panoramiques de 108 mégapixels peuvent être visionnées sur PC, tablette ou avec des lunettes de réalité virtuelle (www.panono.com).



Ces lunettes encombrantes étaient partout sur l'IFA. Le Fraunhofer IIS leur ajoutait même des écouteurs pour une expérience complète (www.fraunhofer-cingo.com/de.html). Les sources sonores (librement disposées dans la pièce) conservent leur place dans l'image comme si elles étaient fixes, même si on bouge la tête. Pour faire la démonstration, un vrai film était projeté dans un cinéma virtuel. Les box de cette salle de cinéma virtuelle donnaient un véritable son *surround*. Quoi de plus attrayant pour un fan de home cinéma comme moi ?



On connaît déjà l'utilisation de la tablette comme second écran, par exemple lors d'un match à la télé. L'appli Ultra-HD-Zoom du Fraunhofer HHI peut automatiquement suivre l'un des joueurs (dans cet exemple Bastian Schweinsteiger). Pendant la troisième mi-temps, on peut affirmer en tant qu'expert si son joueur préféré a mouillé le maillot, ou non.

Éducatif

Sur l'IFA, le stand commun aux associations d'électriciens, électroniciens et électrotechniciens (VDE, ZVEI et ZVEH) s'adressait aux électroniciens en herbe avec une maquette de maison automatisée, un atelier de soudage et une présentation de robots Lego entre autres (www.zvei.de). ◀

(150475 – version française : Thierry Destinobles)

trucs et astuces

les lecteurs écrivent aux lecteurs



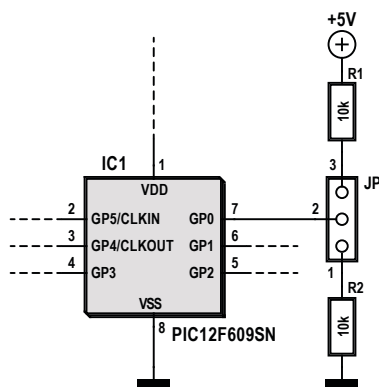
Encore des solutions fûtées qui facilitent la vie des électroniciens.

Jamais assez de lignes de port ?

Martin Oppermann



On utilise souvent des cavaliers pour configurer un circuit à microprocesseur. C'est pratique lorsqu'il ne faut paramétrer que deux états possibles: Un cavalier (fermé ou ouvert) sur une broche détermine sa valeur. Un deuxième cavalier est nécessaire s'il faut distinguer trois états, donc une deuxième broche est sacrifiée. Il y a quelques années, un ingénieur d'applications de Microchip a donné une astuce qui devrait fonctionner à tous les coups sur les PIC : une broche seule peut prendre l'un des trois états haut, bas ou ouvert (figure ci-dessous). La procédure suivante permet de déterminer l'état de la broche :

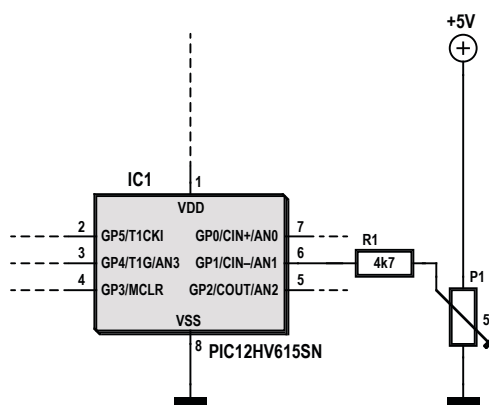


- Régler la broche en sortie.
- Écrire un « 1 » sur la broche.
- Régler la broche en entrée.
- Lire l'état de la broche et enregistrer la valeur.
- Régler la broche en sortie.
- Écrire un « 0 » sur la broche.
- Régler la broche en entrée.
- Lire l'état de la broche et ajouter la valeur précédemment enregistrée.

Si le résultat de l'addition est 0 (0+0), la broche de port est à l'état bas (low). Si le résultat de l'addition est 2 (1+1), la broche est à l'état haut (high) (1+1). Cependant, si le résultat de l'addition est 1 (1+0), la broche est ouverte (flottante), ce qui corres-

pond au troisième état possible (afin de ne pas tirer trop de courant, une broche de port ouverte doit rapidement être réglée en sortie pour prendre un niveau défini).

Tout cela est possible grâce aux capacités parasites toujours présentes, à la résistance d'entrée élevée des entrées du μC et au nombre réduit de commandes nécessaires. Le code est traité avant que les capacités parasites soient déchargées.



On peut également appliquer cette méthode à une entrée analogue. Je l'ai mise en œuvre pour un régulateur de température (voir figure ci-dessus). Si un potentiomètre est connecté au port analogue, la température de consigne est donnée par ce potentiomètre ; si la broche de port GP1 est libre (sans P1), la température de consigne sera une valeur fixe.

La détermination de l'état de la broche GP1 se fait de la même manière que dans l'exemple précédent. Si un potentiomètre est connecté, une tension fixe (entre 0 V et 5 V) est appliquée à la broche GP1 (selon la position du potentiomètre), broche déclarée comme entrée numérique pendant la procédure. La caractéristique du trigger de Schmitt des E/S numériques permet de toujours lire un état haut ou bas. Ce n'est que lorsque l'entrée est ouverte que l'on obtient le résultat « 1 » après l'addition des deux valeurs lues. ◀

(150479 - version française : Thierry Destinobles)

Vous avez une solution fûtée 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 € !



« The invisible touch ! »

µC ARM pour néophytes pour passer de 8 bits à 32 bits – 8^e partie

Viacheslav Gromov (Allemagne)

Un dernier sprint : l'ultime partie du cours de programmation ARM ! Quoi de mieux en guise d'au revoir que de décrire une dernière fonction que notre MCU maîtrise si bien ? Il s'agit des fonctions *QTouch* du SAM D20, pour lesquelles Atmel Studio fournit ici une superbe interface graphique. La mise en œuvre des fonctions tactiles est plus facile qu'il n'y paraît.

Pour cette partie, nous n'utiliserons pas de plaque d'essai, mais uniquement le kit QT1 Xplained Pro d'Atmel [8]. Il comporte deux cartes d'extension presque similaires avec deux boutons, une molette et un curseur, tous tactiles (*touch*) (figure 1). Il y a une LED jaune sur chaque bouton, pas moins de huit LED sur le curseur et une LED RVB au cœur de la molette. Toutes les surfaces tactiles à plusieurs électrodes sont, comme les LED, reliées à des embases femelles sur la gauche de la platine. On peut utiliser les LED pour visualiser l'état des surfaces tactiles,

rien ne vous y force d'ailleurs – elles ne sont pas reliées aux surfaces *Touch*. Les deux cartes d'extension du kit se distinguent uniquement par la technologie *QTouch* utilisée, spécifique à Atmel, soit *Self Capacitance* (à auto-capacité) soit *Mutual Capacitance* (à capacité mutuelle). Le routage des surfaces tactiles sur les couches des platines est différent, mais peu visible.

Nous allons rendre les deux cartes fonctionnelles de manière à ce que leurs LED respectives s'allument lors d'une action sur les surfaces tactiles. Lors de la « rotation » de la molette, la LED RVB doit changer sa couleur proportionnellement à l'angle. Pour cela, il faut commencer par télécharger les deux outils gratuits *QTouch Composer* et *QTouch Library* dans Atmel Studio 6 et les installer. Puis, étape par étape et avec l'aide d'une interface graphique, nous réaliserons le projet avec la carte *Mutual Capacitance*. Nous aborderons aussi un programme similaire pour la carte *Self Capacitance*. Commençons par nous intéresser au matériel et aux technologies *Touch*.

Contrôleur tactile périphérique

Le PTC (**P**eripheral **T**ouch **C**ontroller) est souvent présent sur les microcontrôleurs SAM D et AVR XMeta. Il se charge, quasi indépendamment de la CPU, des mesures *Touch*. Activé, ce périphérique fonctionne en parallèle et analyse en permanence les surfaces tactiles. Les encadrés montrent les différences entre les technologies *Mutual Capacitance* et *Self Capacitance*. En général, il travaille, de pair avec le logiciel, comme automate



Figure 1. Le kit QT1 Xplained Pro avec les deux cartes d'extension.

fini (*state machine*).

On trouve, en **figure 2**, le synoptique de ce périphérique un rien complexe dans le cas de notre SAM D20, pour les deux modes. On y voit, entre autres, les éléments *X Line Driver* et *Input Control* qui configurent les affectations des broches et commandent les multiplexeurs. *X Line Driver* n'existe pas dans la configuration *Self Capacitance*, car il n'y a qu'un seul canal Y par électrode tactile. En aval on trouve les éléments servant effectivement à l'évaluation et à la mise en forme du signal. On identifie une résistance numérique variable (potentiomètre) chargée d'atténuer le niveau de bruit, et un amplificateur de sortie. Il existe nombre d'autres paramètres et fonctions du CTP, mais leur description nous amènerait trop loin. La documentation [1] et le *QTouch Project Builder* (voir ci-dessous) les décrivent en détail.

Ce qui rend le CTP si intéressant est qu'il définit de lui-même au démarrage les paramètres les plus importants – il se calibre lui-même pour certains paramètres, ce qui évite à l'utilisateur bien des réglages.

Installation des outils QTouch

Voyons le logiciel. Commençons par installer les deux outils logiciels mentionnés plus haut dans Atmel Studio 6. Nous ouvrons Studio et sélectionnons sous *Tools*, l'*Extension Manager...*, déjà connu des mises à jour de l'ASF. On saisit, cf. la **figure 3**, en haut à droite, le mot « QTouch » dans le champ prévu pour la recherche des deux outils. On y voit différentes versions. Toujours installer la version la plus récente, la 6.5. à l'écriture de ces lignes. Avec une version plus récente, les étapes d'installation peuvent être différentes de celles décrites ici. Commencez par *QTouch Composer*. Sélectionnez-le et appuyez sur *Download*. Une fenêtre contextuelle d'identification s'ouvre (**figure 4**). Vous pouvez utiliser les mêmes données que celles de l'enregistrement lors du téléchargement de Studio (1^{ère} partie du cours). Si à ce moment-là vous vous êtes identifié comme *Guest* auprès de myAtmel, il vous faut refaire l'enregistrement ici en cliquant sur *New User?*. Si vous sélectionnez *Remember Me*, vous n'aurez plus à ressaisir vos données lors de l'installation de la bibliothèque QTouch.

Après l'enregistrement, l'*Extension Manager* télécharge *Composer* dans la foulée et l'installe. Entretemps il peut apparaître des messages de sécurité pour votre système d'exploitation, du genre de celui de la **figure 5**. L'outil ne présente pas de danger, vous pouvez autoriser des modifications. Après le téléchargement, mais avant l'installation, il apparaît un message dans lequel il est déjà paramétré que l'outil devra être intégré dans Atmel Studio. Il vous reste encore, par *Install*, à signifier votre accord avec le contrat de licence avant que ne commence le processus d'installation. Après un court laps de temps apparaîtra, espérons-le, le message *Installation Complete* (**figure 6**), que vous devez valider par *Close*. L'*Extension Manager* signale alors qu'il faut redémarrer Studio pour intégrer automatiquement et définitivement l'outil dans Atmel

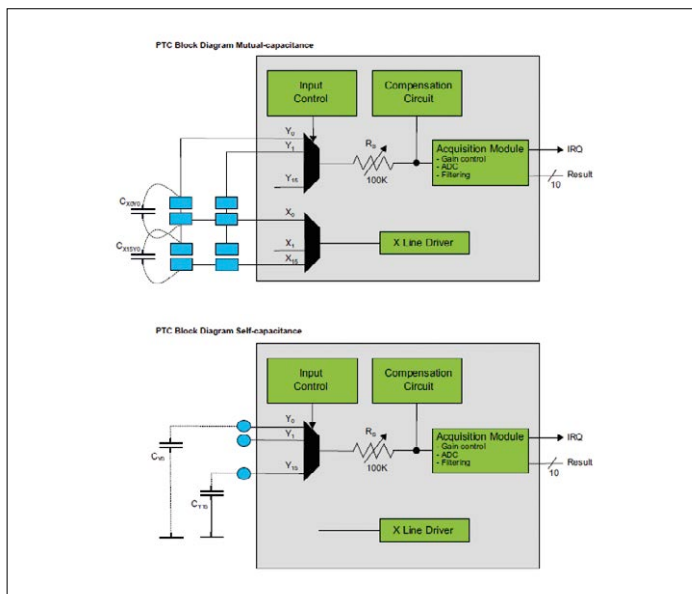


Figure 2. La structure de principe du contrôleur tactile périphérique (*Peripheral Touch Controller*) dans les modes *Self Capacitance* et *Mutual Capacitance* (diagrammes et captures d'écran : Atmel).

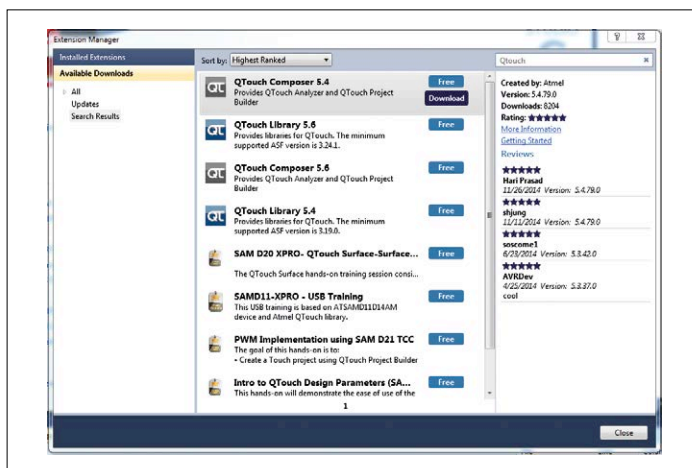


Figure 3. Lors de la recherche des deux outils, vous aurez vite fait de les trouver.

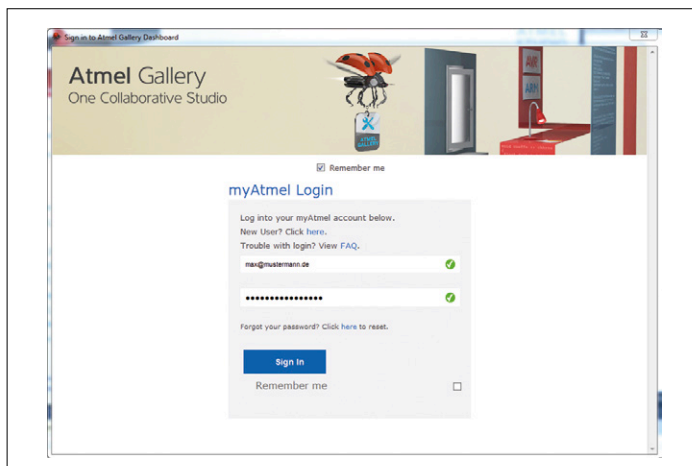
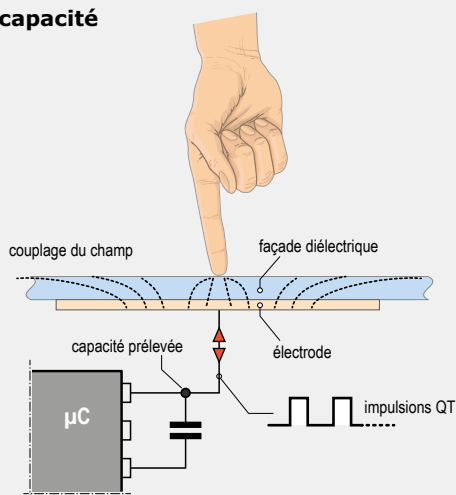


Figure 4. Ce n'est sans doute pas la première fois que vous voyez cette fenêtre de connexion (écran *login*).

Auto-capacité ou capacité mutuelle ?

Nous nous intéresserons ici d'une manière générale aux deux technologies QTouch de base d'Atmel, pour lesquelles il existe d'ailleurs différents circuits Touch, et plus en détail à ce qui les différencie.

Auto-capacité



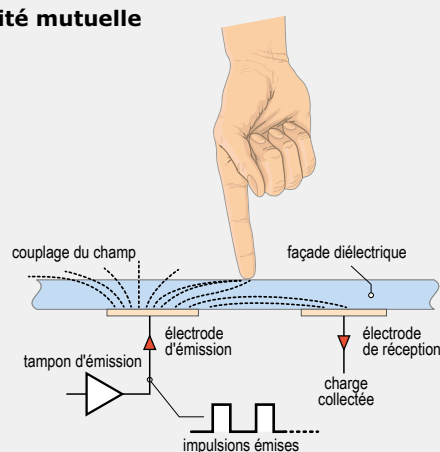
La technologie à auto-capacité (*self capacitance*) est très utilisée et simple ; elle a cependant quelques inconvénients. Il suffit, pour créer un bouton par exemple, d'une simple surface de cuivre tactile (*Touch*) connectée à une broche du μC . Le μC peut alors mesurer la capacité entre la surface tactile et la terre avec laquelle nous sommes toujours (plus ou moins) en contact, et donc enregistrer un toucher.

Cette technologie *Touch* ne requiert donc pas de dessin de platine complexe et se contente d'une seule broche du μC . Elle est simple, très polyvalente, mais relativement sensible aux interférences, ce qui explique la présence souvent d'un blindage autour des surfaces de contact et la nécessité d'un bon contact avec la terre de la part de l'utilisateur.

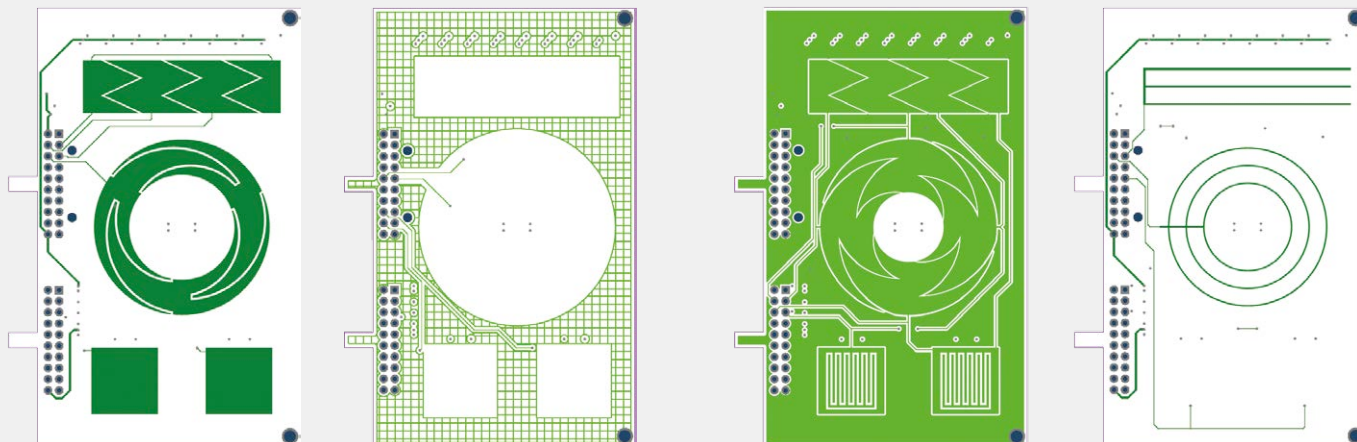
L'examen du dessin de la platine à auto-capacité du kit QT1 montre que les surfaces de contact ne sont reliées chacune qu'à une broche et qu'elles se résument à une simple surface de cuivre (vert foncé). À noter également la présence d'une couche pour le blindage ou la terre (vert clair).

Pour plus d'informations, voir [6], ou la fiche de caractéristiques [5], page 569.

Capacité mutuelle



La technologie à capacité mutuelle (*mutual capacitance*) d'Atmel est surtout très fiable et flexible. Une surface tactile (*Touch*) se compose de deux surfaces de cuivre ou de deux électrodes, une électrode Y et une électrode X. L'électrode X envoie un signal (périodique), l'électrode Y le reçoit. Un contact du doigt agit sur la capacité entre deux électrodes, changement détecté. Pour un bouton, il faut deux broches de MCU, mais on peut utiliser cette technologie dans un circuit en matrice, de sorte que les canaux X et Y sont reliés simultanément à une série de surfaces tactiles. Si cette technologie *Touch* requiert un dessin plus sophistiqué et plusieurs broches du μC pour un unique bouton *Touch* ; dans le cas de plusieurs surfaces tactiles comme avec des boutons ou des molettes, cette technologie peut créer une matrice de points, requérant alors même moins de broches que la technologie *Self Capacitance*. Cette technologie est en outre moins sensible aux interférences et peut fort bien être réalisée sur verre. Même une fine pellicule d'eau (humidité) ou les variations de température n'ont pratiquement pas d'effet. L'examen du dessin de la platine à capacité mutuelle du kit QT1 révèle rapidement l'agencement des surfaces X (vert clair) et Y (vert foncé). Pour plus d'informations, voir [7], ou la fiche de caractéristiques [5], page 568.



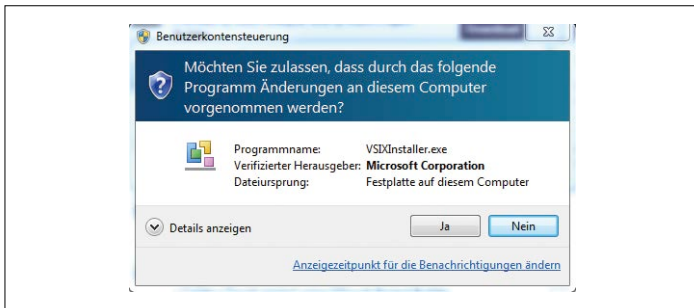


Figure 5. Selon le niveau de sécurité que vous avez sélectionné, il se peut que ce message Windows apparaisse (dans la langue de votre système).

Studio (*Restart Now*). Après le redémarrage, ouvrez l'*Extension Manager*, faites une recherche sur « QTouch » ; dans les résultats de recherche, deux symboles au niveau de *QTouch Composer*, vous informent du téléchargement et de l'installation.

Utiliser la même procédure d'installation pour la bibliothèque QTouch. Après un nouveau redémarrage de Studio, on voit s'afficher la page d'accueil QTouch (**figure 7**) avec les différentes bibliothèques QTouch pour, par exemple, des fonctions de sécurité spécifiques ou pour des µC sans contrôleur *Touch* de type matériel. Si ce n'est pas le cas, il vous faudra appeler manuellement cette page d'accueil par l'icône directement en dessous de *Help*. Tous les outils logiciels nécessaires sont maintenant installés.

Le premier projet Touch

Nous pouvons maintenant démarrer le premier projet, mettre en service la carte *Mutual Capacitance* et toutes ses surfaces tactiles. Il faut que lors d'une action sur les boutons la LED correspondante s'allume et qu'avec le curseur tactile, il s'en allume plusieurs, selon le point de contact. La molette est

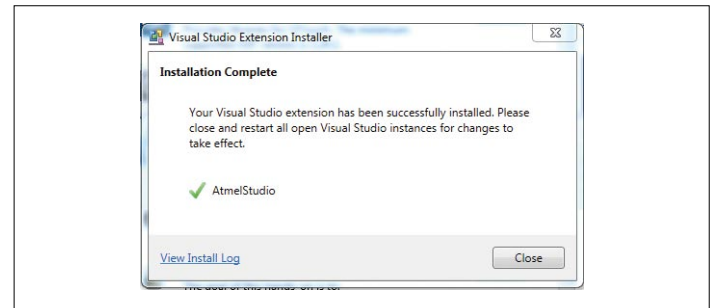


Figure 6. Vous avez réussi : l'installation de *QTouch Composer* s'est faite correctement. Au tour de la bibliothèque QTouch maintenant...

subdivisée « virtuellement » en trois angles de 120°, une des LED de la LED RVG doit s'allumer pour chacun d'entre eux. Les instructions de port classiques permettent de contrôler les LED, mais comment réalise-t-on la détection d'une action tactile avec *QTouch Composer* et *QTouch Library* ? *Composer* va produire le code pour les fonctions *Touch*, si au préalable nous choisissons à l'aide de l'interface graphique les éléments **Touch** nécessaires, les broches de la MCU reliées à ces éléments ainsi que les réglages du CTP voulus. Pour ce faire, on appelle l'Assistant de projet sous *File/New/Project...* Là, il faut sélectionner, non pas comme d'habitude *C ASF Board*, mais *GCC C QTouch Executable Project* et l'appeler *First test with QT1 MC* (**figure 8**) par exemple. Si, maintenant, vous cliquez sur *OK*, *QTouch Project Builder* apparaît. Là, vous pouvez par *Create QTouch Library Project* choisir la bibliothèque QTouch normale. Pour les autres bibliothèques, voir plus tard. Dans *QTouch Project Builder*, vous devez maintenant saisir successivement les informations requises par le projet. On commence d'abord par intégrer les éléments tactiles présents sur la carte en double-cliquant sur chaque élément dans la zone de travail (*QTouch Kit*) (**figure 9**). Tout d'abord les deux boutons, puis

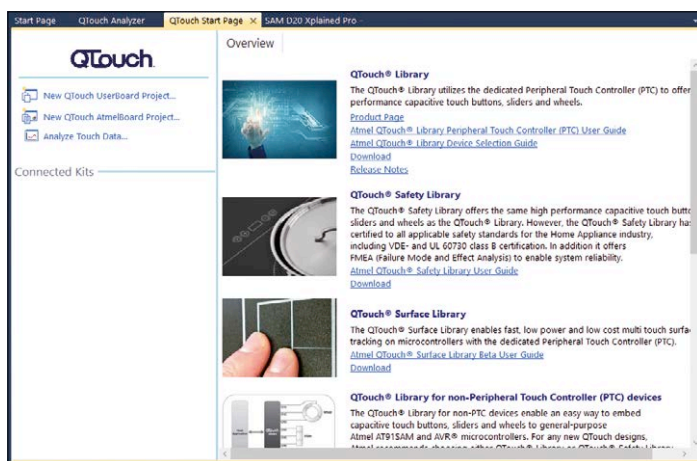


Figure 7. *QTouch Composer* vous salue cordialement.

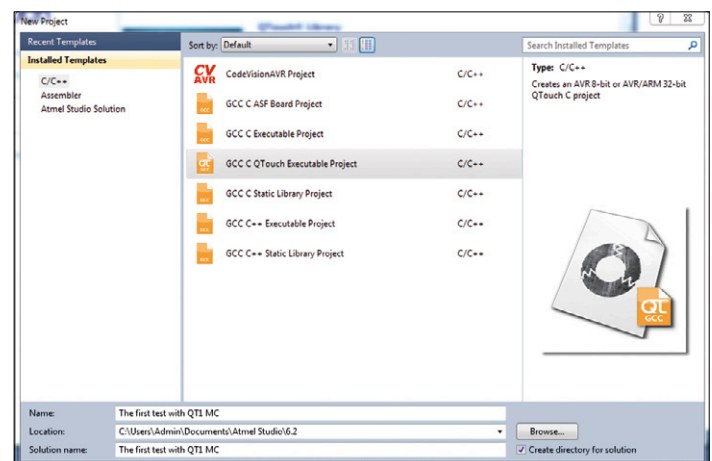


Figure 8. L'Assistant de projet ne vous est pas étranger ; cependant vous devez cette fois créer un projet QTouch.

L'Atmel Studio 7 nouveau est là !

Faisons la fête ! Le nouveau Studio, sa version 7 (en bêta) est là ! Vous pouvez le télécharger depuis <http://atmel-studio.s3-website-us-west-2.amazonaws.com/#> et l'essayer. Le nouveau Studio propose encore plus d'outils avec interface graphique ; vous vous y habituerez rapidement.

le curseur et enfin la molette. Une petite fenêtre qui demande les dimensions apparaît (**figure 10**). Saisissez pour chacun des éléments tactiles les dimensions du **tableau 1** (nécessaires uniquement pour la représentation graphique). Pas besoin cependant de placer les éléments exactement comme sur la carte – de l'à peu près, pour une bonne vue d'ensemble, suffit. Une fois tous les éléments souhaités sélectionnés, ouvrez l'onglet suivant *Device Selection* de *Project Builder*. On y sélectionne la MCU « SAMD20J18 » à utiliser (**figure 11**). La sélection faite, s'affiche en bas l'espace requis pour l'évaluation des éléments tactiles dans les deux modes ainsi que le nombre de canaux ou d'éléments tactiles que peut implémenter notre MCU. En *Mutual Capacitance* on peut, grâce à la structure en matrice, évaluer bien plus d'éléments tactiles. Dans ce projet, nous travaillons avec la carte *Mutual Capacitance* et cochons par conséquent le bouton radio tout en haut.

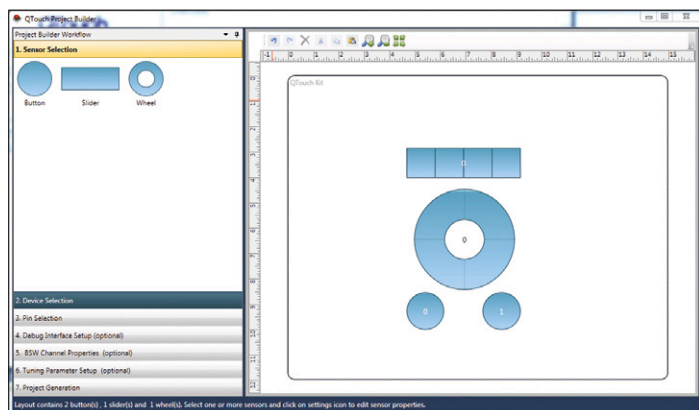


Figure 9. Le QTouch Project Generator graphique veut d'abord savoir quels sont les éléments tactiles que vous voulez scruter.

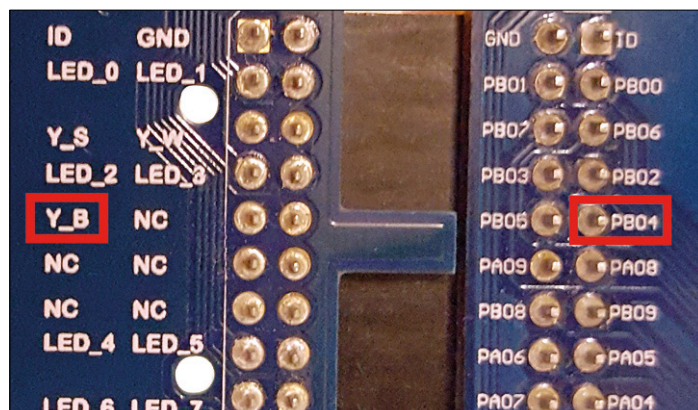


Figure 13. Vous pouvez ainsi lire sur le dos des deux cartes interconnectées quelle électrode Touch est connectée à quelle broche.

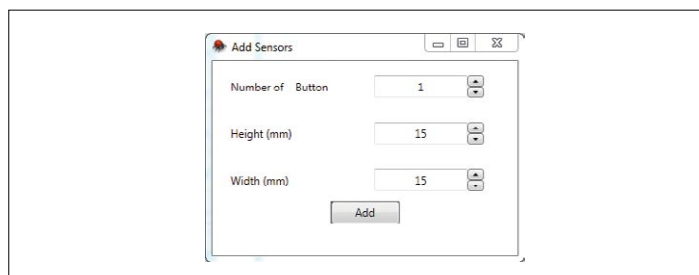


Figure 10. Les dimensions n'ont guère d'importance avec les petits projets, mais avec des projets plus gros, tout devient sensiblement plus clair.

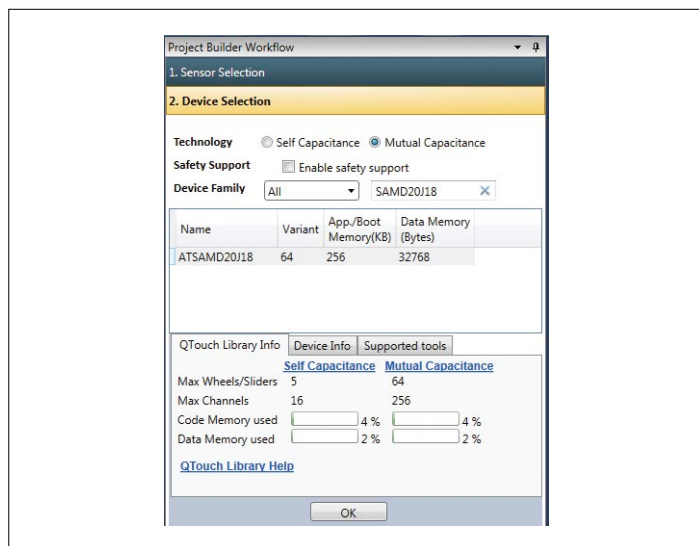


Figure 11. Ici vous pouvez, en bas à gauche, jeter un coup d'œil à d'autres caractéristiques de la MCU.

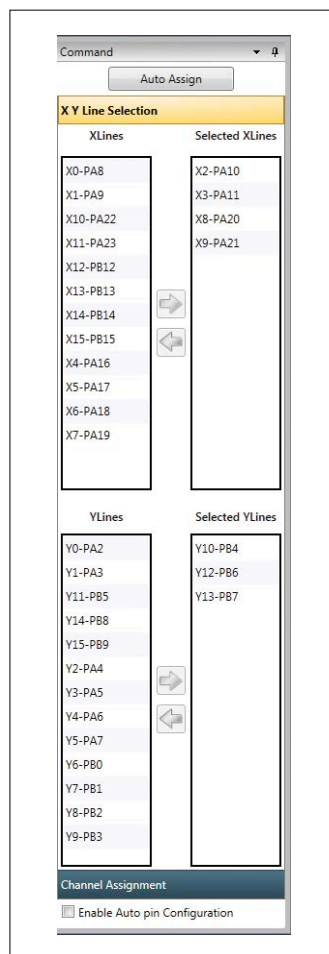


Figure 14. À droite, il vous faudra saisir avec soin la sélection des canaux utilisés et ...

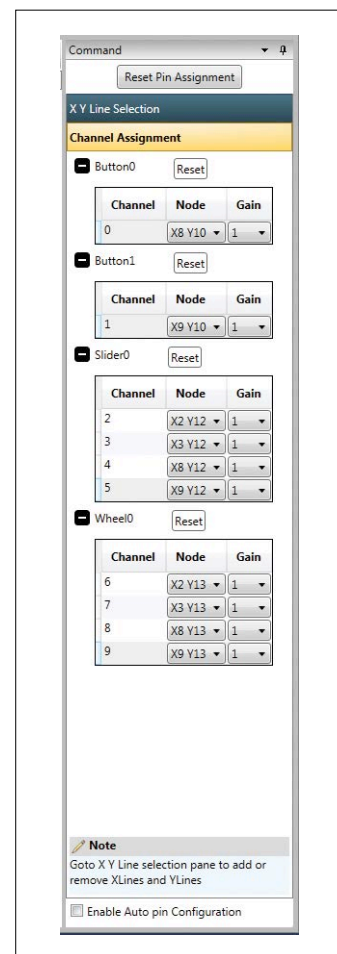


Figure 15. ... maintenant l'affectation de chaque électrode Touch.

Avec « OK » on passe à l'onglet suivant de sélection des broches. Ici, en deux étapes, on connecte les broches (canaux) aux différentes électrodes. Il faut d'abord sélectionner les canaux nécessaires et ensuite les affecter aux électrodes. Cependant vous devriez auparavant cliquer sur *Reset Pin Assignment* pour supprimer les affectations de broches proposées par Studio. Vous pouvez ensuite, par consultation des tableaux de la **figure 12** ou du manuel de l'utilisateur [2], trouver la corres-

pondance canaux-électrodes tactiles définie sur la carte QT1-MC. On peut lire, sur le dos de la platine des cartes interconnectées quelles sont les broches de la MCU connectées aux électrodes tactiles (**figure 13**). Les combinaisons de broche et de nom du canal mentionnées dans Atmel Studio ne correspondent pas à celles de la carte QT1-MC. PA10 par exemple est appelée X2 dans Studio, mais c'est en fait le canal X_1 de notre carte qui y est relié. Une fois la sélection de tous les canaux effec-

Tableau 1. Dimensions approximatives des éléments Touch de la carte QT1 Xplained Pro - mode capacité mutuelle

Boutons	15 mm x 15 mm
Curseur	45 mm x 12 mm
Molette	8 mm Øin x 20 mm Ø _{out}

Tableau 2. Dimensions approximatives des éléments Touch de la carte QT1 Xplained Pro - mode auto-capacité

Boutons	15 mm x 15 mm
Curseur	45 mm x 12 mm
Molette	12 mm Øin x 20 mm Ø _{out}

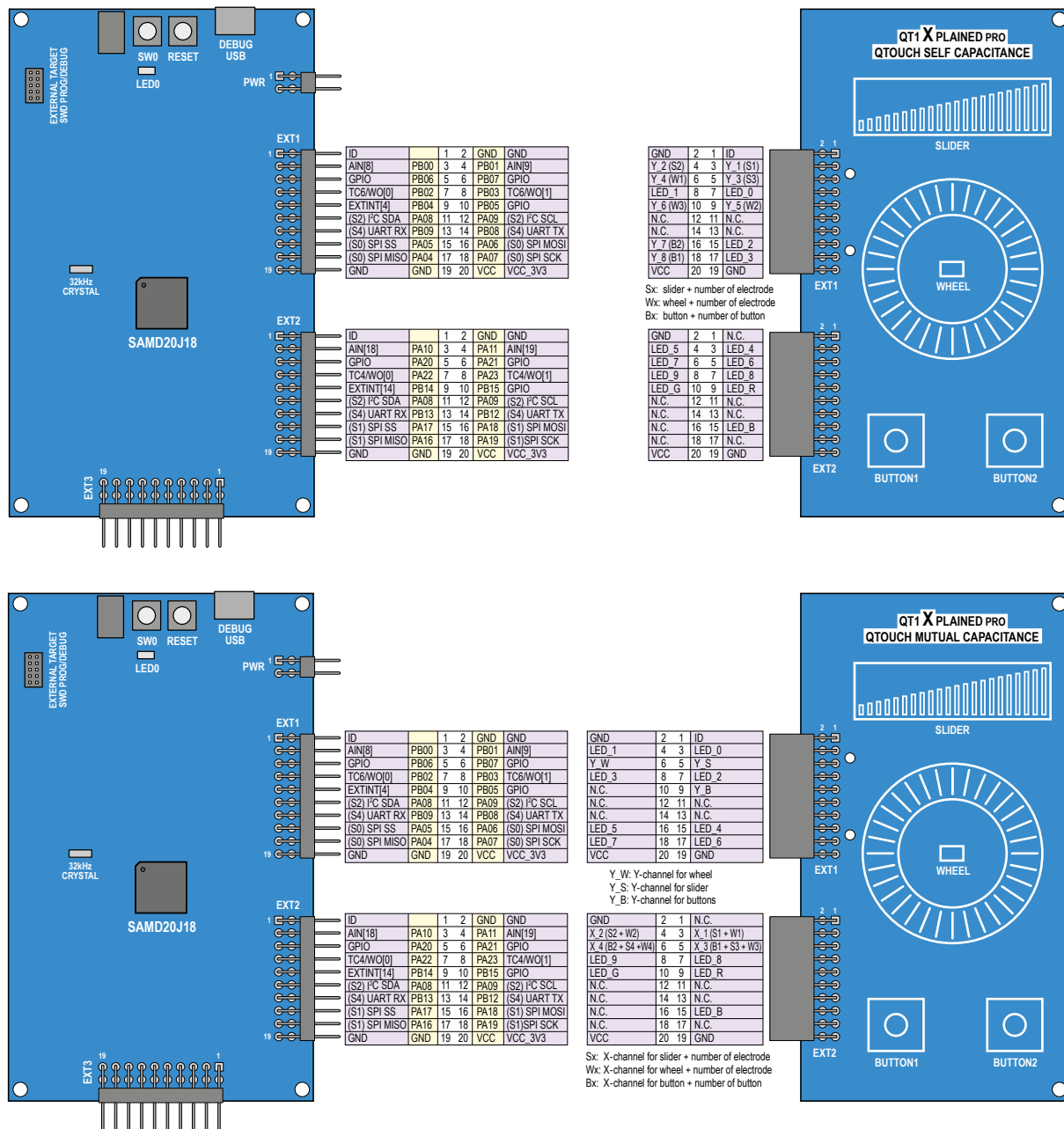


Figure 12. Les tableaux donnent les connexions des deux cartes QT1 Xplained Pro.

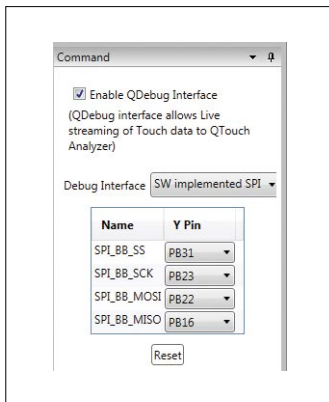


Figure 16. De cette façon, l'affectation des broches d'interface est effectuée correctement.

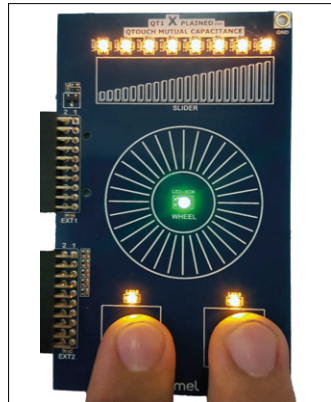
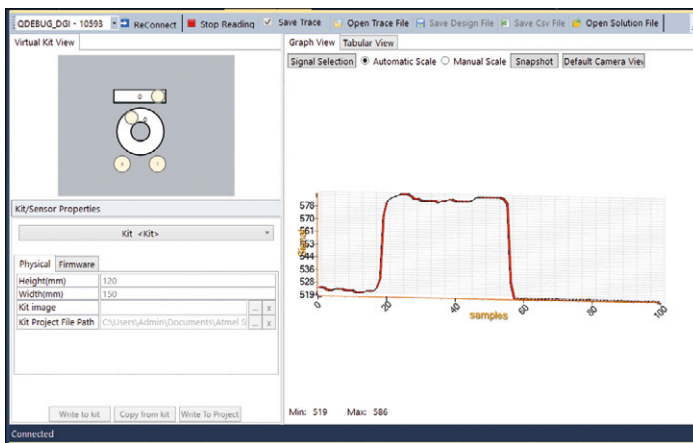


Figure 17. Les LED de la molette et du curseur s'allument aussi après que l'on ait relâché les éléments tactiles. On peut, si nécessaire, changer ce mode de fonctionnement.

Figure 18. Allure du signal en cas d'action brève sur le bouton 1.



tuée (**figure 14**), il vous faudra, dans l'onglet suivant *Channel Assignment*, utiliser les noms de canaux d'Atmel Studio à droite pour la sélection (**figure 15**). Pour la première électrode du curseur on choisira, par exemple X2 Y12. Après la sélection, vous pouvez passer à l'onglet suivant pour configurer l'interface de débogage du *QTouch Analyzer* (très utile plus tard avec les broches de l'EDBG correspondantes). L'outil pourra ainsi, plus tard, lire via l'EDBG les données de QTouch du moment. Le guide de l'utilisateur [3] mentionne les broches d'interface individuelles ; il faudra comme le montre la **figure 16** les sélectionner dans *QTouch Project Builder*.

Sur les deux onglets suivants, on procédera aux réglages du CTP. On commence par garder les paramètres par défaut. Dans le septième et dernier onglet, vous trouverez un résumé de tous les réglages réalisés dans *Project Builder* ; il ne reste plus pour créer le projet qu'à confirmer avec *Generate Project*. Il faut un certain temps ; le projet créé fini est ensuite affiché. Les fichiers **touch_api_ptc.h**, **touch.c** et **touch.h** sont les plus intéressants et contiennent l'API. Jetez-y un œil. Passons au code ! La détection des actions tactiles est déjà réalisée en fond de tâche ; il n'y a rien à y bricoler. Il ne reste

Listage 1. Constantes symboliques et variables requises.

```
#define LED0 PIN_PB00
#define LED1 PIN_PB01
#define LED2 PIN_PB02
#define LED3 PIN_PB03
#define LED4 PIN_PA05
#define LED5 PIN_PA06
#define LED6 PIN_PA04
#define LED7 PIN_PA07
#define LED8 PIN_PA22
#define LED9 PIN_PA23
#define LEDR PIN_PB14
#define LEDG PIN_PB15
#define LEDB PIN_PA17

void configure_port_pins(void);

bool sensor_state_button_1, sensor_state_button_2;
uint8_t sensor_state_wheel, sensor_state_slider;
```

Listage 2. Fonction de configuration du port pour les LED requises.

```
void configure_port_pins(void)
{
    struct port_config config_port_pin;
    port_get_config_defaults(&config_port_pin);

    config_port_pin.direction = PORT_PIN_DIR_OUTPUT;
    port_group_set_config(&PORTB, 49167, &config_port_pin);
    port_group_set_config(&PORTA, 12714224, &config_port_pin);
}
```

plus qu'à réaliser la fonction principale, l'allumage des LED. Il faut pour cela, après inclusion de **asf.h**, commencer par déclarer en début de fichier *Main* les constantes symboliques pour les différentes broches de LED QT1, un prototype de fonction pour la fonction de configuration des broches à venir et enfin les variables requises ultérieurement (cf. **listage 1**).

Avant d'en venir au corps du programme, nous devons, à la suite de toutes les autres fonctions de configuration qui ont été produites automatiquement dans le fichier *Main* pour QTouch, insérer la fonction de configuration pour les broches des LED (**listage 2**). Toutes les broches de LED y sont configurées en sortie. Malheureusement, les broches LED se trouvent sur deux ports différents ; il faut donc toujours les configurer et les commander l'un après l'autre. Il est très facile, tout au long du programme, de calculer toutes les valeurs et tous les masques des instructions de port (comme dans le cas des µC à 8 bits) sur la base de la valeur des différents bits. Les valeurs peuvent être données en binaire ou en hexadécimal. Cette fonction de configuration doit être appelée immédiatement après l'ensemble du code créé et avant la boucle :

Liens

- [1] www.atmel.com/Images/atmel-42195-qtouch-general-library-peripheral-touch-controller_user-guide.pdf
- [2] www.atmel.com/Images/Atmel-42193-QT1-Xplained-Pro_User-Guide.pdf
- [3] www.atmel.com/Images/Atmel-42102-SAMD20-Xplained-Pro_User-Guide.pdf
- [4] www.elektormagazine.fr/150368
- [5] www.atmel.com/Images/Atmel-42129-SAM-D20_Datasheet.pdf
- [6] www.atmel.com/products/TouchSolutions/bsw/qtouch.aspx
- [7] www.atmel.com/products/TouchSolutions/bsw/qmatrix.aspx
- [8] <http://fr.farnell.com/atmel/atqt1-xpro/carte-extension-xplained-pro-samd20/dp/2399861>

```
configure_port_pins();
```

Il faudra insérer dans le fichier *Main*, tout à la fin après le code QTouch produit et les commentaires, le code du **listage 3**. Initialement, on interroge, par le biais de `GET_MUTLCAP_SENSOR_STATE(x)` et de `GET_MUTLCAP_ROTOR_SLIDER_POSITION(x)` les états des différents éléments tactiles que l'on enregistre ensuite dans les variables souhaitées. La première macro peut interroger des boutons, la seconde des molettes et curseurs. Les macros ne requièrent que le numéro du capteur (boutons : 0 et 1 ; curseur : 0 et molette : 1). Elles fournissent une valeur booléenne pour les boutons et une valeur à 8 bits

pour les molettes et curseurs. Les valeurs sont stockées dans les variables mentionnées plus haut. Cependant, la valeur des curseurs subit parallèlement ici une division par $256/8 = 32$, et celle de la molette par $256/3 \approx 85$, de sorte que nous puissions piloter huit LED ou trois.

Les deux LED 8 et 9 sont, par le biais des lignes ci-après, commutées en fonction de l'inverse (affaire de câblage) de la valeur *Touch* des boutons :

```
port_pin_set_output_level(LED8,
!sensor_state_button_1);
port_pin_set_output_level(LED9,
!sensor_state_button_2);
```

Publicité

Personnalisez vos montages Arduino

techniques pratiques et fonctions avancées



L'objectif de ce livre est de vous emmener à pas guidés vers la maîtrise d'Arduino. Les projets, regroupés par thème, accompagnés de bases théoriques, sont des applications concrètes : chenillard à LED, voltmètre, thermomètre numérique, horloges sous différentes formes, ou encore bras de robot commandé par la souris.

Vous apprendrez ainsi à exploiter des techniques essentielles comme la conversion analogique-numérique, la modulation de largeur d'impulsion, ou encore les interruptions. Après avoir mené à bien tous ces projets vous maîtriserez les fondamentaux de la technique des microcontrôleurs.

272 pages | ISBN 978-2-86661-191-0 | 34,50 €



www.elektor.fr/arduino

Ne suivent plus que deux interrogations de commutateurs simples. La première commute les LED 0 à 7 en fonction de l'état de la valeur du curseur dans `sensor_state_slider`, la seconde commute la LED RVB en fonction de la position de la molette dans `sensor_state_wheel`.

Listage 3. Programme principal avec l'analyse des surfaces tactiles et la commutation des LED correspondantes.

```
sensor_state_button_1 = GET_MUTLCAP_SENSOR_STATE(0);
sensor_state_button_2 = GET_MUTLCAP_SENSOR_STATE(1);
sensor_state_slider = GET_MUTLCAP_ROTOR_SLIDER_POSITION(0) / 32;
sensor_state_wheel = GET_MUTLCAP_ROTOR_SLIDER_POSITION(1) / 85;
port_pin_set_output_level(LED8, !sensor_state_button_1);
port_pin_set_output_level(LED9, !sensor_state_button_2);
switch (sensor_state_slider)
{
    case 0:
        port_group_set_output_level(&PORTB, 15, 14);
        port_group_set_output_level(&PORTA, 240, 240);
        break;

    case 1:
        port_group_set_output_level(&PORTB, 15, 12);
        port_group_set_output_level(&PORTA, 240, 240);
        break;

    case 2:
        port_group_set_output_level(&PORTB, 15, 8);
        port_group_set_output_level(&PORTA, 240, 240);
        break;

    ...
}

switch (sensor_state_wheel)
{
    case 0:
        port_pin_set_output_level(LED8, 0);
        port_pin_set_output_level(LED9, 1);
        port_pin_set_output_level(LED0, 1);
        break;

    case 1:
        port_pin_set_output_level(LED8, 1);
        port_pin_set_output_level(LED9, 0);
        port_pin_set_output_level(LED0, 1);
        break;

    case 2:
        port_pin_set_output_level(LED8, 1);
        port_pin_set_output_level(LED9, 1);
        port_pin_set_output_level(LED0, 0);
        break;
}
```

Il est temps maintenant de compiler le projet. Éventuellement, il apparaît un message d'erreur signalant l'absence des fichiers `power.h` et `reset.h`. Doter alors `#include <power.h>` et `#include <reset.h>` du fichier `asf.h` d'un // de commentaire. Maintenant, vous devriez mettre la carte en fonction comme en **figure 17** et pouvoir l'essayer.

Vous pouvez en outre démarrer l'outil *QTouch Analyzer* par l'icône appropriée d'Atmel Studio. Dans la fenêtre, sélectionnez votre carte en haut à gauche et cliquez ensuite sur « *Connect* ». Il apparaît une autre fenêtre de configuration, dans laquelle il vous suffit de cliquer sur « *OK* ». L'outil est maintenant relié à la MCU par le biais de l'EDBG, aux fins de récupérer les données *Touch*. Vous pouvez, en haut au centre, sous « *Signal Selection* », sélectionner les valeurs que vous souhaitez suivre. Dotez par exemple, sous « *Signal* », « *Button0* » d'une coche et lancez la mesure par « *Start Reading* » en haut à gauche. Le signal *Touch* relatif de *Button_1* peut être observé dans le diagramme en direct (*Graph View*) ou dans le tableau (*Tabular View*) (**figure 18**). Avec cet outil, vous pouvez aller relativement loin dans le cœur du CTP et, avec le recul, peut-être perfectionner quelques réglages dans *QTouch Project Builder*, que vous pouvez appeler à tout moment par un clic droit sur votre projet dans l'explorateur de solutions. Le projet fin prêt est également disponible au téléchargement [4].

Auto-capacité

Vous pouvez maintenant mettre en œuvre le même projet avec la carte *Self Capacitance* ; le logiciel ne change pratiquement pas (instructions similaires, les commentaires produits automatiquement dans la boucle sans fin peuvent aider). Dans le *QTouch Project Builder*, la conversion est un peu plus simple, car il n'y a pas cette fois deux canaux auxquels sont connectées les surfaces tactiles.

En utilisant *QTouch Project Builder*, créez un nouveau projet appelé « *First test with QT1 SC* », où, cette fois, vous opterez pour *Self Capacitance* et où il ne vous faudra affecter qu'une seule broche (canal Y) aux électrodes tactiles. Le code est très proche ; il vous faudra cependant modifier plus spécifiquement les instructions *Touch* et les broches de LED. Mais vous pouvez également vous simplifier la vie et examiner directement le projet prêt à l'emploi sous [4].

Chaque fin est source d'un début !

Espérons que ce cours vous aura permis de découvrir la substantifique moelle de la famille ARM et que vous pouvez maintenant vous en servir pour vos propres projets. À vous la parole maintenant ; après tout, il est toujours possible d'améliorer les choses. Qu'avez-vous aimé dans le cours ? Comment pourrions-nous faire mieux la prochaine fois ? Ou aimeriez-vous nous présenter votre propre projet ARM ?

Prenez la plume : redaction@elektor.fr ! ◀

(150368 – version française : Guy Raedersdorf)

VFD ou afficheur fluorescent drôle de composant n°20

Neil Gruending (Canada)

Dans cette rubrique, j'ai déjà abordé quelques types de tubes, en voici un autre, le VFD, pour *Vacuum Fluorescent Display*. Sans vraiment y prêter attention, nous en voyons tous les jours dans différentes situations.

Le premier afficheur fluorescent à sept segments date des années 60. Avec sa basse consommation à faible tension et sa longue espérance de vie, il devait remplacer les tubes Nixie. Depuis lors, les technologies à LED et cristaux liquides ont pris la relève dans bien des domaines, mais les VFD sont encore largement répandus dans les appareils de mesure, l'électronique grand public et les applications qui requièrent une durée de vie longue et un grand angle de vue. Ils font aussi merveille dans le domaine automobile par leur solidité et leur tenue en température. On les distingue facilement à leur teinte bleu vert caractéristique comme à la **figure 1**, mais il y en a de différentes couleurs. Un VFD ressemble au tube triode, il a une cathode à filament, une ou plusieurs grilles et des anodes recouvertes de phosphore qui s'éclairent sous le flux d'électrons. La **figure 2**, reprise de [1], illustre leur disposition. Les électrons émis par le filament sont accélérés vers les anodes par une tension positive sur la grille ou bloqués quand la grille est négative. Le champ électrique produit par les anodes positives attire les électrons qui y font scintiller le phosphore.

La structure de triode du VFD demande souvent des circuits de commande plus complexe que dans les autres technologies. La principale difficulté pour les constructeurs, c'est d'égaliser la tension le long du filament d'un bout à l'autre de l'afficheur pour obtenir une luminosité uniforme. On peut alimenter le filament en courant continu, mais la chute progressive de tension réduit d'autant la luminosité de l'affichage. La solution préconisée est d'alimenter le filament en alternatif au moyen d'un transformateur à prise médiane, laquelle est mise à la masse. La variation d'intensité est alors diminuée de moitié, c'est appréciable sur-



Figure 1. Le VFD du Fluke 8808A.

tout sur les grands affichages.

On commande grilles et anodes d'un VFD par des tensions continues qui souvent dépassent les niveaux logiques des CMOS, une cause de soucis. Certains petits VFD n'ont qu'une seule connexion de grille, ce qui oblige à câbler toutes les anodes. Ils sont dits afficheurs statiques parce que la totalité de l'affichage est permanent, la grille étant toujours active, ce qui simplifie le schéma des circuits tout en réduisant les tensions d'anodes et de grille pour une luminosité convenable. Mais de nombreux VFD sont plus complexes, il leur faut des commandes dynamiques ou en multiplex. Ils sont dotés d'anodes partagées entre plusieurs grilles qui sélectionnent par groupes les anodes qui doivent être activées. Cela nécessite des circuits de commande plus élaborés et des tensions plus hautes que sur les modèles statiques pour atteindre la même intensité, du fait que chaque grille n'est active qu'à temps partiel.

Heureusement, on trouve des modules VFD compatibles CMOS et il existe même des puces de commande spéciales pour VFD. Vous n'aurez plus aucune excuse d'écarter les VFD de vos nouveaux projets ! ◀

(150484 - version française : Robert Grignard)

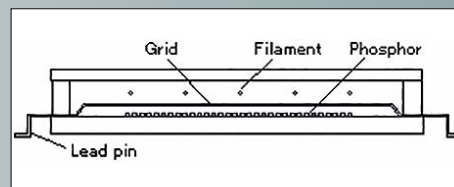


Figure 2. La structure de base d'un VFD. Source : [1]



Lien

[1] www.noritake-elec.com/display/vfd_operation.html



Envoyez-moi vos idées en quelques lignes ou sous forme d'un article déjà rédigé : neil@gruending.net

trucs & astuces pour Eagle (2)

créer une nomenclature avec ULP



Neil Gruending (Canada)

Au menu aujourd'hui :
comment accéder aux données d'un schéma et créer une nomenclature avec ULP.

Dans l'article précédent, j'avais choisi un exemple très simple pour vous donner un aperçu de la syntaxe du langage utilisateur ULP (*User Language Program*) d'Eagle, et également souligner la ressemblance avec le C, un avantage pour qui connaît déjà ce langage. Je vais maintenant vous montrer comment utiliser les objets de données ULP, puis comment lire les données d'un

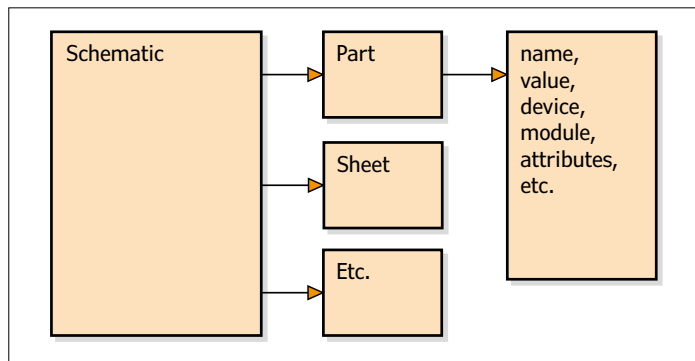


Figure 1. Hiérarchie de l'objet *schematic* d'Eagle.

schéma à partir de la nomenclature ULP d'Eagle.

Objets de données

Un objet de données peut être vu comme un conteneur qui regroupe et stocke toutes les informations sur un objet. La **figure 1** montre un exemple d'objet de type *schematic*. Ce schéma est formé d'autres objets, dont un objet de type *sheet* (page de schéma) et de plusieurs objets de type *part* (composant), chacun caractérisé par son nom et ses attributs. Qu'un objet puisse contenir d'autres objets permet de les hiérarchiser. Eagle possède également des objets de type *library* pour accéder aux bibliothèques de composants, ainsi que des objets de type *board* pour accéder aux dessins d'une carte et à leurs propriétés. Ces objets sont décrits en détail dans la documentation ULP d'Eagle. La manipulation d'objets se fait avec une syntaxe légèrement différente de celle vue précédemment. Voyons d'abord comment afficher tous les composants d'un schéma (**listage 1**).

Listage 1

```

schematic(sch) {
    sch.allparts(part) {
        printf("%s\n", part.name);
    }
}

```

La première ligne demande à Eagle de créer un objet de type *schematic* appelé *sch*, nom utilisé ensuite dans les instructions comprises entre la première et la dernière accolade ; ce bloc limite la portée de l'objet *sch*, ce qui évite toute confusion lorsqu'il y a des références multiples au schéma.

L'instruction suivante, *sch.allparts()*, est pour Eagle un membre de boucle (*loop member*). Cet itérateur, comme l'appellent d'autres langages, permet de parcourir une à une toutes les composantes d'un objet. Ici *allparts()* parcourt tous les objets de type *part* qui composent l'objet *sch*. L'instruction suivante (*printf*) est exécutée pour chaque composant trouvé, le composant est alors référencé dans *printf* avec le nom *part*. L'instruction *printf()* affiche le nom du composant. Les champs d'un objet de type *part* qui peuvent être référencés sont décrits dans la documentation (cf. *UL_PART*).

Programme de la nomenclature

Voyons maintenant comment créer une nomenclature (BOM, *bill of materials*). Le code est assez long, je n'en décrirai que les parties éclairant le mécanisme d'un « gros » programme ULP. Le **listage 2** montre le premier bloc d'instructions exécuté.

Listage 2

```

schematic(SCH) {
    sprintf(SeparatorString, "%c", Separator);
    CurrentVariant = variant();
    SCH.variantdefs(VD) {
        if (CurrentVariant == VD.name) VDsel = cntVD;
        sprintf(Variants[cntVD], "%s", VD.name);
        cntVD++;
    }
}

```

Ce bloc construit une liste de toutes les variantes d'assemblage du schéma. La fonction intégrée *variant()* est appelée pour récupérer le nom de la variante actuelle, stocké ensuite dans la variable *CurrentVariant*. L'itérateur *SCH.variantdefs()* parcourt toutes les variantes de schéma définies et les copie dans le tableau *Variants*. Le nom de la variante actuellement sélectionnée est aussi comparé à chaque nom de variante de schéma, et l'indice de tableau correspondant est stocké dans *VDsel*.

Contrairement à d'autres langages, en ULP la longueur des tableaux augmente automatiquement si nécessaire, ce qui est plutôt pratique. Le tableau *Variants* a ainsi d'abord une longueur de 0, puis sa taille augmente lorsque le code trouve

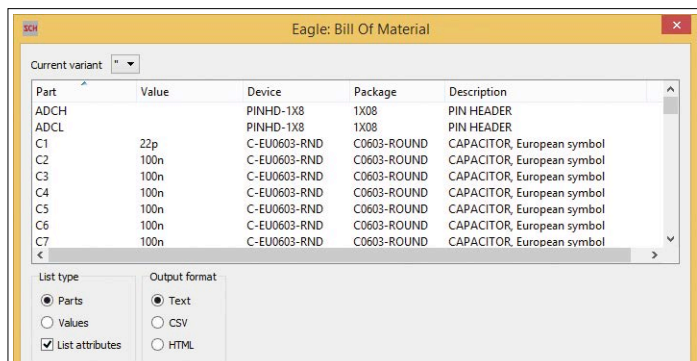


Figure 2. Affichage de la nomenclature.

un élément du tableau qui n'existait pas auparavant. Dans notre exemple, `cntVD` sert à garder une trace de la longueur du tableau `Variants`.

Une fois chargés tous les noms de variantes, la fonction `CollectData` charge toutes les informations sur les composants pour la variante en cours. Le **listage 3** montre comment cette fonction plutôt longue passe en revue tous les composants de la variante.

Listage 3

```
schematic (SCH){
    SCH.allparts (P){
        if (P.device.package){
            if (P.populate){
                PartName[NumParts] = P.name;
                PartValue[NumParts] = P.value;
                PartDevice[NumParts] = P.device.name;
                PartPackage[NumParts] = P.device.package.name;
                PartHeadline[NumParts] = P.device.headline;
                PartDescription [NumParts] = P.device.description;
                PartValueOn[NumParts] = P.device.value == "On";
                // save part attributes, etc
            }
        }
    }
}
```

Notez qu'ici le code n'utilise pas la fonction habituelle `parts()` pour parcourir tous les composants, mais la fonction `allparts()`. Celle-ci peut en effet manipuler un schéma hiérarchique, ce que l'on souhaite dans la plupart des cas. Pour cela la fonction `allparts()` crée des composants virtuels utilisés dans chaque instance de module, qui correspondent à des composants physiques sur la carte.

La première partie de la boucle teste le composant afin de vérifier qu'il possède bien une empreinte sur le circuit imprimé ; ainsi les composants faisant uniquement partie du schéma ne seront pas ajoutés à la nomenclature. La boucle teste ensuite le drapeau `P.populate` (`P` pour *part*) ; si le composant doit être monté, il est alors ajouté aux données de nomenclature enregistrées. Cette boucle parcourue, le programme crée toutes les listes de composants requises et les affiche au moyen d'un objet de type *boîte de dialogue* (**listage 4**).

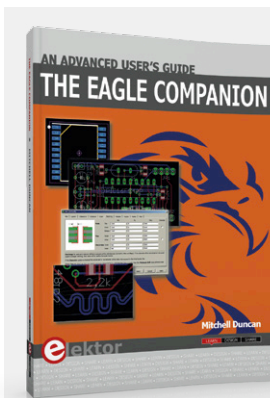
Listage 4

```
dlgDialog (tr ("Bill Of Material"))
{
    dlgHBoxLayout {
        dlgLabel(tr ("Current &variant "));
        dlgComboBox(Variants, VDsel) {
            CurrentVariant = Variants[VDsel];
            setvariant(CurrentVariant);
            CollectPartData(CurrentVariant);
            GenerateList();
        }
        dlgStretch(1);
    }
    dlgListView ("", Lines, Selected);
    // more dialog box setup...
    dlgHBoxLayout {
        dlgPushButton (tr ("&View")) ViewList ();
        dlgPushButton (tr ("&Save...")) SaveList ();
        dlgPushButton (tr ("H&elp")) DisplayHelp ();
        dlgPushButton (tr ("-Close")) dlgAccept ();
        dlgStretch(1);
        dlgLabel("Version " + Version);
    }
};
```

Ici on ne passe pas de nom de variable aux objets de type *dialogue*, mais une chaîne de caractères les décrivant, p. ex. le titre de la boîte de dialogue pour `dlgDialog`. La fonction `tr()` renvoie une traduction des chaînes de caractères utilisées si celles-ci existent dans le tableau prédéfini `I18N[]` d'Eagle, sinon elles restent non traduites. La fonction intégrée `language()` permet de détecter la langue du système (anglais par défaut).

La partie suivante construit la boîte de dialogue au moyen de conteneurs (*layouts*), boîtes combinées et boutons, comme avec Visual Basic par exemple. La fenêtre de dialogue est ensuite affichée (**fig. 2**), le bouton `Save...` permet d'enregistrer le fichier de nomenclature créé. ◀

(150360 – version française : Hervé Moreau)



Votre nouvel assistant : The EAGLE Companion !

Comme pour de nombreux livres techniques, le sous-titre annonce l'essentiel : *An Advanced User's Guide*. Exclusivité, ce livre (en anglais) propose en annexe le manuel complet EAGLE ULP, autrement dit l'essentiel de la documentation facilement accessible sur p-a-p-i-e-r. Une vaste collection de commandes et options Eagle

complète le cœur de l'ouvrage.

<http://www.elektor.fr/the-eagle-companion>

sac à malices

aux Olympiades de Sciences de l'Ingénieur

Elektor a soutenu les 6^e *Olympiades de Sciences de l'Ingénieur* [1] de l'académie de Rennes et récompensé les gagnants. Ce concours, ouvert aux élèves de première et de terminale des lycées d'enseignement général et technologique, est organisé par l'Union des Professeurs de Sciences et Techniques Industrielles (UPSTI) avec l'appui du Ministère de l'Éducation nationale.



L'objectif du concours est d'apprécier et de distinguer des projets expérimentaux pluritechnologiques dans le domaine des Sciences de l'Ingénieur, menés par des équipes de lycéens pendant plusieurs mois. Les projets répondent à des problèmes comme l'accès à l'eau, la transformation et le stockage de l'énergie, l'assistance aux personnes, le développement durable

Le travail des équipes de deux à quatre lycéens est trié en trois temps : présélection des meilleurs projets au sein de chaque établissement, finale académique (régionale) puis finale nationale. Lors de la finale académique, chaque équipe présente son projet à deux reprises devant des jurys distincts, constitués d'enseignants, d'inspecteurs de l'Éducation nationale, de chefs d'établissement, d'ingénieurs, de chercheurs, de chefs d'entreprises, de responsables d'unités de production...

Un sac très connecté

L'équipe gagnante dans l'académie de Rennes est issue du lycée Jean Guéhenno de Fougères (35), elle a présenté le LOCKing Bag. Il s'agit d'un sac qui permet de prévenir les tentatives de vol de matériel ou de données numériques.

Le propriétaire du sac LOCKing Bag est équipé d'un bracelet avec une puce RFID. Lorsqu'il ouvre son sac, le système électronique intégré au sac reconnaît la puce du propriétaire. S'il n'a pas le bon bracelet, une alarme est déclenchée et un

SMS avec la position du sac (longitude, latitude, lien *Google Maps*) est envoyé au propriétaire du sac.

En outre, la doublure du sac est renforcée par un blindage en aluminium qui permet de bloquer les ondes extérieures, donc

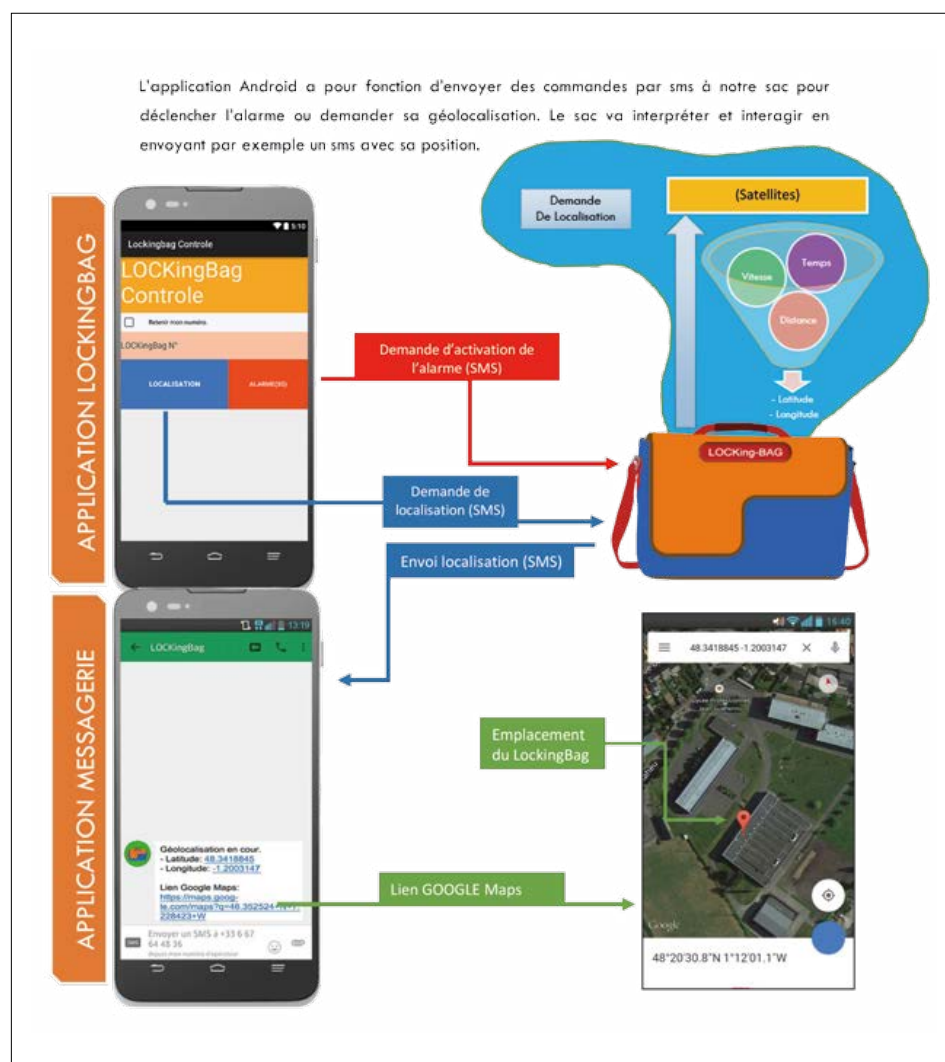


Figure 1. Fonctionnement de l'application Android.

d'empêcher le piratage d'un téléphone ou d'une carte bancaire sans contact (il y a quand même une poche sans blindage pour rester en contact avec le monde extérieur).

En cas de perte ou de vol du sac, une application Android permet à tout moment de déclencher l'alarme du sac et de demander sa géolocalisation.

Enfin, le sac est également équipé d'un système de recharge par induction pour entretenir la batterie qui alimente les modules électroniques. Il faut juste penser à poser le sac sur sa base pour le recharger. La base a été installée dans la partie supérieure d'une tour d'ordinateur, équipée d'une trentaine de LED allumées pendant la recharge et recouverte d'une toile grise. Une plaque en aluminium permet de transmettre le courant de recharge.

Le cœur du système électronique dans le sac repose sur une carte Arduino UNO ; elle est complétée par un module GPS (pour obtenir la position géographique du sac) et un module GSM (pour envoyer et recevoir des SMS). Si nos jeunes concepteurs avaient superposé les différentes cartes, comme c'est l'usage, le système aurait occupé un volume de 74 x 53 x 45 mm : c'est beaucoup trop pour la doublure du sac. Ils ont donc choisi de réaliser un support sur lequel sont alignés et soudés entre eux les différents modules (pas de fils de liaison). Cette « carte-support » à double couche accueille également des connecteurs pour les modules extérieurs (LED, capteur magnétique, etc.) ainsi que le module RFID (lecteur de puce).

Au départ, l'ouverture du sac était détectée par une photorésistance. Toutefois, pendant les tests, nos développeurs en herbe se sont rendu compte que l'ouverture n'était pas décelée lorsque la lumi-

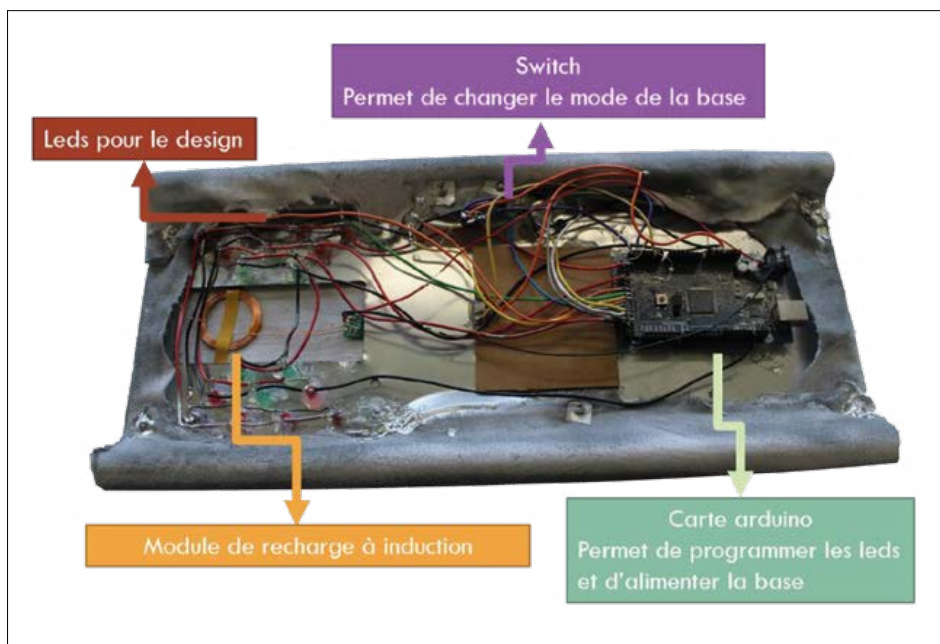


Figure 2. Base de rechargement également contrôlée par une carte Arduino.

nosité était trop faible. Le problème a été résolu par l'utilisation d'un capteur magnétique.

Pauline, Ronan et Yoann ont confié la réalisation sur mesure du sac et du bracelet à Margaux, également élève au lycée Jean Guéhenno, mais en BAC PRO Métiers de la mode, option Maroquinerie. Ils ont réfléchi ensemble à comment conserver un maximum de place dans le sac tout en intégrant les modules électroniques : un bel exemple de collaboration interdisciplinaire.

Toujours des idées

Cette équipe a déjà réfléchi aux améliorations à apporter à son projet : pour que le système d'alarme soit beaucoup plus discret, ils voudraient avoir recours à une montre connectée. Si le sac est trop éloigné de la montre (rupture de la liaison Bluetooth), la montre se met à vibrer et

demande si c'est normal.

Au lieu d'utiliser le réseau GSM (avec carte SIM) pour envoyer des commandes au sac, les lycéens voudraient employer le réseau SIGFOX. Il s'agit d'une alternative au GSM qui permet d'envoyer des messages courts (limités à 12 octets), avec la même couverture que le réseau GSM mais pour beaucoup moins cher.

Si, vous aussi, vous organisez des actions éducatives qui visent à développer chez les élèves l'esprit d'initiative, le goût pour la recherche et les études d'ingénieur, contactez-nous (redaction@elektor.fr).

(150525 - Mariline Thiebaut-Brodier)

Lien

[1] www.ac-rennes.fr/olympiadesSI

Critères du concours

Les critères d'évaluation sont : démarche de projet (planification) et travail collaboratif ; démarche interdisciplinaire ; modélisation, hypothèses et évaluation des écarts entre le souhaité, le simulé et le réalisé ; capacité à proposer des solutions novatrices ; réalisation d'un prototype fonctionnel ; qualité et pertinence des solutions techniques choisies ; maîtrise du sujet ; qualité

de la présentation et dynamisme des participants ; esprit d'initiative et esprit critique.

Les jurés souhaitent connaître la démarche suivie, les problématiques posées, les problèmes rencontrés, les méthodes et moyens mis en place pour résoudre ces problèmes... Chaque présentation dure dix minutes, suivies de cinq minutes d'échange avec les membres du jury.

FX3S et USB 3.0 : la solution anti-goulot d'étranglement

Abhishek Gupta (Cypress Semiconductor)

Pour illustrer les avantages que procurent un bus série normalisé comme USB 3.0, prenons l'exemple des caméras à haute ou ultra-haute définition : les données en temps réel qu'elles collectent doivent être traitées sur place, puis rapidement transférées vers des écrans ou des services de stockage distants. Même une caméra à 5 mégapixels et 24 images par seconde peut produire un flux de données aussi élevé que 2,4 Gbit/s. Une interface série plus ancienne comme USB 2.0 ou Wi-Fi 802.11n est incapable de traiter autant de données, encore moins de les transférer en temps réel vers d'autres périphériques du réseau utilisateur.

L'absence de voies rapides pour les données peut entraîner un surdimensionnement de la mémoire d'image ou encore l'ajout de supports de stockage. Cette complexité supplémentaire accroît le coût total et le temps de conception.

USB 3.0 répond parfaitement aux exigences de performance des systèmes embarqués actuels. Son architecture à double bus permet de communiquer avec les périphériques les plus anciens alors que le mode de transfert SuperSpeed atteindra un taux de transfert de 5 Gbit/s. L'efficacité énergétique de la norme 3.0 est par ailleurs trois fois supérieure à celle de la 2.0. Très répandu dans l'industrie, le mode SuperSpeed s'impose comme choix naturel pour la gestion des chemins de données internes et externes, d'autant plus qu'il combine au

mieux haute performance, consommation réduite et bas coût. Dans les caméras avec une architecture de type *système sur puce* (SoC), les données des capteurs à haute résolution sont collectées et traitées au moyen de puces spécialisées dans le traitement numérique des images. L'implantation de la **figure 1** utilise une interface SDXC pour le stockage local des images, typiquement sur des cartes mini/micro SD. Les ports HDMI s'occupent du transfert vidéo vers les écrans externes. Le SoC doit aussi gérer l'écran LCD embarqué. Le partage des données avec l'extérieur peut *a priori* se faire via les ports USB 2.0 et Wi-Fi, mais ces interfaces ne peuvent hélas pas suivre le débit imposé par les capteurs. Une solution peut être d'adapter le SoC, mais l'opération est complexe et demande du temps.

Bande passante

Certaines caméras sont pourvues d'une mémoire de stockage embarquée limitée, p. ex. les caméras de sécurité. Dans ce cas, ce sont des interfaces externes rapides de type LAN (câblé ou Wi-Fi) qui permettent de transférer les flux vidéo vers un serveur local. Certaines caméras grand public ont des cartes de stockage amovibles (mini/micro SD), mais pas de LCD intégré pour visionner le contenu filmé. L'utilisateur doit donc pouvoir transférer son enregistrement vers un périphérique externe, souvent au moyen d'une interface de type Wi-Fi, USB 2.0 ou Bluetooth.

Le Wi-Fi dépend fortement de la rapidité et de la fiabilité du réseau utilisé. Même un réseau 802.11n optimisé peut ne supporter qu'un débit de 300 Mbit/s.

La bande passante réelle du bus USB 2.0 n'est que de 280 à 320 Mbit/s (35 à 40 Mo/s), mais pour la plupart des périphériques le débit ne dépasse pas 200 à 240 Mbit/s. Les débits USB 2.0 réels dépendent beaucoup du pilote du logiciel et des optimisations apportées à la plateforme.

Le **tableau 1** indique les durées théoriques de transfert par Wi-Fi et USB 2.0 d'une vidéo de 30 min et de résolution 1080p HD (H.264), soit environ 18 Go de données. Dans certains cas la durée du transfert excède celle de la vidéo !

Les entreprises qui fabriquent des composants pour caméras ont deux solutions pour se débarrasser de ce goulot d'étranglement : concevoir eux-mêmes un ASIC de prochaine génération, ou attendre les SoC de prochaine génération.

Dans un cas comme dans l'autre, cela implique de retarder la production de 18 à 24 mois. Une échappatoire consiste à mettre à niveau la plateforme au moyen de composants USB 3.0 du commerce.

Leur mode SuperSpeed peut traiter les données dix fois plus vite qu'USB 2.0, ce qui en fait une solution idéale pour réduire les goulots d'étranglement liés aux transferts de flux vidéo. Le contrôleur **EZ-USB® FX3STM SuperSpeed USB** de Cypress [1] est ainsi équipé d'un processeur ARM9, d'un bus USB 3.0 et de deux ports de stockage (configurés en SDIO 3.0 ou eMMC 4.41) (**fig. 2**).

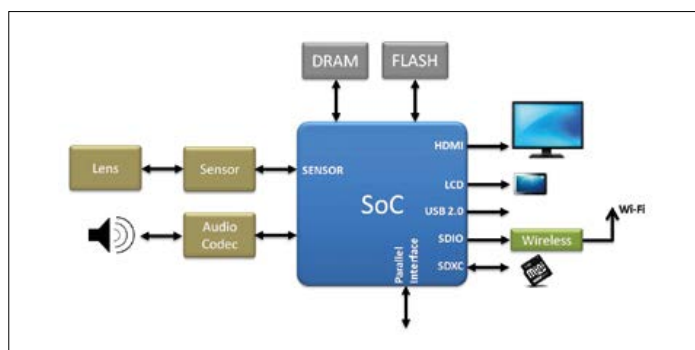


Figure 1. Schéma de principe d'une caméra vidéo.

Tableau 1. Durées de transfert d'une vidéo de 30 min en 1080p HD (H.264)		
Interface	Débit	Durée
Wi-Fi 802.11n optimisée	300 Mbit/s	8 min
Wi-Fi 802.11n typique	80 Mbit/s	30 min
Wi-Fi 802.11g	54 Mbit/s	45 min
Wi-Fi 802.11b	11 Mbit/s	3 h 39 min
USB 2.0 optimisée	320 Mbit/s	7 min 30 s
USB 2.0 typique	200 Mbit/s	12 min



Le mode *SuperSpeed* d'USB 3.0 est exploité dans les PC, les périphériques de stockage ou encore les écrans vidéo. Les concepteurs de systèmes embarqués découvrent que des dispositifs USB 3.0 hautement intégrés permettent de transférer des données internes à grande vitesse. La norme USB 3.0 permet aussi d'interconnecter à peu de frais d'autres dispositifs, et ce avec un débit rapide et des consommations, empreintes, coûts et temps de développement moindres.

Déploiement du FX3STM

Il est facile d'intégrer le FX3S dans la plateforme d'une caméra en le reliant au SoC via son interface programmable GPIF (fig. 3). L'interface GPIF II conçue par Cypress est une interface parallèle entièrement configurable qui permet de relier au FX3S n'importe quel ASIC, SoC ou FPGA. Dans cette configuration, le SoC voit simplement le FX3S et ses deux cartes SD comme un périphérique de stockage USB. Les données vidéo compressées peuvent être stockées comme dans les implantations existantes des caméras sans qu'il n'y ait d'effet sur la pile logicielle.

Une telle implantation est avantageuse pour l'architecture de la caméra : d'abord c'est une façon économique et peu risquée d'ajouter un port USB SuperSpeed, ensuite elle soulage le SoC de la prise en charge de la carte SD, permettant ainsi au processeur de traiter d'autres tâches critiques.

Les opérations d'écriture (fig. 2, flèche 1) sont environ 1,8 fois plus rapides que celles des implantations actuelles lorsque l'interface GPIF II du FX3S est configurée en bus de données/adresses à 16 bits, multiplexé et cadencé à 100 MHz.

L'interface double-SD du FX3S peut être configurée en RAID 0 à l'aide d'une seconde carte mini/micro SD ; les performances sont améliorées et l'utilisateur dispose de plus d'espace de stockage. Les opérations de lecture (fig. 2, flèche 2) peuvent atteindre un débit d'environ 720 Mbit/s avec une configuration à double carte SD. Cet écart de performance devrait encore se creuser avec l'augmentation des capacités de stockage des caméras, des durées d'enregistrement, et avec l'adoption de hautes définitions d'images telles que 4K.

Le FS3S n'affecte guère l'autonomie de la batterie puisqu'il ne prélève qu'environ 97 mW de la consommation totale de la plateforme de la caméra. Avec une batterie de 3,8 V et 1160 mAh, le FX3S prélèvera 13,43 mAh pour transférer les 30 min de la vidéo ci-dessus (en supposant un rendement de conversion de 95 %), soit 1,2 % de la capacité de la batterie. L'ajout d'une seconde carte SD (~270 mW) pour augmenter le stockage et les performances affecte l'autonomie de la batterie d'environ 4,3 %.

Conclusion

Le choix de l'USB SuperSpeed s'impose naturellement lorsque l'application embarquée requiert des débits rapides pour les transferts de données. Les concepteurs peuvent attendre la prochaine génération de SoC, mais au risque d'un manque à gagner, ou se tourner vers les solutions du commerce. Des composants aussi puissants que le FX3S de Cypress peuvent réduire les temps de transfert d'images d'un facteur 4 à 10, sans guère affecter l'autonomie des batteries. ◀

(150466 – version française : Hervé Moreau)

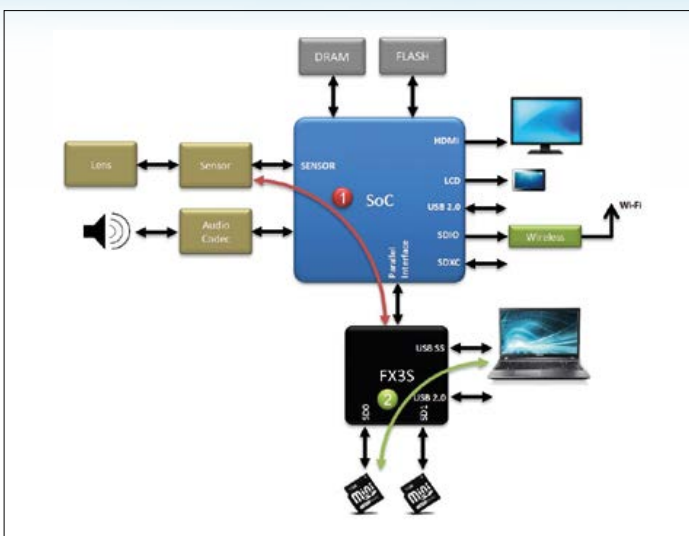


Figure 2. Schéma de principe du contrôleur FX3S à USB 3.0 de Cypress.

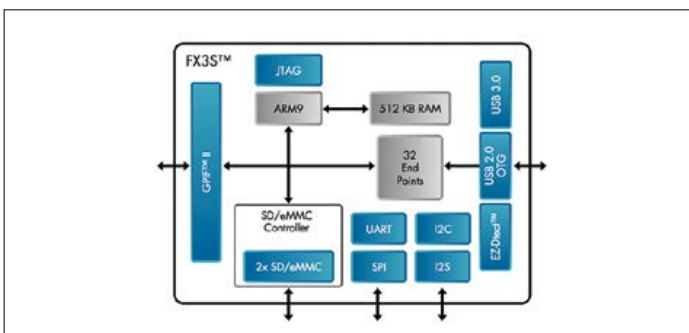


Figure 3. Architecture améliorée avec le FX3S de Cypress.

Lien

[1] Contrôleur EZ-USB® FX3STM SuperSpeed USB : www.cypress.com/?docID=45991

L'auteur

Abhishek Gupta est analyste commercial chez Cypress Semiconductor. Il est diplômé en électronique & communications de l'université Maharishi Dayanand de l'État indien Haryana. Il a été responsable de la logistique (spécialiste RoHS) chez Agilent Technologies. Contact : agni@cypress.com



Bienvenue dans Elektor Labs

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

Notre offre : La célébrité



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

Notre histoire

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

Notre équipement

Il y a l'eau courante, le gaz, l'électricité et des machines à café. Trois des plus belles pièces du château d'Elektor sont réservées au labo, mais nous avons bien du mal à ne pas envahir le reste. Nos paillasses débordent de cartes, de composants en vrac, d'accessoires mécaniques, de prototypes et d'un fatras ingérable. L'appareillage lourd est heureusement dans un local séparé.

Nos produits

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

412

propositions

65

études en cours

182

projets achevés

683

projets en tout

Nos principes

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

Nos webinaires

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

Nos experts et nos concepteurs

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

elektor labs

Sharing Electronics Projects

Home [Proposals](#) [In Progress](#) [Finished](#)

Search

Log In

Plus on est de fous d'électronique, plus on rit !

Notre site communautaire elektor-labs.com est le port I/O bidirectionnel idéal pour mettre en vedette **votre projet** et pour suivre ceux des autres. Il est partagé par des milliers d'électroniciens comme vous. Avec eux et comme eux, ouvrez votre atelier personnel sur le monde. En circulant, vos idées s'enrichiront, vos circuits s'amélioreront. Les projets les plus suivis sur www.elektor-labs.com sont mûris par l'équipe d'Elektor-Labs et certains deviennent des articles (rémunérés !) dans le magazine.

Read/Write ?

Si vous souhaitez publier sous votre nom et pour votre bénéfice un projet dans ce magazine, en quatre langues et avec la présentation soignée d'Elektor, pour être lu par des dizaines de milliers d'électroniciens dans le monde, **rejoignez la communauté avec une carte de membre GREEN ou GOLD** (www.elektor.fr/membres). Les membres peuvent publier des projets sur le site elektor-labs.com, les autres se contentent de regarder.

bienvenue dans la section **CRÉER**

Clemens Valens, Elektor Labs



Garder le meilleur pour le début

La dernière fois que j'ai pris l'avion à Charles de Gaulle (CDG) pour me rendre à Bombay, j'ai remarqué un panneau annonçant que l'aéroport était en lice pour le prix 2015 du *World's Most Improved Airport*. J'ai trouvé cela étrange. Je le fréquente plusieurs fois par an, et jamais je n'ai remarqué la moindre amélioration. Mes vols sont toujours à une heure ou plus de la station RER, et il y a encore des retards. J'ai même craint le pire des retards ce jour-là lorsque, ô surprise, au lieu de l'avion j'ai aperçu sur le tarmac un bus indiquant « Bombay » comme destination. Mais bref, ce qui m'avait vraiment surpris était le manque d'ambition de ce concours.

Imaginez que vous conceviez un produit objectivement mauvais, mais que vous décidiez de le commercialiser quand même. Un jour vous le modifiez pour qu'il remporte le prix du « Produit s'étant le plus amélioré au monde », et il le remporte. En est-il pour autant devenu un bon produit ? Non, au mieux est-il devenu moins mauvais qu'il ne l'était. Ce genre de prix ne veut rien dire. Pour concevoir un produit de qualité, il faut dès le départ aspirer au titre de « Meilleur produit du monde ». La vie serait

vraiment plus agréable si plus d'entreprises suivaient cette logique.


P.-S. – Vous ai-je dit que les *World Airport Awards 2015* avaient été remis à Paris ? Inutile de dire que CDG a remporté le prix en question.

www.worldairportawards.com

Porte 85/86



L'aéroport international Chhatrapati Shivaji de Bombay (BOM) a remporté la troisième place des *World's Most Improved Airport 2015*. Je l'ai connu avant et après sa reconstruction, et je ne peux qu'applaudir à cette marque de reconnaissance. À mes yeux il aurait même mérité de figurer sur une marche plus haute du podium, car l'endroit est vraiment moderne et agréable.

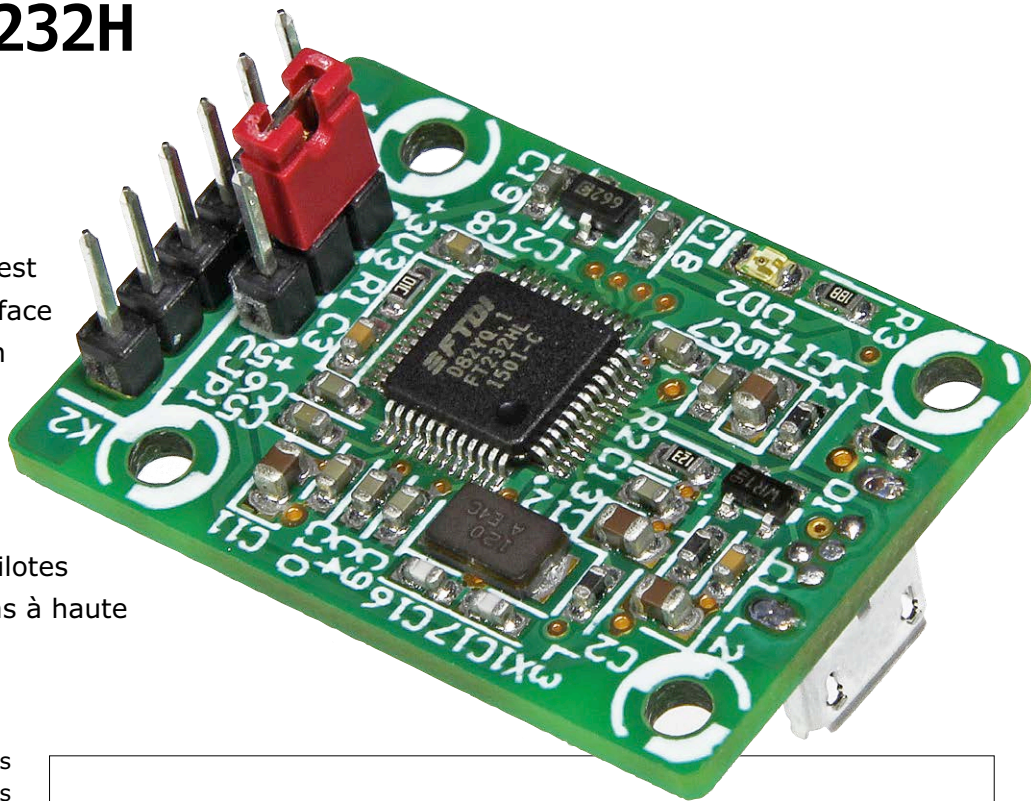
C'est en déambulant dans ses corridors parsemés d'œuvres d'art indiennes que j'ai remarqué une vaste mosaïque représentant une carte de Bombay et de ses environs. En m'approchant, j'ai découvert qu'elle était constituée de circuits imprimés, de composants et de fils. L'océan est réalisé avec des condensateurs en alu, les rivières des fils et les champs des FR4 assemblés. La photo est un peu floue à cause de la vitre blindée. C'est porte 85/86 si un jour vous passez par là. 

(150476-I – version française : Hervé Moreau)

convertisseur USB-série compact avec la puce FT232H

Ton Giesberts (labo d'Elektor) et
Thijs Beckers (rédaction des Pays-Bas)

Le port série, bien que démodé, est toujours très utilisé comme interface entre PC et électronique. Voici un module compact sur la base du circuit intégré FT232H de FTDI, qui pourra être connecté « à chaud » (*plug-and-play*) au bus USB 2.0, après installation des pilotes (*drivers*) ad hoc, pour des liaisons à haute vitesse (*high speed*).



Encore une interface série ? Eh oui, nous n'en avons jamais assez ! Nous vous avons présenté il y a peu un convertisseur USB-série multiprotocole, basé sur le FT232H de FTDI (Elektor, 04/2015, p. 70 [1]). Ce projet est similaire, mais le module est destiné à être monté sur un panneau. La prise USB se trouve donc sur le dessous du circuit imprimé. Ce montage est très pratique pour ajouter rapidement une interface USB à un projet, mais il n'est pas conçu comme une carte de liaison standard. Il était essentiel de réduire les dimensions, ce qui est réussi, avec seulement 22 x 23 mm².

Caractéristiques

- USB 2.0 haute vitesse (*High Speed*)
- Compatible (broches) avec le câble FTDI (UART) USB-TTL
- Prise USB verticale pour un montage compact
- Tension de sortie commutable entre 3,3 et 5 V
- Reconnu comme port série (COM) après installation des pilotes VCP [3]

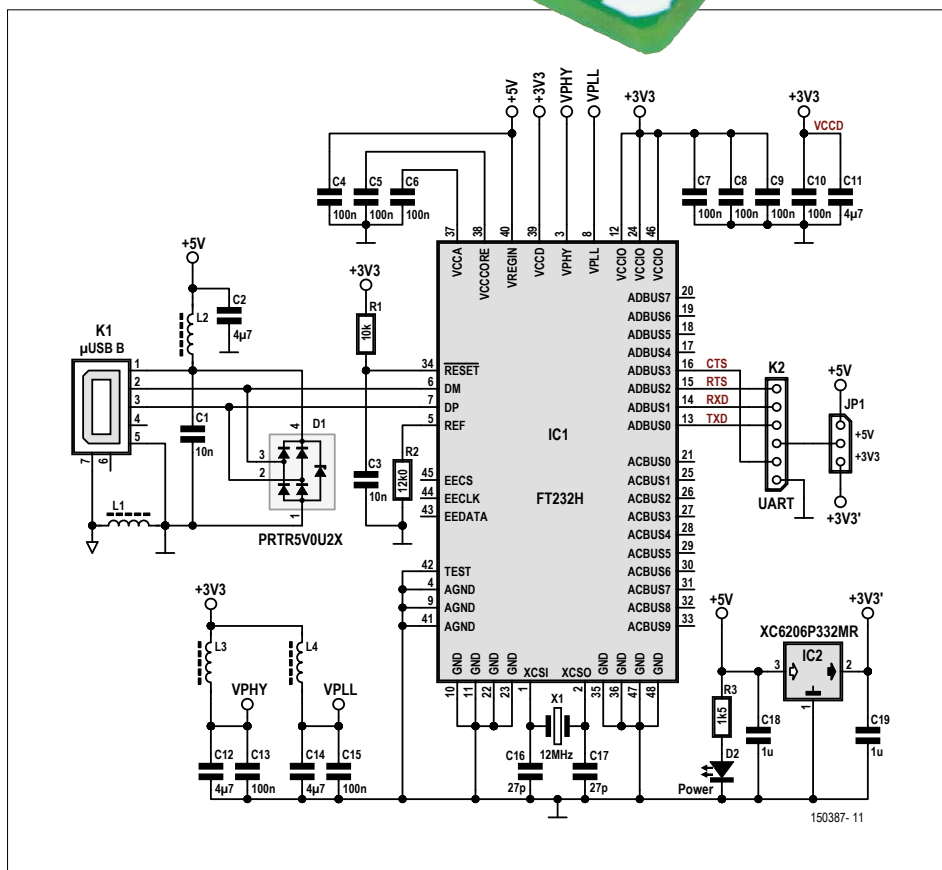


Figure 1. Le circuit est une variante « dégraissée » du convertisseur USB-série multiprotocole (Elektor, 04/2015).

MPSSE

Le *Multi-Protocol Synchronous Serial Engine* (MPSSE) est une interface configurable entre un port USB et des appareils série synchrones. Trois signaux sont nécessaires pour une liaison série synchrone : données, horloge et masse. Le signal d'horloge est généralement délivré par le « maître », et les divers appareils connectés au bus synchronisent leur fonctionnement sur ce signal. Les bus SPI, I²C et JTAG en sont des exemples.

L'UART du port série d'un microcontrôleur a en revanche un fonctionnement asynchrone. Les signaux de données et d'horloge sont combinés, et l'horloge doit être reconstituée à partir de ce signal mixte.

Le MPSSE facilite la communication avec divers appareils synchrones. Le formatage des données et la synchronisation de l'horloge peuvent être configurés de différentes manières, ce qui permet une liaison jusqu'à 30 Mbit/s. Une communication parallèle est aussi possible, par exemple en mode FT1248. Avec ce type de liaison en semi-duplex, la bande passante peut être répartie entre 1, 2, 4 ou 8 lignes de données. Les modes FIFO synchrone et asynchrone (FT245) sont également envisageables, et le *bit banging* (protocole série logiciel) n'a pas été oublié non plus.

Pleine vitesse (*full speed*) ou haute vitesse (*high speed*) ?

Bien que la norme USB 3.0 date d'il y a sept ans déjà, et que ses successeurs USB 3.1 et USB type C commencent à apparaître sur les derniers modèles d'ordinateurs portables, le débit de l'USB 2.0 est largement suffisant pour les interfaces RS-232 et RS-485, et pour communiquer avec le port série d'un microcontrôleur. Le circuit intégré FT232H répond aux spécifications *high speed* de l'USB 2.0 (480 Mbit/s), et possède – avec ses interfaces de types I²C, SPI et parallèle – une longueur d'avance sur le FT232R, limité au mode *full speed* à 12 Mbit/s. Les autres fonctions de la version « H » sont implantées dans le moteur de liaison série synchrone multiprotocole (*Multi-Protocol Synchronous Serial Engine* – MPSSE), voir l'encadré.

Nous n'utilisons pas ces fonctions additionnelles ici, mais si vous voulez en tirer parti, voyez l'article du numéro d'avril 2015. Il est cependant logique d'utiliser la version « H » du circuit intégré plutôt que la « R », elle est (actuellement) moins chère.

Liste des composants

Résistances : (0,1 W, 1 %, CMS 0603)

R1 = 10 kΩ
R2 = 12 kΩ
R3 = 1,5 kΩ

Condensateurs : (CMS 0603, sauf mention contraire)

C1, C3 = 10 nF, 50 V, 10 %, X7R
C2, C11, C12, C14 = 4,7 µF, 10 V, 10 %, X7R, CMS 0805
C4-C10, C13, C15 = 100 nF, 50 V, 10 %, X7R
C16, C17 = 27 pF, 50 V, 5 %, COG/NPO
C18, C19 = 1 µF, 6,3 V, 20 %, X5R

Bobines :

L1-L4 = 600 Ω @ 100 MHz, 25 %, 0,15 Ω, 1,3 A, CMS 0603

Semi-conducteurs :

D1 = PRTR5V0U2X (CMS SOT-143B)
D2 = LED verte (CMS 0805)
IC1 = FT232HL (CMS LQFP-48)
IC2 = XC6206P332MR (CMS SOT-23)

Divers :

K1 = prise micro-USB 2.0, type B, pour montage vertical sur circuit imprimé
K2 = connecteur SIL à 6 broches, pas de 2,54 mm
JP1 = embase à 3 broches avec cavalier, pas de 2,54 mm
X1 = quartz 12 MHz, 18 pF, CMS 5 x 3.2 mm²
Circuit imprimé 150387-1 (www.elektor.fr)

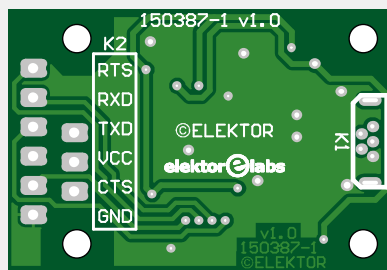
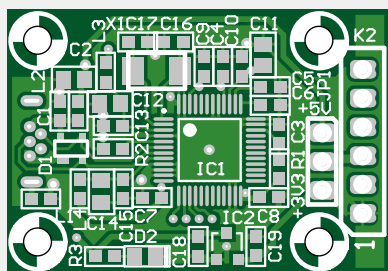
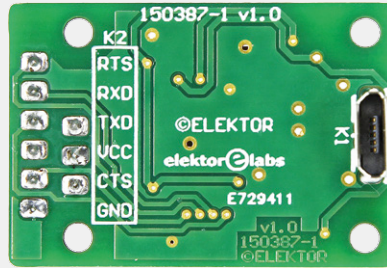
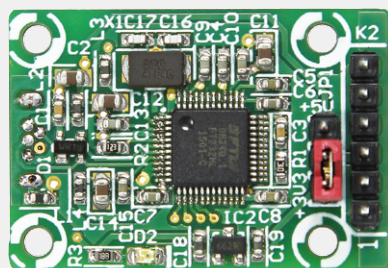


Figure 2. Presque chaque mm² est utilisé, pour miniaturiser le circuit imprimé.



► La haute vitesse !
Quoi d'autre ?

Trouvez la différence

Il est évident que le schéma de la **figure 1** est très proche de « l'original » [1]. Le FT232H est entouré d'un régulateur de tension 3,3 V (IC2), et des divers condensateurs de découplage et déparasitage nécessaires. Une diode de protection spéciale (D1) protège les entrées du FT232H contre les décharges électrostatiques jusqu'à 8 kV. La broche 34 (RESET) est à 3,3 V via R1 et C3, tel que préconisé par la fiche de caractéristiques. De même, la broche 5 (REF) est mise à la masse via R2 (12 kΩ, 1 %). Le quartz de 12 MHz (±30 ppm) est aussi un composant obligatoire, avec ses deux condensateurs d'adaptation C16 et C17. L1, L2 et C1 protègent contre le bruit haute-fréquence présent sur la ligne d'alimentation et le blindage du connecteur. La LED D2 (verte) indiquera que le circuit est bien alimenté.

Tableau 1. Connecteur K2 (câble FTDI).

Broche	Fonction	Couleur
1	GND	Noir
2	CTS (AD3)	Brun
3	VCC	Rouge
4	TX (AD0)	Orange
5	RX (AD1)	Jaune
6	RTS (AD2)	Vert

K1 est une prise micro-USB, montée sous le circuit imprimé. Les signaux de l'UART (émetteur-récepteur asynchrone universel) atteignent le FT232H via K2. Le brochage est compatible avec un câble FTDI [2], le **tableau 1** décrit la fonction des broches. JP1 permet de sélectionner la tension de sortie V_{CC} sur la prise : 5 ou 3,3 V (voir **tableau 2**). Comme dans le circuit « original », le régulateur IC2 est relié directement à la ligne 3,3 V, afin de

Tableau 2. Cavalier (jumper) JP1.

Contact	Fonction
1-2	Broche 3 de K2 = 5 V
2-3	Broche 3 de K2 = 3,3

soulager l'alimentation 3,3 V du FT232H. La mémoire EEPROM qui offre la possibilité de configurer le convertisseur USB-série multiprotocole a été « liquidée ». On ne peut donc pas configurer le FT232H via *FT Prog* pour divers modes de fonctionnement ; seul le mode par défaut – comme UART – est disponible.

En pratique

Le petit circuit imprimé (**figure 2**) peut être fixé par quatre boulons M2. La prise USB (modèle vertical) est montée sous le circuit imprimé ; les broches resteront donc accessibles même si l'on utilise un panneau (avant ou arrière) relativement épais.

Lors du raccordement à un PC, le FT232H est reconnu comme UART, si les pilotes de port virtuel de communication (*Virtual COM Port – VCP* [3]) ont été préalablement installés. La **figure 3** montre cette détection sous Windows 7. Si des difficultés se présentent, vous trouverez de nombreuses aides sur l'internet, par exemple chez Adafruit [4]. ◀

(150387 – version française : Jean-Louis Mehren)

Cypress CY7C65211

FTDI n'est pas le seul fabricant de circuits intégrés pour interfaces série. Cypress a développé le CY7C65211 : un *USB-Serial Single-Channel (UART/I²C/SPI) Bridge with CapSense® and BCD*. Ce circuit présente de nombreux avantages : EEPROM intégrée, donc pas besoin d'un circuit additionnel ; basculement du mode de fonctionnement via une API (*Application Programming Interface*), alors que chez FTDI il faut passer par un logiciel de configuration ; tension du port série entre 2,2 et 5 V, pour n'en citer que quelques-uns.

Nous sommes cependant restés fidèles au FT232H pour ses caractéristiques *USB 2.0 High Speed*. Tout simplement parce que c'est la haute vitesse. Et aussi parce que le circuit est configuré en UART par défaut, et qu'il peut donc être mis en œuvre directement.

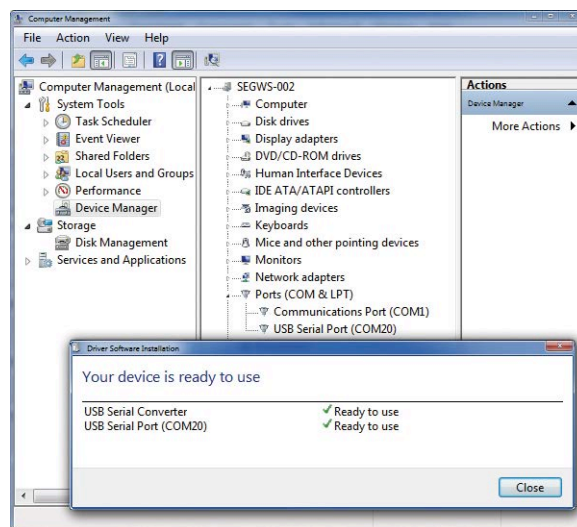
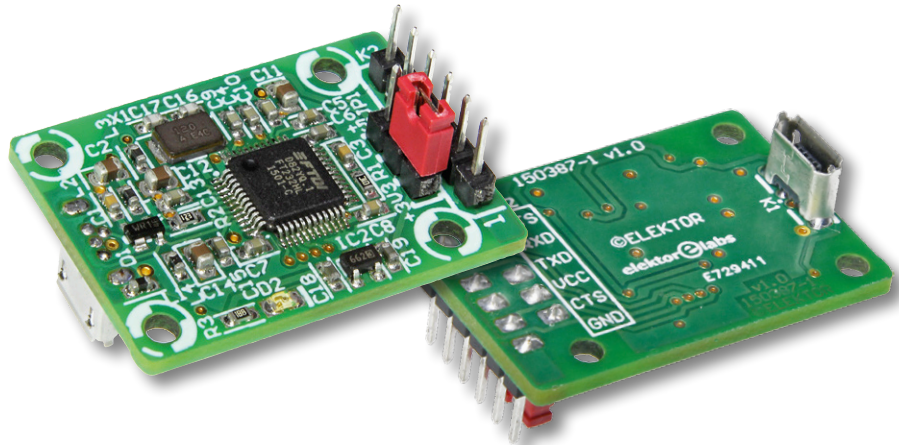


Figure 3. Après installation des pilotes VCP, le FT232H est directement reconnu comme port COM.

Liens

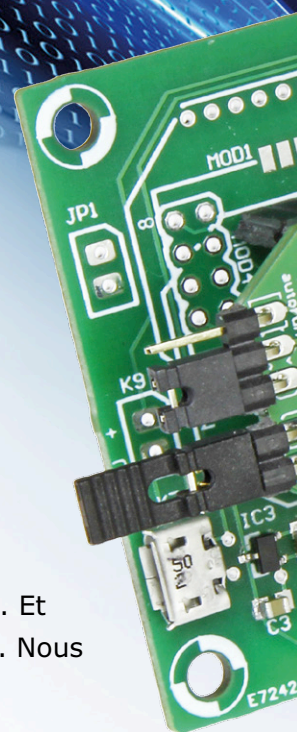
- [1] Convertisseur USB-série multiprotocole, Elektor, avril 2015 : www.elektormagazine.fr/130542
- [2] Câbles FTDI : 080213-71 (version 5 V) et 080213-72 (version 3,3 V), disponibles sur www.elektor.fr.
- [3] Pilotes (drivers) à télécharger sur le site de FTDI : www.ftdichip.com.
- [4] Aide à l'installation sur Adafruit : <https://learn.adafruit.com/adafruit-ft232h-breakout>.

carte d'E/S pour Android (3)

l'enregistreur de données

Elbert Jan van Veldhuizen (Pays-Bas)

Cette troisième partie doit vous permettre de transformer la carte d'E/S pour Android en un enregistreur de données aux multiples possibilités avec de nombreuses fonctions. Les données y sont stockées dans une grande mémoire flash de 32 Ko embarquée sur la carte. Et pour la piloter depuis votre téléphone portable en toute tranquillité, il y a une appli simple. Nous accorderons en plus une attention particulière au chargeur d'amorçage et au micrologiciel.



La carte d'E/S pour Android est équipée d'une mémoire flash sérielle de 32 Ko, disponible pour l'enregistrement autonome de données de mesure. Supposons que vous en stockiez toutes les minutes depuis deux canaux analogiques, vous aurez ainsi de la place pour une semaine facilement.

Un autre mode permet de mémoriser des mesures statistiques sur une plus longue durée. Par exemple, on prend une mesure toutes les 0,9 s ; puis toutes les heures, le système calcule la moyenne de 4 096 échantillons, l'écart type, le minimum et le maximum et stocke les résultats. Surveillez la température, par exemple, tout en conservant assez d'informations pour observer des événements importants. Dans cette configuration, vous pouvez retrouver toutes les données d'un canal sur une période de neuf mois environ. On peut enregistrer les données de maximum huit entrées en même temps, par exemple quatre du CA/N, le compteur, un capteur capacitif (CPS) et deux numériques. La première partie a décrit toutes les configurations possibles des entrées. On peut encore réduire le bruit du CA/N en multipliant le nombre d'échantillons ou utiliser la tension de référence. Il est possible de prévoir l'arrêt du micrologiciel quand

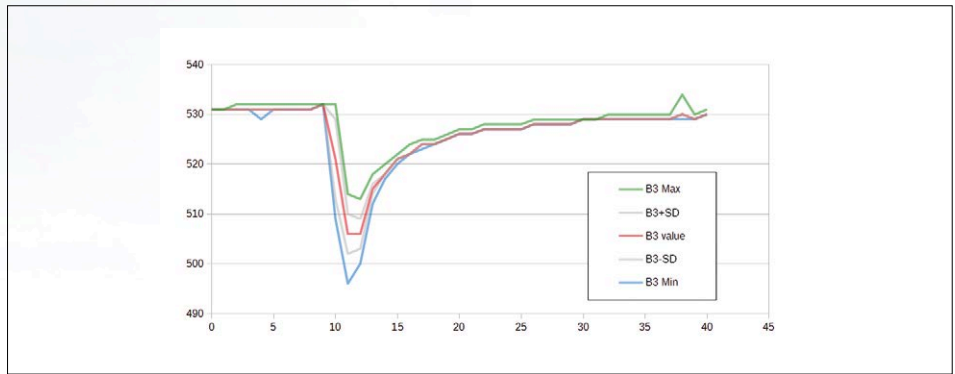


Figure 1. Les différentes valeurs statistiques des mesures.

la mémoire est pleine ou y réécrire en boucle ou encore reprendre l'écriture où elle avait été interrompue par une panne de courant, par exemple.

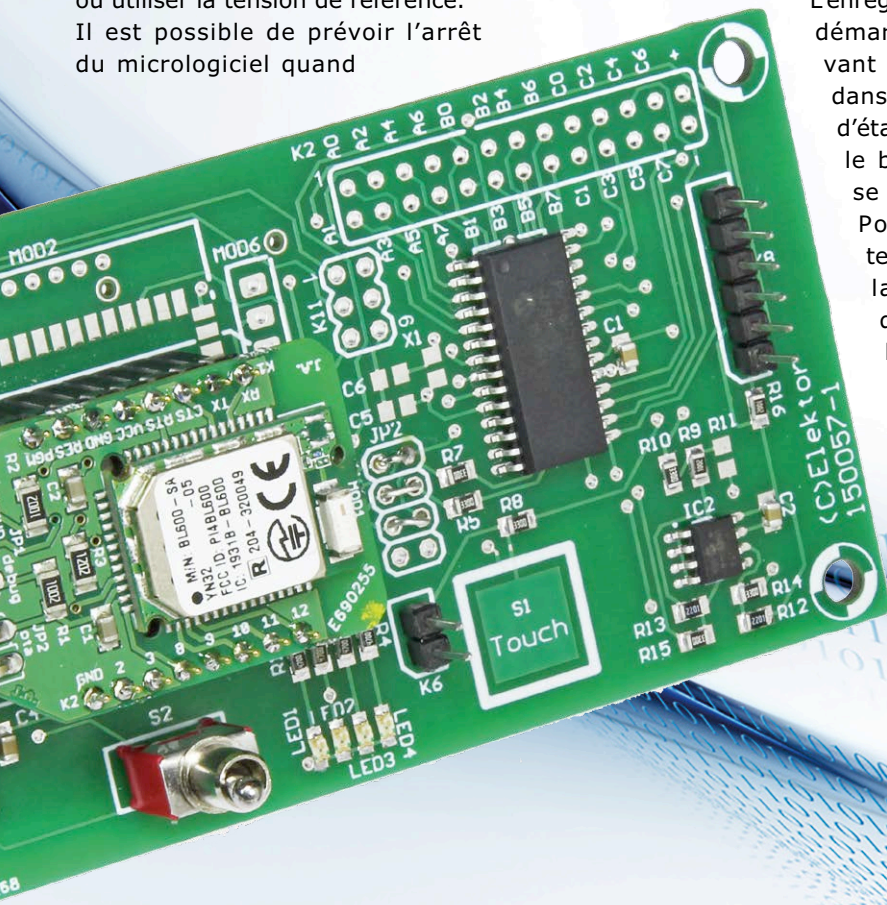
Les registres d'état S0, S1 et S3 assurent le service de l'enregistreur de données. Pour enregistrer une entrée, il faut que le bit 7 de son registre d'état 1 soit mis au niveau haut. Comme indiqué précédemment, d'autres réglages sont aussi possibles. Pour consigner les valeurs de l'entrée B2 converties en numérique avec une tension de référence de 2,048 V, vous devrez d'abord donner la commande `s b2 5#s b21 129`.

L'enregistrement démarre en écrivant la valeur 1 dans le registre d'état S3. Alors, le bit 0 de S0 se lève aussi. Pour l'arrêter, on inscrit la valeur 0 dans S3. Le bit 0 de S0 est alors également abaissé. Pour

reprendre l'enregistrement où il s'était arrêté, il faut inscrire la valeur 2 ou 3 dans S3. Avec un 2, il continuera dans le bloc de 64 octets (cf. plus loin) où il en était, tandis qu'après une initialisation du microcontrôleur, il repart du début. Avec un 3, l'enregistreur entame un bloc vierge de 64 octets.

Avec les bits 5 à 7, vous réglez l'intervalle entre les écritures dans la mémoire flash sérielle. Dans l'ordre croissant, vous fixez ainsi un délai de 10 ms, 1 s, 15 s, 1 min, 15 min, 1 h ou 1 jour. Les bits 3 et 4 définissent le nombre de mesures à effectuer au préalable. Elles vont ainsi de 1 à 64, 4 096 ou 4 194 304. Lever le bit 7 du registre d'état S1 vous permet de voir un clignotement de la LED sur B0 lors de chaque mesure.

L'enregistreur peut prendre la moyenne des mesures (leur somme divisée par le nombre de mesures) ou le cumul de leurs valeurs. On peut également écrire les valeurs minimales et maximales (cf. **figure 1**) et même l'écart type calculé. C'est ce qu'on règle pour toutes les entrées au moyen des cinq premiers bits du registre d'état 1 de S1. Il y a toutefois une configuration spéciale dans le cas où la valeur d'entrée ne comporte qu'un seul bit, comme pour une entrée numérique ou un CPS. La moyenne, comme nombre entier, ne serait pas



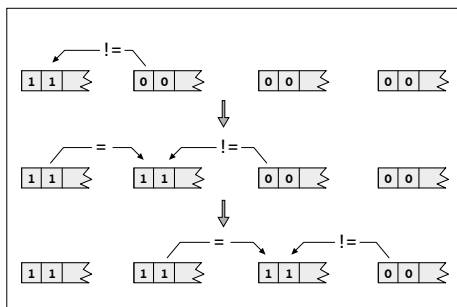


Figure 2. Les deux premiers bits dans le *runID*.

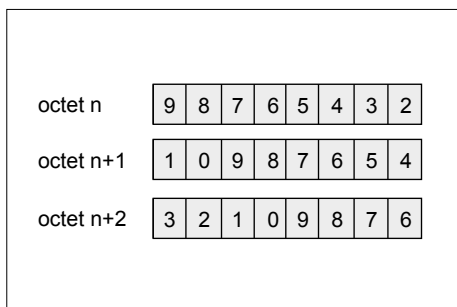


Figure 3. L'enchaînement des bits en octets, ici les bits 9 à 0 de mesures du CA/N.

significative. Si les bits 0 et 1 de S1 sont tous deux bas, pour des valeurs sur 1 bit, on écrit la somme des mesures, mais la moyenne pour des valeurs sur plusieurs bits. Quand on s'attend à des valeurs distribuées normalement, il suffit de sauvegarder la moyenne et l'écart type. À supposer que vous mesuriez la température dans un réfrigérateur en vue de découvrir des pics de température, vous enregistrerez sans doute la moyenne et le maximum (le minimum de résistance avec une CTN). Vous aurez ainsi une indication de la durée du pic, plus elle a été longue, plus la moyenne se situe près du maximum.

N'importe quelle géométrie ne donne pas nécessairement de bons résultats, entre autres à cause du nombre de bits des variables. La somme des mesures a une longueur maximale de 32 bits (4 194 304) et les valeurs de maximum et de minimum, 16 bits (65 535). Si, d'un CA/N, vous ne retenez que la moyenne et la somme, vous pourrez faire exécuter 4 194 304 mesures. Comme l'écart type se calcule en prenant le carré de la somme des mesures, il doit s'écrire dans un registre à 32 bits. S'il faut mémoriser l'écart type d'un CA/N à 10 bits, il n'y aura plus que 64 mesures possibles tout au plus. Il convient alors de transmettre

la moyenne (registre d'état 1 du CA/N).

La chronométrie a aussi son importance. L'inscription des données de 64 octets dure 5 ms. Le calcul de l'écart type prend 2,5 ms par canal et la mesure elle-même une fraction de milliseconde. Envisageons une périodicité de 10 ms pour quatre canaux, avec 64 mesures et l'écart type, le système ne pourra même pas suivre la moitié de la vitesse demandée et causera beaucoup de gigue. D'ailleurs, nous verrons plus loin qu'il y a une possibilité de vérifier si une instabilité s'est produite. En prévision d'un redémarrage initié par le bit 2 du registre d'état S0, il faut avoir sauvegardé ce réglage et la configuration des cellules dans l'EEPROM du microcontrôleur par la commande 2 adressée au registre d'état Z3. Sans quoi, le microcontrôleur les aura oubliés lors du lancement. Il est aussi nécessaire que le bit 2 du registre d'état Z soit levé pour qu'il aille vraiment lire ces valeurs.

Comme l'appli pour l'enregistreur lit les valeurs brutes dans la mémoire flash sérielle, il est nécessaire de connaître la structure sous laquelle les données y ont été stockées.

Mémoire flash sérielle

La mémoire choisie est une 24FC256 dotée du bus I²C. Le microcontrôleur la lit par l'intermédiaire de deux lignes. Le protocole I²C utilise les cellules C3 et C4 mises à collecteur ouvert, chaque ligne est polarisée par résistance sur Vcc et peut être tirée à la masse. Une méthode qui empêche tout court-circuit si deux puces tentent de transmettre des données en même temps. Une des lignes transporte l'horloge, l'autre, les données. La valeur sur la ligne de données est saisie lors de la transition de 0 à 1 du signal de l'horloge. Le débit binaire maximum est de 1 Mb/s.

Avec la mémoire flash 24FC256, il ne faut pas perdre de vue qu'elle s'utilise quand on s'en sert. Pour y écrire un seul octet, c'est toute une rangée, les 64 octets qui sont d'abord enregistrés localement. Ensuite, toute cette rangée est effacée, puis les données sont réécrites. Écrire les 64 octets un à un demande alors 64 effacements de la même rangée. Or, une mémoire flash n'a qu'une durée de vie garantie d'un million d'effacements. En procédant de la sorte, la mémoire ne pourra servir que 15 000 fois environ.

Il est préférable que le microcontrôleur utilise une mémoire tampon comme intermédiaire pour écrire les 64 octets en une seule fois dans la mémoire flash, de quoi prolonger sa fiabilité d'un facteur 64. Dans cette optique, on utilise un bloc de 64 octets de la RAM du microcontrôleur en tampon. L'inconvénient survient en cas de panne de courant, parce qu'on ne sait plus où les données ont été écrites dans la mémoire flash. Le truc que j'ai utilisé pour retrouver les données, c'est de les jalonner, d'écrire au début de chaque bloc de 64 octets une sorte d'identifiant (*runID*). Le protocole consiste à donner au premier bit du bloc de 64 octets la même valeur que le deuxième du bloc précédent et au deuxième bit la valeur inverse de celle du premier bit du bloc suivant. La **figure 2** est sans doute plus parlante à ce sujet. Le résultat obtenu, c'est que de toutes les données concernées, le bit 6 du bloc de 64 octets et le bit 7 du suivant, sont égales alors que les anciennes données divergent. On peut ainsi retrouver où l'enregistreur en est resté.

En outre, on utilise les bits 3, 4 et 5 comme drapeaux. Le bit 3 indique que la mémoire a été toute remplie et qu'on y écrit actuellement en boucle au moins pour la deuxième fois. Avec le bit 4, on sait que ce bloc a été réutilisé depuis le début, par exemple à la suite d'une panne de courant, et que donc une discontinuité dans les données est possible. Le bit 5 signale une irrégularité dans les intervalles de mesure du fait que le microcontrôleur a été pris de vitesse. Il importe de le savoir parce que les données ne sont pas horodatées. On suppose que les mesures sont normalement enregistrées à la fréquence prescrite. Mais si le bit 5 le démentit, c'est qu'il y a eu des ratés à cause de la gigue. Le bit 5 n'est remis à zéro que lors d'une nouvelle session d'enregistrement.

Les données sont chaînées au fur et à mesure. Donc s'il y a deux valeurs de 10 bits à inscrire, le premier octet prend les 8 bits les plus hauts (9 à 2) et le reste, les deux bits de poids faible entament le deuxième octet par le haut. Suivent alors les 6 bits de poids fort (9 à 4) de la seconde valeur pour compléter l'octet et les 4 bits qui restent se placent en haut du troisième octet et ainsi de suite. C'est ce qu'illustre la **figure 3**. On ne peut pas écrire plus de 63 octets de

données par mesure. Il est possible de paramétrer la carte d'E/S pour Android de manière à stocker plus de 63 octets pour une mesure, mais alors elle réagit en levant le bit 1 du registre d'état (demander *r c4* pour le lire) et arrête l'enregistrement.

L'appli doit savoir avec précision le nombre de bits dont chaque canal dispose pour s'y retrouver dans l'enchevêtrement des données brutes de la mémoire flash sérielle. C'est le bloc d'en-tête de 64 octets qui détient toutes ces informations. On y apprend quelle cellule il faut lire, quelles données y sont inscrites et sur combien de bits. Toutes les précisions sur le format sont reprises dans le **tableau 1**.

Pour connaître ce qu'il y a dans la mémoire flash, on envoie au registre C3 une commande *Read* avec la bonne adresse, comme *r c3 128*. Le contrôleur répond sur 32 octets, exactement la dimension du tampon d'écriture. S'il y en a davantage, le contrôleur s'arrête, l'enregistrement des données aussi, jusqu'à ce que ces données aient été transférées par la liaison à 9600 bauds. L'appli doit attendre que les 32 octets soient partis avant de réclamer de nouveaux mots de 32 octets, de manière à éviter la gigue et les retards dans les mesures.

Avec les adresses 32 768 et 32 800, on lit dans le tampon actuel au lieu de la mémoire flash sérielle. L'appli peut tout de même lire la mesure en cours quand la périodicité des mesures est longue : avec une mesure à l'heure, par exemple, le tampon pourrait ne devoir se vider qu'une fois par jour.

On peut obtenir le statut de la mesure en interrogeant le registre C4. Il donne en hexadécimal l'état, la position dans les blocs de 64 octets de la mémoire flash, la position en octets et en bits dans le tampon, le nombre de mesures qui doivent encore être effectuées avant la prochaine écriture dans le tampon et le nombre de millisecondes avant la prochaine mesure. Les bits les plus intéressants dans l'état sont le bit 0 qui indique si l'enregistrement est en cours ou pas, le bit 2 qui signale la gigue et le bit 4 pour savoir si la mémoire a déjà été parcourue jusqu'au bout.

Appli de l'enregistreur de données

Pour la préparation de l'enregistreur de données, l'appli Android (**figure 4**)

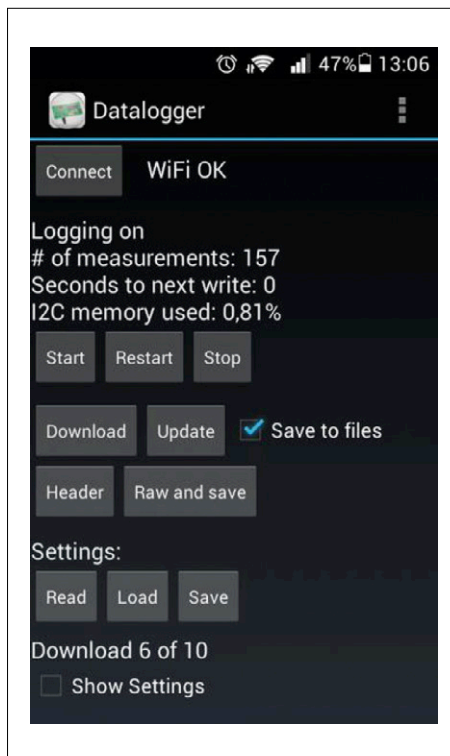


Figure 4. L'appli développée pour l'enregistreur de données.

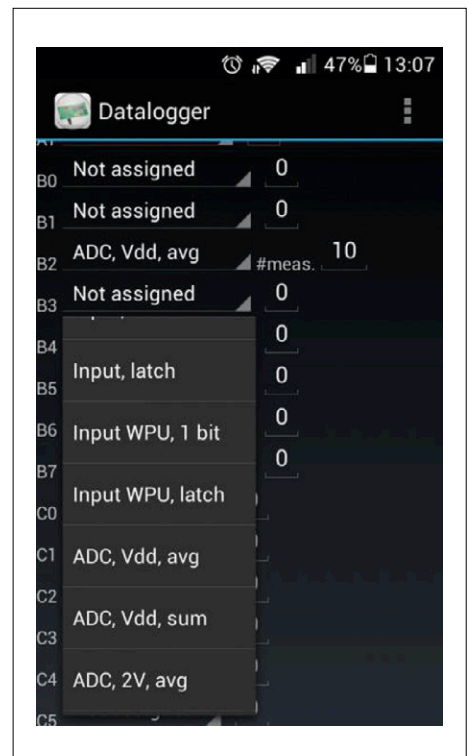


Figure 5. Les options possibles pour chaque cellule, répertoriées dans un menu déroulant.

donne accès à tous ces paramètres : la fréquence, le nombre de mesures, le choix de les effectuer en boucle et de pouvoir relancer tel quel l'enregistrement au (prochain) démarrage. On peut en outre décider de ce qu'il faut sauvegarder : la somme ou la moyenne, le maximum, le minimum et l'écart type. Mais aussi paramétrer les différentes cellules à l'aide d'un menu déroulant (un *spinner* selon Android) qui présente (**figure 5**) les options possibles pour chacune d'elles. Pour certaines, il y a un second paramètre

à introduire : le nombre de mesures pour le CA/N et le seuil pour le CPS.

L'appli sert à démarrer et arrêter l'enregistreur, mais aussi à le relancer, à condition de ne rien changer aux paramètres ni aux fonctions des cellules. C'est pour cela que lors d'un redémarrage, même si l'appli a changé des paramètres, les nouveaux peuvent ne pas être pris en compte.

L'appli fournit encore l'état de l'enregistreur, pour savoir s'il est en marche, l'état

Tableau 1. Contenu de l'entête

octet	contenu (octet de poids faible = octet bas)
0	runID
1	registre d'état S0
2	registre d'état S1
3	nombre de bits supplémentaires à la somme
4,5	total des bits par mesure (tous canaux)
6,7,8,9	nombre de mesures par écriture
10,11,12,13	nombre de millisecondes entre les mesures
16	numéro de broche du premier canal (255 = aucune)
17	nombre de bits du premier canal
...	
30	numéro de broche du dernier canal (255 = aucune)
31	nombre de bits du dernier canal



Figure 6. Le rendu graphique des données mesurées.

d'avancement de l'écriture en mémoire et combien de temps à attendre avant la prochaine mesure et sa conservation. Il y a différentes manières de lire les données, un téléchargement complet ou seulement une mise à jour depuis la lecture précédente. Ces données se présentent sous forme graphique et peuvent aussi être transcrites en fichier csv, lisible par Excel de Microsoft ou Calc de LibreOffice. Un fichier d'information est aussi disponible avec les données principales des mesures, par exemple les cellules interrogées et les paramètres utilisés. Ces fichiers sont transcrits dans le dossier « datalogger » de l'appareil Android. On peut ne prendre que le bloc d'en-tête

pour afficher les informations utiles, mais aussi télécharger et sauvegarder les données brutes de la mémoire flash sérielle.

La structure de l'appli est similaire à celle de l'appli Android pour contrôleur Etch décrite dans la partie 2. Dans « onCreate », on lance les éléments graphiques, la boucle d'arrière-plan et le *handler* de messages. Les *spinners* pour les fonctions des cellules ne sont pas définis en fichier xml, mais faits d'une boucle en Java. En outre, toute la structure graphique est installée dans une *scrollView* qui automatiquement, sans programmation supplémentaire, organise la fonction de défilement. La communication avec la

carte d'E/S pour Android se déroule avec le même code que celui utilisé dans l'appli de la partie 2.

Cette appli-ci rencontre aussi la complexité d'envoi des commandes à la carte d'E/S et la réception des commandes. Ce qui signifie qu'une partie des fonctions se trouve dans le *thread*, piloté par des variables qui servent de drapeaux (éventuellement levés par un détecteur d'événement *clicklistener*), avec le retour effectué par un *handler* de message pour transférer les éléments graphiques. Ce qui fait que les possibilités de chacun sont réparties en plusieurs passages du code, au détriment de la clarté d'ensemble, mais c'est inhérent à la programmation en Java.

Cette appli contient aussi un tampon pour créer des retards dans la transmission des commandes. C'est ainsi que les fonctions des différentes cellules sont soumises à un retard de 10 ms pour empêcher que les tampons de la carte d'E/S soient saturés et que des commandes se perdent. Ces fonctions pourraient aussi faire l'objet d'applis à rédiger soi-même. D'autre part, lors de la lecture des données dans la mémoire flash, on ne lit les blocs qu'au rythme d'une fois toutes les 120 ms.

Quand vous programmez une appli, pensez toujours à aller voir s'il n'y en a pas déjà une adéquate dans les bibliothèques. Pour la construction du graphique, j'ai fouiné dans la bibliothèque *open source* AndroidPlot. Il y a d'autres bibliothèques comparables qui font la même chose. Pour éditer soi-même la source en Eclipse, il faut télécharger le fichier jar *android-plot-core-x.x.x.jar* [1] et le glisser dans le répertoire lib/. AndroidPlot assure un rendu impeccable des graphiques, avec légendes et axes, ainsi que la faculté de zoomer (figure 6). Davantage d'information sur AndroidPlot en suivant [2].

Chargeur d'amorçage

Pour ce projet, j'ai rédigé un chargeur d'amorçage du microcontrôleur et je l'ai logé dans une partie sécurisée de la mémoire pour empêcher le micrologiciel de l'atteindre. Ce chargeur d'amorçage simplifie le remplacement du micrologiciel par une nouvelle version à expérimenter. La partie protégée du microcontrôleur ne compte que 512 instructions. J'ai dû mettre en œuvre différentes astuces pour faire tenir le chargeur dans cet espace

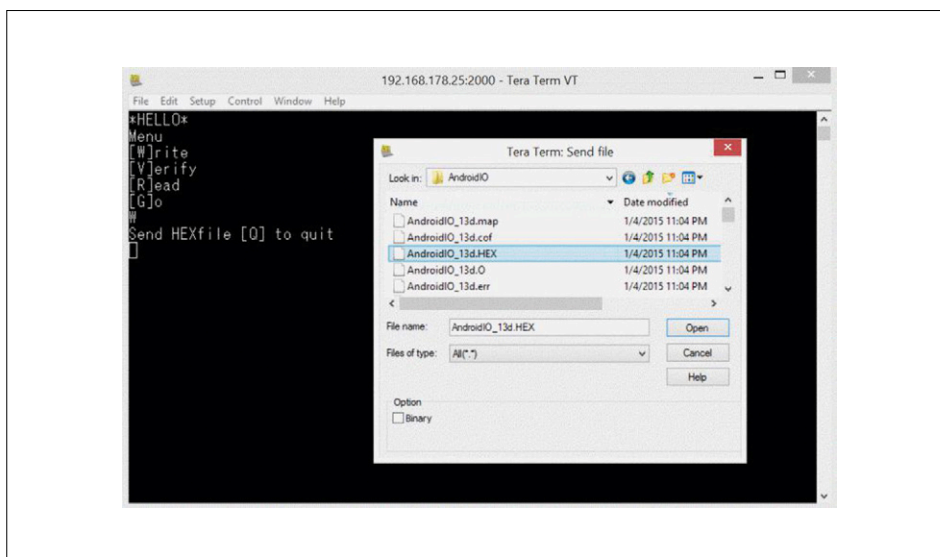


Figure 7. Chargement d'un fichier dans la carte d'E/S à partir d'un programme de terminal sur PC.

tout en préservant la convivialité à l'utilisation. Cela consistait à n'occuper que des mots de 14 bits pour deux caractères de texte et à grouper les variables du code pour minimiser autant que possible le nombre de changements de banque.

Comment activer le chargeur d'amorçage ? En connectant à la masse la broche B2 au cours du lancement du microcontrôleur, au moyen d'un bouton-poussoir ou d'un cavalier sur K6. La LED3 sur RB0 s'allume alors. Un programme de terminal, en liaison avec la carte par Bluetooth, Wi-Fi ou USB, donne à voir le menu. Différentes options (cf. **figure 7**) sont disponibles. Pour charger un programme dans le microcontrôleur, poussez sur W pour *write*. Le chargeur d'amorçage vous demande alors le fichier hexadécimal du programme. Avec TeraTerm Pro, par exemple, on choisit le fichier hexadécimal par « file/send file... » et on le transfère. Mais à 9 600 bauds, selon la longueur du fichier, cela peut durer jusqu'à une minute et demi. Depuis toutes les options du menu, vous retournez dans le menu principal en tapant Q.

Avec le V de *verify*, vous contrôlez de la même façon si le fichier hexadécimal est correctement programmé. Le R de *read* vous procure un listing de la mémoire de programme depuis l'adresse 0x0200 jusqu'à 0x3FFF. S'il y a plus de quatre adresses vides (valeur 0x3FFF), le listing saute à la prochaine place occupée. Pour finir et par souci de clarté, la dernière adresse est toujours révélée.

On lance le micrologiciel par R (*run*). On peut retourner au chargeur d'amorçage en inscrivant dans le micrologiciel 20 ou 21 vers z3. Dans le premier cas, le chargeur d'amorçage ne démarrera qu'après une action sur le bouton. Dans le second cas, le programme de chargement démarre toujours. On peut aussi

sauter vers le chargeur en envoyant la capitale Q lorsque le mode de débogage est en service. On peut aussi sauter vers 0x0000 ou 0x0019 dans le micrologiciel lui-même. N'oubliez pas dans tous les cas de désactiver l'interruption et de mettre PCLATCH et BSR à 0.

Adaptation de micrologiciel

Il y a plusieurs raisons de vouloir modifier le micrologiciel et vous pouvez le faire vous-même. D'abord, vous pouvez y ajouter les paramètres requis au démarrage pour initialiser automatiquement les ports, de manière à ne pas devoir les introduire chaque fois dans la carte d'E/S pour Android et les enregistrer en EEPROM. Un supplément de code est aussi requis pour des puces I²C ou SPI spécifiques ou des formats particuliers que le logiciel doit comprendre, comme ceux de la télécommande IR ou des modules à 433 MHz.

Toute adaptation du micrologiciel suppose la connaissance du langage assembleur à 14 bits de Microchip. Vous pouvez alors vous servir de l'IDE MPLAB ou MPLAB X, qui permettent aussi la compilation et les essais. Les fichiers sources nécessaires, AndroidIO.asm et ppmacros.asm sont disponibles au téléchargement sur le site d'Elektor [1].

Le plan du micrologiciel est exposé à la **figure 8**. Deux boucles se complètent. On appelle la routine d'interruption pour exécuter les tâches du temps réel, la production de signaux MLI, la mesure du CPS, la lecture du compteur et le suivi des variables à référence temporelle, comme les drapeaux pour le CA/N, l'horloge, le temporisateur chien de garde et la chronométrie elle-même. La routine d'interruption ne s'occupe pas du texte pour le port sériel, il est toujours géré par la boucle principale pour éviter le blocage de la routine de service d'interruption quand le tampon est plein. De la boucle

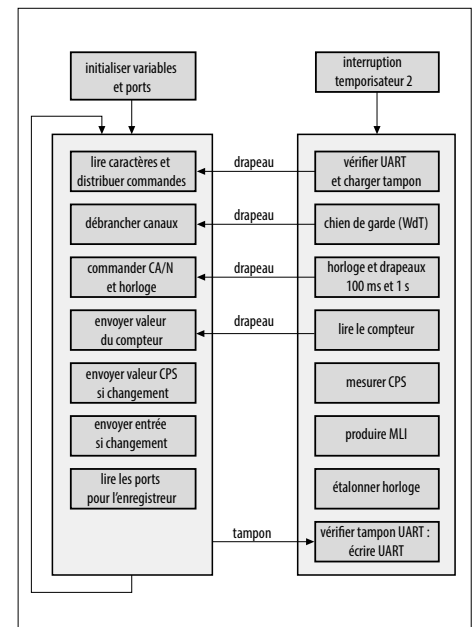


Figure 8. Le plan du micrologiciel.

principale, on écrit dans le tampon et la routine d'interruption ne transmet qu'un caractère à la fois vers le port sériel à destination de l'appareil Android quand l'UART est disponible. La boucle principale agit en distributeur de commandes et, si on le lui demande, inscrit les modifications intervenues sur le CPS, le CA/N et l'horloge. C'est aussi la boucle principale qui dirige l'enregistreur de données. Vous trouverez dans le code source lui-même des conseils complémentaires pour l'adaptation du micrologiciel, la manière d'ajouter des paramètres ordinaires ainsi que sur ce qui réclame spécialement votre attention.

D'autres systèmes

Dans cet article, on s'est principalement focalisé sur la commande de la carte d'E/S par un appareil Android, mais elle marche très bien avec un PC ou un portable sous Windows ou Linux et avec d'autres systèmes comme Raspberry Pi.

Bogues dans le PIC16F1938

Dans les deux premières versions du PIC16F1938 (révisions A et B), il y a une erreur qui empêche une bonne lecture du CA/N à des fréquences d'horloge supérieures à 16 MHz, ce qui peut entraîner l'arrêt du contrôleur, cf. [4]. On peut lire la lettre de la version dans le registre d'état Z par la commande 11.

Mais il y a une solution pour y remédier qui permet d'utiliser le CA/N de ces versions. Il faut pour cela désactiver

les interruptions pendant la lecture du CA/N et ralentir temporairement la fréquence d'horloge. Ceci peut provoquer un peu de gigue dans le signal MLI et le compteur.

C'est pourquoi il y a deux versions du micrologiciel : la normale pour la révision C (et peut-être supérieure dans le futur) du PIC16F1938 et une autre avec le remède pour les premières versions, avec « _AB » dans le nom.

Listage 1. Script de mesure périodique de température avec dessin du graphique

```
#!/bin/bash
IOcmmd () { sleep 5; echo; echo s b3 3#; sleep 1; echo s b32 10#; sleep 1; echo r b3#;sleep 3;}
txtfile='/var/tmp/temperaturelog.txt'
giffile='/var/tmp/temperaturelog.gif'
loop=0
beta="3900"
r00='echo "10000*(1024/$adcvalue-1)"|bc -l'
while true
do
  adcvalue1='IOcmmd | telnet 192.168.178.27 2000 | grep "R B3" |tail -n 1 | awk '{ printf $3 }''
  adcvalue=$(if [ "$adcvalue1" -gt "1" ] 2>/dev/null; then echo $adcvalue1; else echo 1; fi)
  rntc='echo "10000/((1024/$adcvalue)-1)"|bc -l'
  temp='echo "$beta/l($rntc/$r00)-273" | bc -l'
  printf "%s %.2f\n" 'date +%Y-%m-%d_%H:%M' $temp >> $txtfile
  if [ "'echo $loop%24 |bc'" -eq "0" ]; then
    xlines='cat $txtfile |wc -l'
    nofxtic='echo $xlines*900/10|bc'
    printf "set terminal gif size 800,480\nset output '$giffile'\nset timefmt '%%Y-%%m-%d_%%H:%%M:%%S'\n\nset bmargin 4\nset xdata time\nset xtics rotate 270\nset xtics $nofxtic\nplot '$txtfile' using 1:2\nwith lines\n" |gnuplot
    echo Temperature |mutt -s "Temperature" xxx@gmail.com -a $giffile
  fi
  loop='echo "$loop+1"|bc'
  sleep 900
done
```

Pendant les vacances, j'utilise cette carte d'E/S avec Wi-Fi pour prendre la température et l'envoyer par courriel au moyen d'une carte Raspberry Pi. Le processus utilise un simple *script bash* pour prendre contact par une commande telnet tous les quarts d'heure avec la carte d'E/S pour Android, convertir le résultat en degrés Celsius et ajouter la valeur à un fichier. Quatre fois par jour, les données sont introduites avec GnuPlot dans un graphique au format GIF pour un envoi par

courriel (**figure 9**). Le script est celui du **listage 1**. Il faut pour cela que les programmes *bc*, *gnuplot* et *mutt* soient installés.

En outre, on peut relier directement par le bloc de cavaliers JP2 la carte d'E/S sur un port sériel de Raspberry Pi, ce qui ne nécessite aucun module de communication. La carte d'E/S fonctionne alors comme extension d'E/S pour le Raspberry Pi. Les broches 6 (Gnd), 8 (TXD) et 10 (RXD) du Raspberry Pi seront alors reliées aux trois premières broches du côté gauche de JP2 (TXD avec R et RXD avec T). On peut éventuellement brancher aussi le 3,3 V du Raspberry Pi sur la quatrième broche pour que l'alimentation soit fournie par RPi, pour autant qu'on ne dépasse pas le courant maximal que RPi est capable de délivrer.

Avec */dev/ttyAMA0*, on peut alors envoyer et recevoir par logiciel sur le port sériel. On règle le débit binaire par *stty -F /dev/ttyAMA0 speed 9600*. Comme le port sériel est normalement utilisé pour transmettre les informations d'enregistrement, il faut le débrancher, ce qui se

fait avec les adaptations */etc/inittab* et *boot/cmdline.txt*. Dans le premier fichier, il faut ajouter :

```
#T0:23:respawn:/sbin/getty -L
ttyAMA0 115200 vt100
```

Dans le deuxième fichier, il faut supprimer :

```
console=ttyAMA0,115200
kgdboc=ttyAMA0,115200
```

(150310 – version française : Robert Grignard)

Liens

- [1] <http://androidplot.com/download/>
- [2] <http://androidplot.com/>
- [3] www.elektor.fr/150057
- [4] <http://ww1.microchip.com/downloads/en/DeviceDoc/80000501F.pdf>

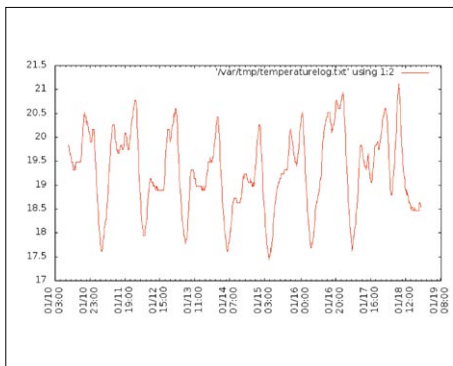


Figure 9. Le graphique des températures mesurées qui sera envoyé par courriel au moyen d'une carte RPi.

zootrope numérique

ou comment faire du neuf avec du vieux

Willem van Dreumel (Pays-Bas)



Vous cherchez de quoi occuper vos (petits-)enfants ? Voilà de quoi leur faire quitter l'écran LCD de leur tablette ou autre : le zootrope numérique ! Ce terme vous rend peut-être nostalgique, ou bien il ne vous dit rien du tout. Quoi qu'il en soit, ce « daedaleum » est un bon projet pour les sombres après-midis dominicaux d'automne, et pour les autres jours aussi bien sûr !

Cet article a une connotation nostalgique. La jeune génération actuelle ne connaît sans doute pas le zootrope [1], mais les plus anciens l'ont sûrement rencontré dans un passé lointain. Le zootrope est en quelque sorte un précurseur du disque de Nipkow [2]. Une image est mise en mouvement d'une manière relativement primitive, en utilisant le même principe que pour un film : plusieurs images défilent rapidement les unes à la suite des autres, ce qui crée ainsi une illusion de mouvement. Les images sont fixées à l'intérieur d'un cylindre rotatif, les scènes sont très courtes et souvent choisies pour donner une animation répétitive.

Ère numérique

Le schéma de ce projet (**figure 1**) montre une version plus moderne, un moteur pas-à-pas qui positionne avec précision les images les unes après les autres. Une LED puissante, illu-

minée au bon moment, donne inconditionnellement une image mobile stable.

Le circuit lui-même compte six composants. Ce petit nombre est bien entendu à mettre au compte d'un microcontrôleur, un ATiny85, qui prend en charge le pilotage. Le programme (*sketch*) Arduino en encadré veille à ce que le microcontrôleur fasse ce que l'on attend de lui. Les LED s'illuminent pendant 1 ms tous les vingt pas.

Un manuel convivial pour la programmation de l'ATiny85 – entre autres – se trouve en [3]. Le programme est écrit sous la forme d'un *sketch* Arduino, on peut donc aussi utiliser une carte Arduino pour piloter le circuit. Il faut juste veiller aux cinq ports nécessaires pour le pilotage du moteur et de la LED. Il faudra les adapter aux sorties de la carte Arduino choisie.

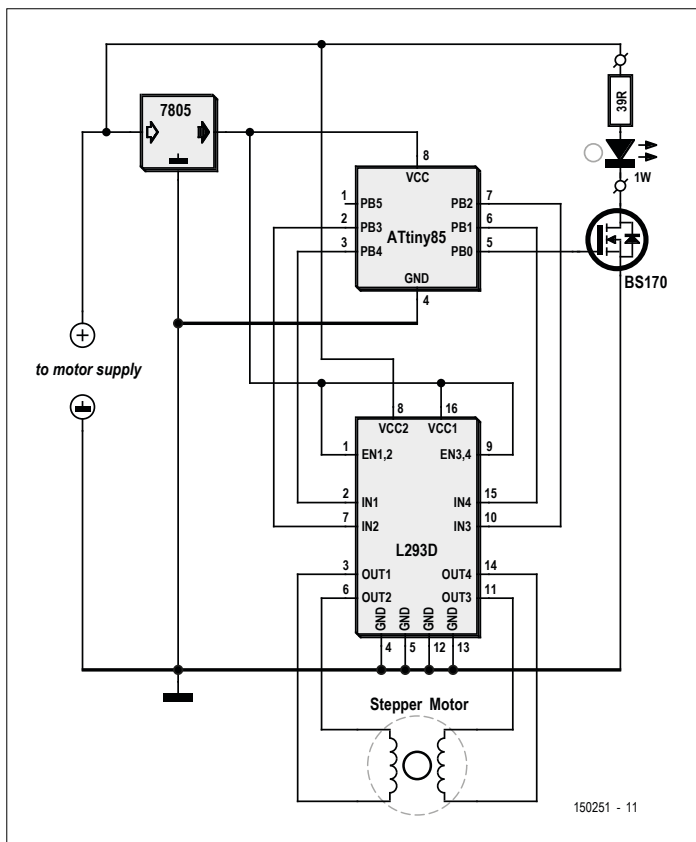


Figure 1. Le schéma ne comporte que six composants (et un moteur pas à pas). Sans condensateurs de découplage, mais ça marche.

Moteur pas-à-pas et LED

L'utilisation d'un moteur pas-à-pas bipolaire peut être très simplifiée, grâce à un circuit intégré tel que le L293D, qui possède tout ce qu'il faut pour le commander. Quatre sorties numériques de l'ATiny pilotent le L293D. Tous les vingt pas, le FET conduit et la LED s'illumine. La LED doit être un modèle puissant, par exemple 1 W, afin de réduire la gêne occasionnée par la lumière ambiante. L'alimentation du moteur peut varier, et la résistance série qui limite le courant dans la LED

dépend de la tension disponible. Par exemple, si l'alimentation est de 12 V, la résistance sera d'environ 33 Ω, avec une dissipation d'un quart de watt, compte tenu du rapport cyclique de 5 % (voir encadré sur le calcul de la résistance). Nous avons une bonne marge de sécurité. Plus la résistance est élevée, moins le courant est intense, ce qui entraîne une dissipation moins élevée que celle calculée dans le cas le plus défavorable.

Circuit imprimé

Le petit circuit imprimé de la **figure 2** accueille quatre composants ; pour des considérations pratiques, la LED, sa résistance série et le moteur ne sont pas sur le circuit imprimé. La LED (avec sa résistance) parce qu'il doit être facile de la monter au-dessus du disque, et le moteur parce que ce n'est pas nécessaire et que ça ne serait pas pratique. La LED et la résistance sont raccordées comme montré sur la figure 2. L'alimentation du moteur arrive sur le connecteur de droite. Le L293D ne supporte pas plus de 36 V, mais le 78L05 sera parti en fumée bien avant... (sa tension d'entrée maximale est de 35 V). Avec une telle tension, les deux circuits intégrés seront aussi très chauds. C'est pourquoi une tension plus basse est recommandée, la plupart des moteurs pas à pas s'en contentent aussi.

En pratique

Nous devons aussi tenir compte de la largeur des pas du moteur. Une largeur usuelle est de 1,8°. La LED s'illumine tous les vingt pas, soit 36°, nous devons donc avoir dix images sur le cylindre. Le disque accomplit un tour complet en 360°, après quoi l'animation se répète. On pourrait bien sûr utiliser un multiple de dix, mais les images seraient alors vraiment petites. Le zootrope numérique est étonnamment stable. Si le sens de rotation est incorrect, il suffit d'inverser le raccordement du moteur.

Afin de faciliter la mise en œuvre, on peut imprimer des disques d'un diamètre de 12 cm, et ensuite les coller sur des vieux CD. Il est aisé de ramener une image au diamètre de 12 cm dans un logiciel de traitement de texte ou de dessin : on imprime, on découpe, et on colle... C'est prêt !

Calcul de résistance

Pour le calcul de la résistance de limitation du courant, nous utilisons une LED de type 1 W :

- Avec 3,0 V, 200 mA circulent dans la LED, qui consomme ($P=U \times I$) 0,6 W.
- Avec 3,3 V, 350 mA circulent dans la LED, qui consomme alors 1,15 W.

Le courant peut donc être fixé approximativement entre 200 et 350 mA. Valeur et dissipation de la résistance pour le courant maximal et un rapport cyclique de $\frac{1}{20}$:

$$R = \frac{U}{I} = \frac{(12 - 3,3)V}{0,350 A} = 25 \Omega \quad \text{et}$$

$$P_{diss} = U \times I \times 0,05 = (12 - 3,3)V \times 0,35 A \times 0,05 = 0,15 W$$

Pour la limite inférieure choisie de 200 mA :

$$R = \frac{U}{I} = \frac{(12 - 3,0)V}{0,20 A} = 45 \Omega \quad \text{et}$$

$$P_{diss} = U \times I \times 0,05 = (12 - 3,0)V \times 0,20 A \times 0,05 = 0,09 W$$

En pratique on choisira un courant légèrement inférieur au courant maximal. Ceci limite la dissipation dans la résistance et la LED, ce qui est favorable à sa durée de vie. Une résistance d'environ 33 Ω est un bon compromis.

Sketch Arduino pour le zootrope numérique [5]

```
//L293D zootrope
int L0 = 1; // rotation vers la droite du moteur
int L1 = 2;
int L3 = 3; // rotation vers la gauche du moteur
int L4 = 4; // durée entre les pas
int flits = 0; // LED
int t = 1; // durée d'illumination
void setup() {
  pinMode(L0, OUTPUT);
  pinMode(L1, OUTPUT);
  pinMode(L3, OUTPUT);
  pinMode(L4, OUTPUT);
  pinMode(flits, OUTPUT);}
void loop(){
  for(int x=0;x<20;x++){ //dépend de la largeur du pas
    digitalWrite(L3, HIGH); digitalWrite(L4, LOW);
    digitalWrite(L0, HIGH); digitalWrite(L1, LOW);
```

```
    delay(t);
    digitalWrite(L3, HIGH); digitalWrite(L4, LOW);
    digitalWrite(L0, LOW); digitalWrite(L1, HIGH);
    delay(t);
    digitalWrite(L3, LOW); digitalWrite(L4, HIGH);
    digitalWrite(L0, LOW); digitalWrite(L1, HIGH);
    delay(t);
    digitalWrite(L3, LOW); digitalWrite(L4, HIGH);
    digitalWrite(L0, HIGH); digitalWrite(L1, LOW);
    delay(t);
  }
  digitalWrite(flits, HIGH);
  delay(1);
  digitalWrite(flits, LOW);
}
```

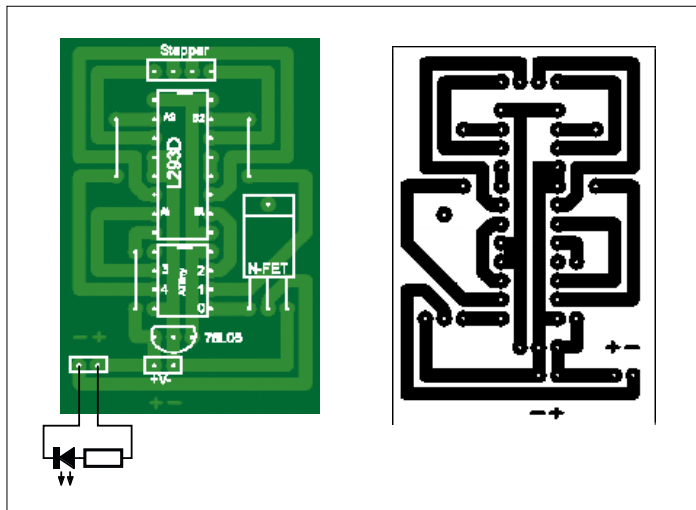


Figure 2. Circuit imprimé pour le zootrope numérique.

Si l'axe du moteur est muni d'un adaptateur pour CD, il est possible de changer de disque pendant la rotation, et les enfants en resteront bouche bée. C'est encore plus chouette de réaliser des films avec des photos ou des petits dessins. Si on colle un disque sur chaque face des CD, on peut se constituer une collection appréciable.

On trouvera en [4] une belle collection de disques pour le zootrope, et si vous entrez le terme en anglais dans un moteur de recherche (*zoetrope* ou *zoetrope disc*), vous obtiendrez bien plus d'exemples d'animations. ◀

(150251 – version française : Jean-Louis Mehren)

Liens

- [1] Zootrope sur Wikipédia : <https://fr.wikipedia.org/wiki/Zootrope>

- [2] Disque de Nipkow sur Wikipédia : https://fr.wikipedia.org/wiki/Disque_de_Nipkow
- [3] Programmation d'un ATtiny avec Arduino (en anglais) : <http://highlowtech.org/?p=1695>
- [4] Collection d'animations pour le zootrope : www.bekkahwalker.net/MediaArcheology/disks.html
- [5] Sketch Arduino : www.elektormagazine.fr/150251

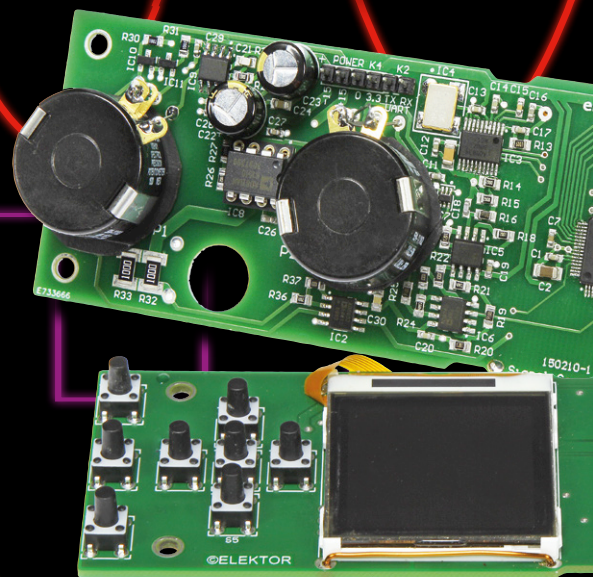


générateur de fonctions par DDS sinus, carrés et dents de scie jusqu'à 10 MHz

Theodorou Gerasimos (Grèce)

Post-ingénierie : Ton Giesberts (Elektor.Labs)

Les puces DDS (Direct Digital Synthesis) simplifient grandement la construction de la partie analogique des générateurs de fonctions. Ajoutez à votre puce DDS une circuiterie de sortie, un microcontrôleur, une interface utilisateur, et voilà. Simple comme un jeu de briques. Du moins en théorie. En pratique, ce n'est pas si facile. À moins de choisir la bonne puce !



Un jour j'ai eu besoin d'un générateur de fonctions pour mon labo personnel. Au départ j'ai envisagé l'achat d'un des modèles commerciaux que j'utilise au travail, mais ils étaient trop coûteux et trop complexes pour ce que je souhaitais faire. J'ai donc décidé de construire mon propre générateur. Et bien sûr de vous en faire profiter.

Le choix des composants

Après quelques recherches sur les différentes techniques de construction possibles, j'ai choisi une architecture exploitant la synthèse numérique directe (DDS, *Direct Digital Synthesis*). La synthèse DDS recourt à un oscillateur numérique et un quartz pour produire des sinusoïdes qui peuvent atteindre des fréquences très élevées.

Comme microcontrôleur, j'ai pris un modèle d'Analog Devices. Ce n'est peut-être pas le premier nom de fabricant qui vous serait venu à l'esprit, toutefois j'apprécie la documentation complète qui accompagne leur famille de microcontrôleurs ARM à 32 bits. Là où d'autres fabricants étalent parfois leurs descriptions sur plus d'un millier de pages, Analog Devices réussit à faire le tour des rouages d'un microcontrôleur complexe

en à peine plus d'une centaine de pages. Outre ses célèbres processeurs de traitement numérique du signal (DSP), Analog Devices propose aussi des puces DDS, des AOP et des puces pour le traitement analogique. Je me suis donc naturellement tourné vers eux, d'autant plus qu'ils offraient des échantillons des composants dont j'avais besoin.

J'ai plus précisément opté pour l'**ADuC7024BSTZ62**, un μC de la famille *Precision Analog Microcontroller*. Il est en boîtier à 64 broches et son cœur ARM7TDMI est cadencé à 44 MHz. Les contrôleurs de cette famille sont dits analogiques, car ils présentent des E/S analogiques (CA/N et CN/A) et un comparateur analogique. Ils sont par ailleurs dotés de MLI, de *timers* et de ports série standard (SPI, UART et I²C). Celui que j'ai choisi possède 8 Ko de RAM et 62 Ko de mémoire flash programmable in situ via un port série ; certains modèles utilisent l'I²C pour leur programmation, donc assurez-vous d'utiliser un μC ayant exactement la même référence que celui-ci. La **puce DDS** est la populaire **AD9834**. La fréquence maximale de son oscillateur externe est de 75 MHz, ce qui autorise une fréquence de sortie maximale de 37,5 MHz (la moitié de la fréquence

d'horloge). L'inconvénient d'un signal d'horloge aussi rapide est que la résolution n'est que de 0,28 Hz, ceci en raison du diviseur de fréquence à 28 bits de la puce. Ça n'a l'air de rien, mais à 20 Hz cela représente une erreur de presque 0,6 %. On peut certes baisser la fréquence pour y remédier, mais au prix d'une fréquence de sortie maximale moindre. La résolution passe à 0,004 Hz en optant p. ex. pour 1 MHz, soit une erreur de 0,005 % à 20 Hz ; bien, mais le hic est que la fréquence maximale tombe alors à un ridicule 500 kHz. J'ai dû trancher entre résolution et plage de fréquence : la fréquence d'horloge DDS est de 75 MHz de façon à ce que le signal reste relativement propre jusqu'à 10 MHz.

J'ai soigné l'interface utilisateur puisqu'elle compte pour beaucoup dans le confort d'utilisation d'un instrument. J'ai ajouté des poussoirs pour faciliter la navigation entre les menus et leurs options, ainsi que deux potentiomètres multitours pour le réglage de l'amplitude et du décalage CC du signal de sortie. **L'afficheur** est un écran de remplacement pour le **Nokia 6100**. Il est bon marché et bien documenté sur Internet, mais les deux modèles existants (Epson



et Philips) ne sont hélas pas entièrement compatibles. Plus embêtant, il est impossible de spécifier le modèle souhaité au moment de l'achat. J'ai donc écrit deux micrologiciels, un pour chaque modèle.

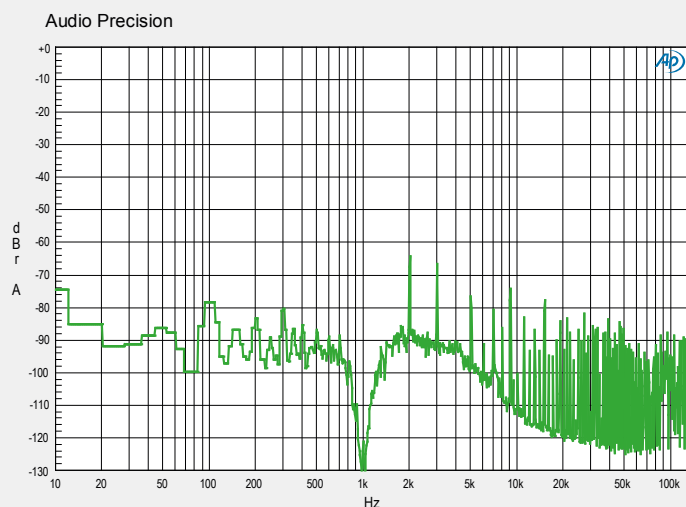
Puisque l'aspect esthétique compte aussi, j'ai habillé le générateur d'un beau **boîtier en aluminium** de chez **Hammond**. Sur ma paillasse trône ainsi un instrument très élégant.

Le circuit

Sur le schéma du générateur DDS (**fig. 1**), c'est la partie supérieure qui produit le signal ; la partie inférieure comprend quant à elle le microcontrôleur et

Caractéristiques

- Synthèse numérique directe (DDS) avec étage d'entrée analogique
- Plage de fréquences : 1 à 10 MHz
- Résolution en fréquence : 0,28 Hz
- Sortie : 0 à 15 VCC
- THD+N (charge de 100 k Ω , bande passante > 500 kHz) :
 - 1 V, 1 kHz : 0,12 % (0,09% pour bande passante = 22 kHz)
 - 5 V, 1 kHz : 0,1 % (0,09 % pour bande passante = 22 kHz)
 - 1 V, 10 kHz : 0,1 % (0,09 % pour bande passante = 80 kHz)
 - 5 V, 10 kHz : 0,09 % (0,08 % pour bande passante = 80 kHz)
 - 1 V, 100 kHz : 0,1 %
 - 5 V, 100 kHz : 0,08 %
- S/B (pour un signal de 1 V) : 72 dB
- Sortie maximale (charge de 10 M Ω) :
 - Sinus : 16 VCC
 - Triangle : 16 VCC
 - Carré : 18 VCC
- Tension de décalage CC : -10 à +10 V
- Impédance de sortie : 50 Ω
- Rapport cyclique (onde carrée) : 1 à 99 %
- Temps de montée et de descente (80 %, onde carrée) : 100 ns
- Mode balayage
- Consommation : 3 VA



Tracé jusqu'à 130 kHz de la FFT d'une sinusoïde de 1 kHz (la fondamentale a été supprimée). Le taux THD+N est principalement dû aux harmoniques du signal. Les quelques composantes visibles liées à la ligne CA (50 Hz) sont proches du bruit de fond.

Vous trouverez d'autres chiffres et courbes sur
www.elektor-labs.com/150210

l'interface utilisateur.

Les composants passifs qui entourent la puce DDS (IC3) sont ceux recommandés par le fabricant. La sortie d'IC3 est filtrée par le réseau R16/C18 avant d'être ampli-

fiée par IC5.A. S'il y a un filtre aussi rudimentaire, c'est à la fois pour des raisons de simplicité et, comme nous le verrons, parce qu'un filtre d'ordre supérieur aurait été inutile ici.

Le signal qui quitte IC5.A suit deux chemins. Celui du haut sert à la formation des sinusoïdes et des dents de scie (IC3 peut former les deux) ; il mène à l'entrée S2 d'IC7, un inverseur unipolaire qui pro-

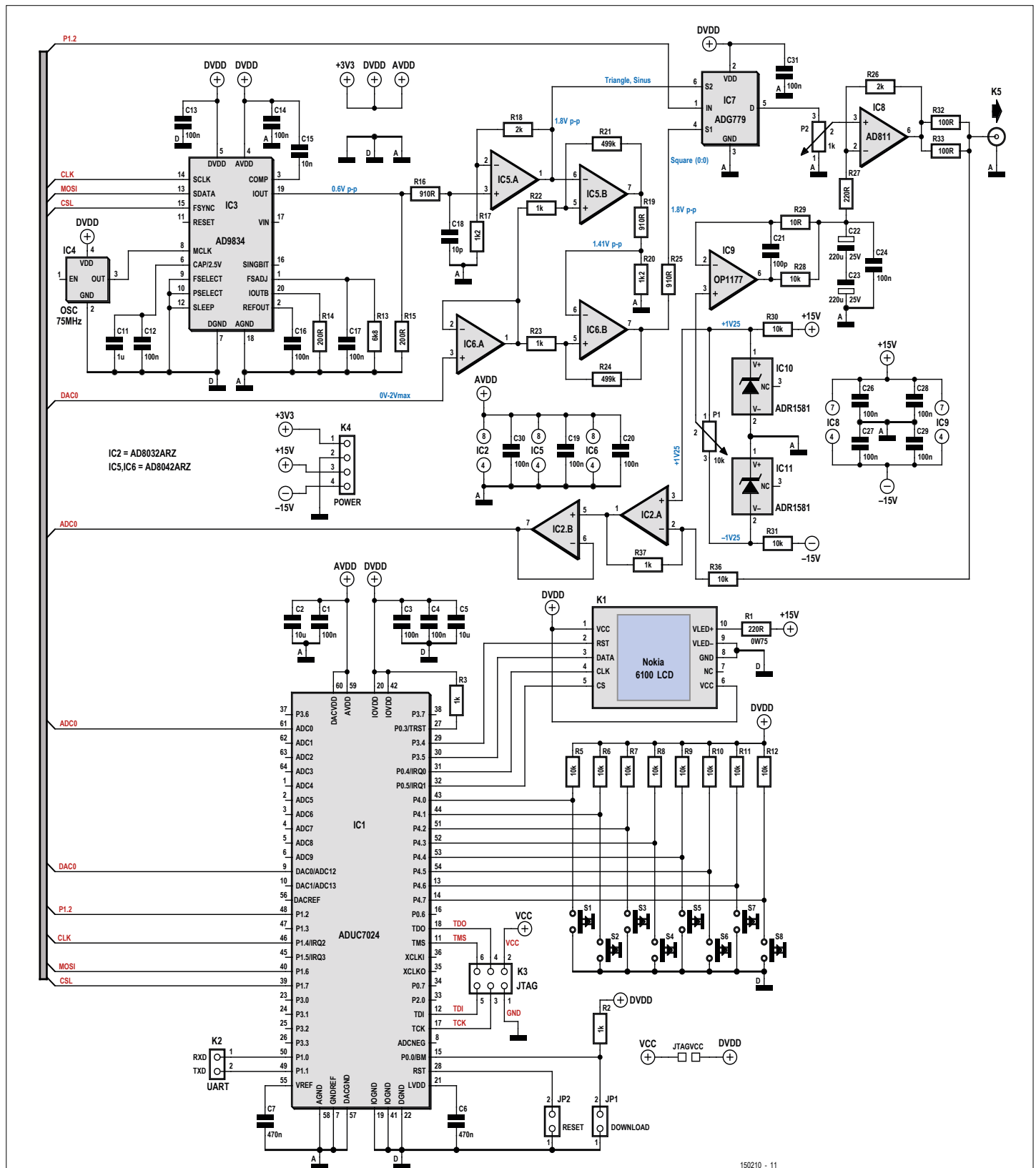


Figure 1. À l'exception de l'alimentation, la carte principale accueille tous les composants du générateur.

cure une très bonne isolation entre ses deux entrées et la sortie. Le chemin du bas va à l'entrée d'IC7 et est utilisé pour former les ondes carrées.

Une onde carrée est créée en produisant un signal triangulaire transmis à IC5.B et IC6.B, deux comparateurs relativement rapides dotés d'une hystérésis bien dosée. La tension de référence nécessaire à ces deux comparateurs provient d'une des sorties analogiques (DAC0) du microcontrôleur et est tamponnée par IC6.A. Cela permet un contrôle précis du rapport cyclique de l'onde carrée.

La sortie d'IC7 commande le potentiomètre P2 puis est amplifiée par l'AOP IC8. Cet amplificateur vidéo est rapide (vitesse de balayage : 2500 V/ms) et peut commander des charges de faible impédance. Notez qu'il s'agit ici d'un composant traversant, donc facilement remplaçable en cas d'avatar.

L'amplitude crête à crête du signal de sortie est définie par P2. Ce potentiomètre influe de manière fondamentale sur la qualité du signal de sortie en raison de ses capacités et inductances parasites (qui le font se comporter comme un filtre passe-bas). Sa fréquence de coupure dépend de sa résistance et de son nombre de tours. Si vous utilisez surtout le générateur pour avoir des sinusoïdes propres et de fréquence élevée, un modèle à dix tours et de 1 k Ω ou plus convient. Si par contre vous avez besoin d'impulsions à flancs raides, optez pour un modèle à cinq tours et de 200 Ω ou moins (**fig. 2**).

Puisque nous parlons de filtres indésirables, observez les chemins des signaux : tous les composants qui s'y trouvent, depuis la sortie d'IC3 et y compris le circuit imprimé, ont une bande passante limitée et contribuent partiellement au filtrage passe-bas du signal de sortie. C'est la raison pour laquelle je n'ai placé qu'un filtre RC du premier ordre (R16/C18). Avec un filtre d'ordre supérieur, il aurait fallu une troisième voie pour acheminer l'onde triangulaire, et le circuit aurait été plus complexe.

La tension de décalage du signal de sortie est créée par les deux références de tension IC10 et IC11 et par le potentiomètre P1, tamponné par l'AOP de précision IC9. L'AOP IC8 ajoute cette tension CC au signal de sortie.

Le signal de sortie alimente une des entrées analogiques du microcontrôleur

(ADC0) après mise à l'échelle et suppression du décalage CC (IC2). Le but est de surveiller le niveau du signal de sortie tel qu'indiqué sur l'afficheur. Consultez le lien [2] pour les mesures.

La circuiterie qui entoure le μ C IC1 parle d'elle-même : huit pushers, un LCD graphique, des condensateurs de découplage, et quelques embases. Rien de mystérieux, donc. Pas besoin de quartz externe, le contrôleur est cadencé par un oscillateur interne.

Notez le connecteur JTAG : c'est lui qui m'a servi à programmer et déboguer le micrologiciel. Le port série est également accessible au cas où vous seriez obligé d'utiliser le chargeur d'amorçage pour programmer le μ C. JP1 sert à placer le μ C dans ce mode *bootloader*.

L'alimentation (**fig. 3**) est construite autour de deux LM317 et d'un LM337. Ces régulateurs produisent une tension stable de ± 15 V pour l'étage de sortie analogique, et de +3,3 V pour le reste du circuit. Contrairement à la carte principale, composée quasi exclusivement de

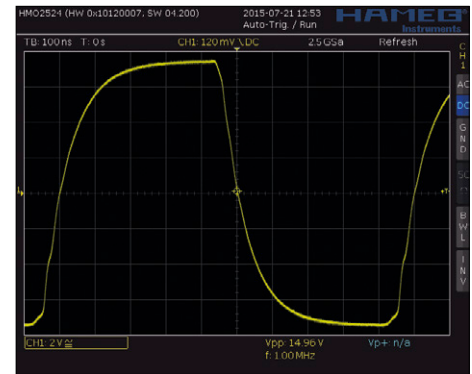


Figure 2. Une onde carrée de 1 MHz (rapport cyclique de 50 %) obtenue avec la sortie presque maximale et un potentiomètre P2 de dix tours et 1 k Ω .

CMS, l'alimentation recourt à des composants traversants de la liste de composants Passe-Partout du labo d'Elektor. Les secondaires du transformateur valent chacun 15 V / 5 VA puisque 10 VA suffisent à alimenter le générateur DDS. Pour minimiser les pertes dans le régulateur de 3,3 V, le circuit comporte un redres-

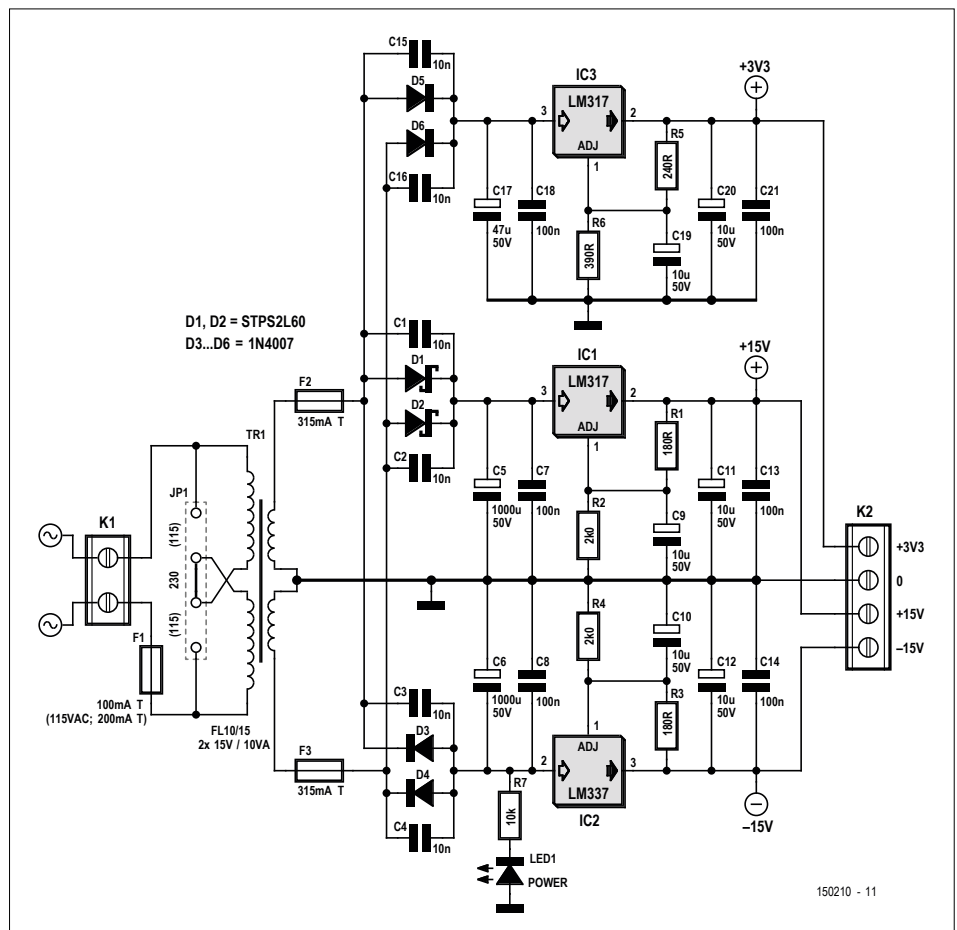


Figure 3. L'alimentation du générateur comprend trois régulateurs de tension LM3x7. Quelques tours de passe-passe ont permis de réduire la consommation de la partie à 3,3 V.

seur séparé (D5/D6) flanqué d'un simple filtre à condensateurs (C17/C18). La forte ondulation restante réduit quelque peu les pertes dans le régulateur IC3.

D1 et D2 sont des diodes Schottky. Lorsque la charge est maximale, la chute de tension aux bornes de diodes « ordinaires » et l'ondulation sur C5 font dangereusement approcher la tension d'entrée d'IC1 de sa valeur minimale permise. À 0,5 A la tension directe des Schottky est inférieure à 0,45 V, alors qu'elle serait

deux fois moindre avec une classique 1N4007.

L'alimentation et le circuit principal sont sur deux cartes séparées. La carte principale a été soigneusement conçue de façon à ce que le signal de sortie reste éloigné des interférences produites par les rapides signaux de commande numériques. Les CMS dominant, mais l'assemblage ne devrait pas poser de problème avec de bons yeux ou de bons verres de lunettes.

Le programme

J'ai écrit le code source en C depuis l'environnement de développement intégré µVision de Keil. Je n'ai pas utilisé de système d'exploitation en temps réel par souci de simplicité.

Le code est réparti sur quelques fichiers. Certains sont des fichiers d'en-tête contenant des définitions et des prototypes de fonctions, les autres contiennent les fonctions elles-mêmes. Deux d'entre eux prennent en charge les deux modèles de

Liste des composants de la Carte du générateur

Résistances

Toutes 1 %, 0,125 W, CMS 0805
 R1 = 220 Ω, 1 %, 0,75 W, CMS 2010
 R2, R22, R23, R37 = 1 kΩ
 R3 = 1 kΩ, 1 %, 0,1 W, CMS 0603
 R5, R6, R7, R8, R9, R10, R11, R12, R30, R31, R36 = 10 kΩ
 R13 = 6,8 kΩ
 R14, R15 = 200 Ω
 R16, R19, R25 = 910 Ω
 R17, R20 = 1,2 kΩ
 R18, R26 = 2,0 kΩ
 R21, R24 = 499 kΩ
 R27 = 220 Ω
 R28 = 10 Ω
 R29 = 10 kΩ, 1 %, 0,1 W, CMS 0603
 R32, R33 = 100 Ω, 1 %, 0,75 W, CMS 2010
 P1 = 10 kΩ, 2 W, pot. 10 tours
 P2 = 1 kΩ, 2 W, pot. 10 tours

Condensateurs

Par défaut : CMS 0603
 C1, C3, C4, C12, C13, C14, C16, C17, C19, C20, C26, C27, C28, C29, C31 = 100 nF, 50 V, X7R

C2, C5 = 10 µF, 16 V, X7R, CMS 1206
 C6, C7 = 470 nF, 25 V, X7R
 C11 = 1 µF, 16 V, X7R, CMS 1206
 C15 = 10 nF, 50 V, X7R
 C18 = 10 pF, 50 V, COG/NP0
 C21 = 100 pF, 100 V, COG/NP0, CMS 0805
 C22, C23 = 220 µF, 25 V, radial, pas de 3,5 mm, diam. 8 mm max.
 C24, C30 = 100 nF, 50 V, X7R, CMS 0805

Semi-conducteurs

IC1 = ADUC7024BSTZ62, LQFP-64, programmé*
 IC2 = AD8032ARZ, SOIC-8
 IC3 = AD9834BRUZ, TSSOP-20
 IC4 = FXO-HC736R-75, 7 x 5 mm
 IC5, IC6 = AD8042ARZ, SOIC-8
 IC7 = ADG779BKSG-REEL7, SC-70 6 pattes
 IC8 = AD811ANZ, DIP-8
 IC9 = OP1177ARZ, SOIC-8
 IC10, IC11 = ADR1581ARTZ-REEL7, SOT-23-3

Autres

K1 = embase, pas de 0,5 mm, hauteur des contacts 1,5 mm, 10 voies,

DF23C-10DS-0.5V(51), Hirose (HRS)
 K2, JP1, JP2 = barrette mâle 1x2 broches, verticale, pas de 2,54 mm
 K3 = barrette mâle 2x3 broches, verticale, pas de 2,54 mm
 K4 = barrette mâle 1x4 broches, verticale, pas de 2,54 mm
 K5 = prise BNC droite encartable, 50 Ω
 S1 à S8 = bouton à effleurement 6 mm, actionneur de 4,9 mm, 24 V / 0,05 A, SPST-NO
 Support DIP à 8 voies pour IC8
 Écran LCD de remplacement N&B pour Nokia 6100

Divers

Boîtier aluminium, modèle Hammond 1455T1601, 165 x 160 x 51,5 mm
 Optionnel : filtre EMI/EMC, prise, CEI, 250 VAC / 4 A
 2 boutons, noir, 16 mm, diam. axe 6,3 mm
 Circuit imprimé 150210-1 v1.11 (www.elektor.fr)

* Indisponible en version préprogrammée

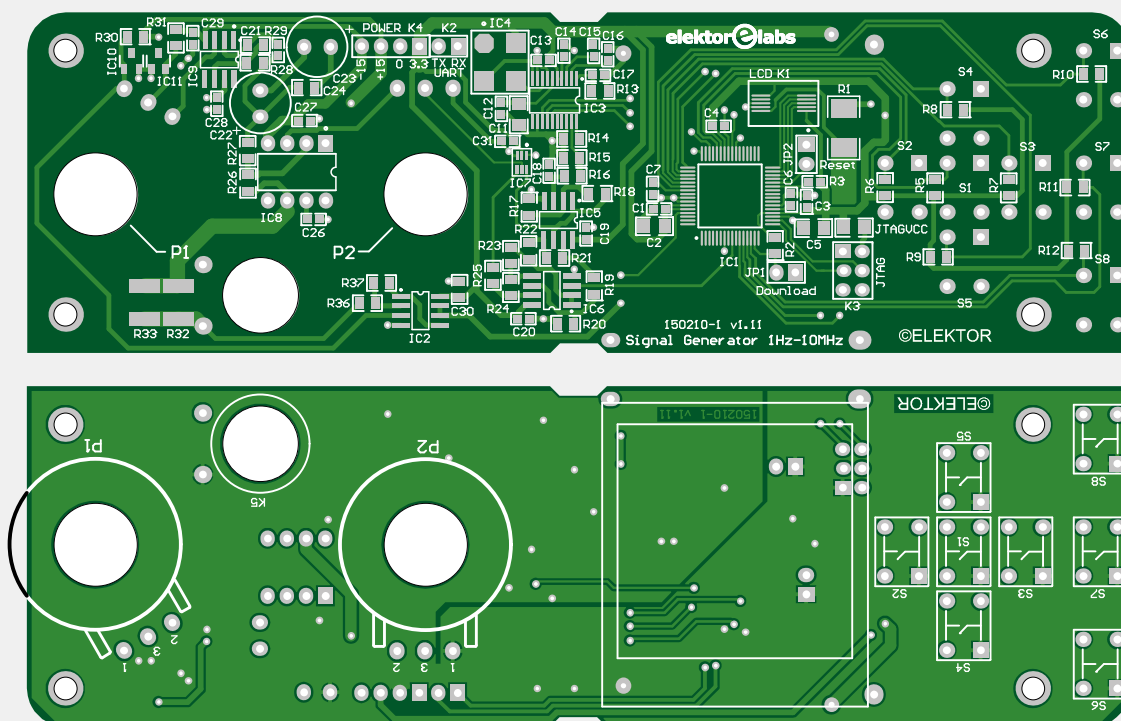


Figure 4a. Sérigraphie de la carte du générateur (double face et composants CMS).

LCD graphiques. Le code du générateur est dans le fichier *main.c*. Le fichier *init.s* est écrit en assembleur et enferme les fonctions d'initialisation du μC ; il ne fait pas partie de l'EDI, je l'ai écrit en partant de zéro.

J'ai écrit une partie du pilote du LCD, le reste repose sur un code source ouvert trouvé sur l'internet. Comme l'écran n'est pas relié à un port SPI matériel, le protocole de communication est reproduit de façon logicielle sur les broches GPIO au travers des fonctions *WriteLcdCommand* et *WriteLcdData*. Ce qui m'a pris le plus de temps ici a été de créer deux polices, une grande et une petite.

L'activation du « clavier » est scrutée dans la boucle sans fin principale.

Je me suis efforcé de n'utiliser ni arithmétique en virgule flottante, ni bibliothèque de maths, ceci afin que le code reste peu volumineux et indépendant de l'EDI.

Tous les fichiers source ont été compilés en « code Thumb » ARM (code à 16 bits), pas tant pour réduire la taille du fichier hex qu'en raison de la mémoire flash à 16 bits du contrôleur : un code à 16 bits y est donc plus vite exécuté. L'exécutable fait environ 8 Ko, auxquels s'ajoutent 20 Ko pour stocker l'écran d'accueil. Je

n'ai eu besoin d'aucune astuce pour faire tenir l'ensemble dans le μC , sa mémoire flash est assez vaste pour ça.

L'archive du projet peut être téléchargée gratuitement depuis [1].

Construction

Les composants du générateur sont répartis sur deux circuits imprimés (fig. 4). Comme je l'ai dit, l'assemblage de la carte principale ne présentera pas de difficultés pour qui a de bons yeux... ou une bonne

loupe. Notez que les poussoirs et le LCD se montent côté cuivre de la carte.

Restez soigneux avec le câble plat souple du LCD : il doit être détaché de son support en plastique (fig. 5) et plié autour de la carte pour atteindre K1 (fig. 6).

Les deux potentiomètres s'insèrent dans le circuit imprimé en laissant dépasser l'axe côté cuivre. Le labo d'Elektor a utilisé des « disques » en caoutchouc (découpés dans une vieille chambre à air de vélo) pour éviter que les potenti-

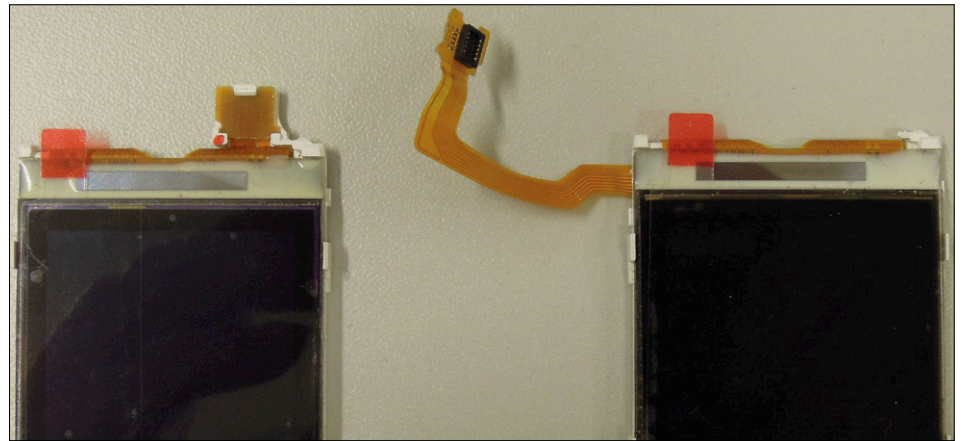


Figure 5. À gauche, l'écran autrefois utilisé dans certains mobiles d'une célèbre marque scandinave qui n'est pas Ikea ; à droite, le même écran adapté à notre générateur.

Liste des composants de l'Alimentation

Résistances (1 %, 0,6 W)

R1, R3 = 180 Ω
R2, R4 = 2,0 k Ω
R5 = 240 Ω
R6 = 390 Ω
R7 = 10 k Ω , 5 %, 0,25 W

Condensateurs

C1, C2, C3, C4, C15, C16 = 10 nF, 50 V, Y5V, pas de 5 mm
C5 = 1000 μF , 50 V, pas de 5 ou 7,5 mm, diam. 16 mm
C6 = 470 μF , 50 V, pas de 5 ou 7,5 mm, diam. 13 mm
C7, C8, C13, C14, C18, C21 = 100 nF, 50 V, X7R, pas de 5 mm
C9, C10, C11, C12, C19, C20 = 10 μF , 50 V, pas de 2 mm, diam. max. 6,3 mm
C17 = 47 μF , 50 V, pas de 2,5 mm ou 3,5 mm, diam. max. 8 mm

Semi-conducteurs

D1, D2 = STPS2L60, boîtier DO-41
D3, D4, D5, D6 = 1N4007, boîtier DO-41
IC1, IC3 = LM317, boîtier TO-220
IC2 = LM337, boîtier TO-220
LED1 = LED, verte, 3 mm

Divers

K1 = bornier pour CI à 2 voies, pas de 7,6 mm, 500 V

K2 = bornier pour CI à 4 voies (2x2), pas de 5 mm, 250 V
TR1 = prim. 2x115 V / sec. 2x15 V, 10VA, p. ex. Block FL 10/15
F1 = fusible, 100 mA (réseau 230 V) ou 200 mA (réseau 115 V) ; action retardée, 250 V, 20 x 5 mm
F2, F3 = fusible, 315 mA, action retardée, 250 V, 20 x 5 mm

Porte-fusible pour F1, F2, F3, 20 x 5 mm, 500 V, 10 A
Caches pour porte-fusibles F1, F2, F3, 20 x 5 mm
JP1 = cavalier
Circuit imprimé 150210-2 v1.1
(www.elektor.fr)

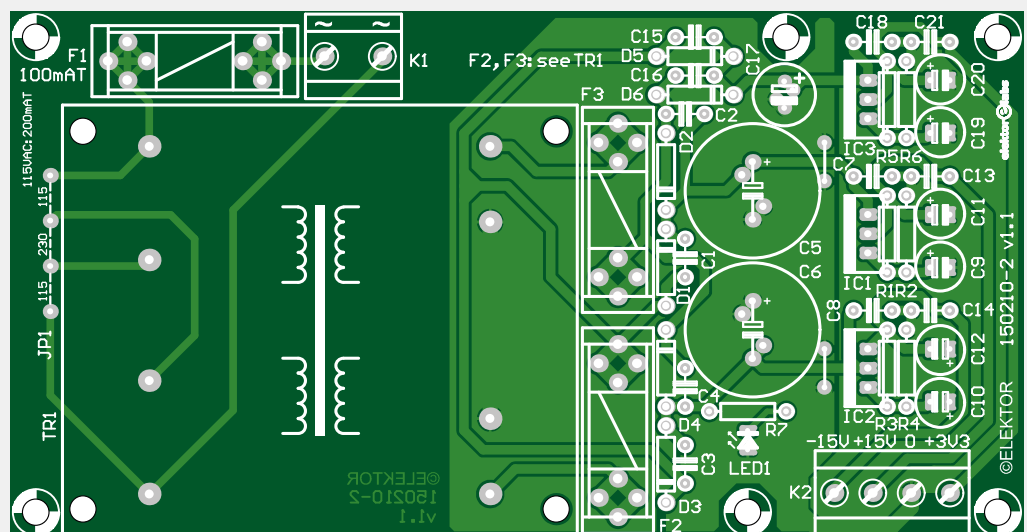


Figure 4b. Sérigraphie de la carte d'alimentation (simple face et composants traversants).

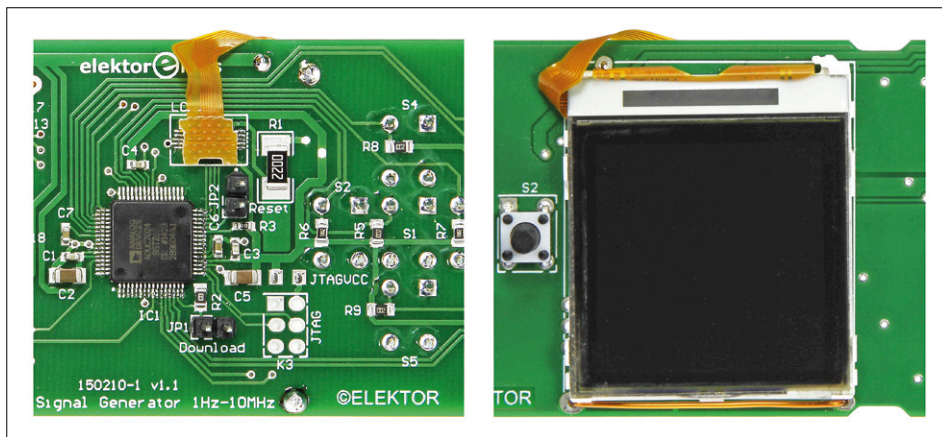


Figure 6. Montage de l'écran des deux côtés de la carte. Utilisez de l'adhésif double face pour le coller à la carte.

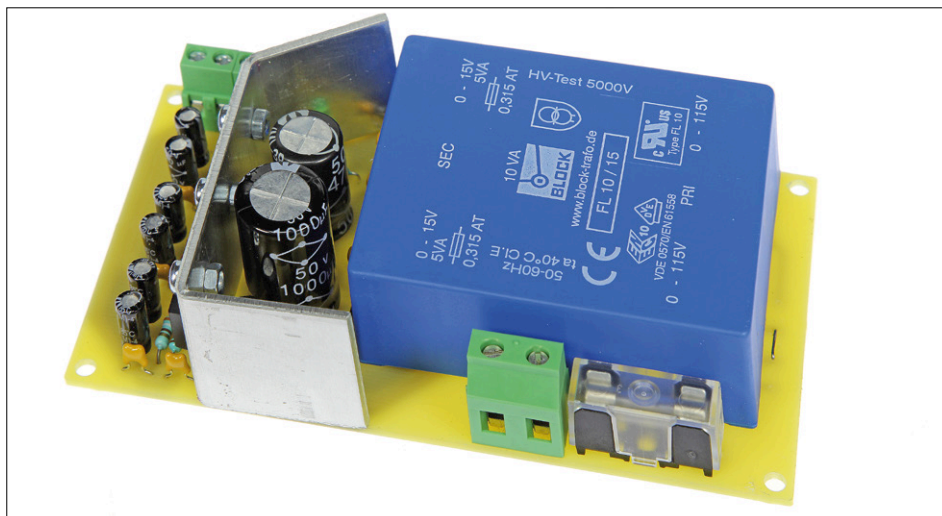


Figure 7. L'alimentation assemblée et son dissipateur maison. Notez le cavalier derrière le transformateur pour la sélection de la tension de réseau (230 V ici).



Figure 8. Le prototype construit par Jan Visser au labo Elektor. Découpes faites avec une machine-outil à 100 k€ des outils simples et de la patience.

mètres ne glissent lors des réglages. Un trou de diamètre 10 mm convient ; le diamètre extérieur des disques doit de préférence être légèrement inférieur à 22 mm. Pour P2, la présence d'un disque évite aussi d'endommager la piste qui jouxte son trou côté composants.

Le montage de la carte d'alimentation ne devrait poser aucun problème. Le transformateur a deux secondaires pour supporter à la fois les réseaux de 115 et 230 V. Placez un cavalier sur JP1 (au milieu des trois lignes pointillées) pour un réseau de 230 V ; placez deux cavaliers (sur les deux lignes pointillées extérieures) si vous êtes sur du 115 V. Ne placez pas les trois cavaliers sur JP1 ! N'oubliez pas d'utiliser le bon fusible pour protéger le primaire : 100 mA(T) pour du 230 V, et 200 mA(T) pour du 115 V, où T est le retard.

Les trois régulateurs ont besoin d'un dissipateur. Celui de la **figure 7** a été fabriqué à partir d'une bande d'aluminium de 2 mm d'épaisseur. N'oubliez pas l'isolation électrique de ces trois régulateurs (rondelle en mica et manchon en plastique). La longueur de la vis M3 doit être d'environ 6 mm. Il est conseillé d'ajouter une rondelle métallique entre la tête de la vis et le manchon en plastique. Le manchon est souvent un peu trop long ; si c'est le cas, coupez-le à la bonne longueur avant de monter les régulateurs sur le dissipateur. Si vous souhaitez loger le générateur dans le même boîtier que le mien (**fig. 8**), vous pouvez télécharger depuis [1] un gabarit pour le perçage des panneaux avant et arrière.

L'embase BNC K5 est isolée ; elle se monte sur le panneau avant. Après avoir fixé la carte principale à ce panneau (j'ai collé les vis à l'arrière du panneau), utilisez des fils courts pour relier l'embase BNC au circuit imprimé.

Programmation

Le générateur doit d'abord être programmé avec le bon micrologiciel. Il en existe deux versions (à télécharger depuis [1]) puisqu'il y a deux configurations possibles pour le LCD. Comment savoir laquelle est la bonne ? Pas d'autres solutions que de les essayer puisqu'il n'existe aucun moyen visuel de connaître le modèle du LCD. Donc programmez le μC , et regardez si des ondes apparaissent. Pas d'ondes ? Essayez l'autre. Toujours rien ? Oh, oh...

Vous pouvez programmer le contrôleur par JTAG ou par chargeur de démarrage.

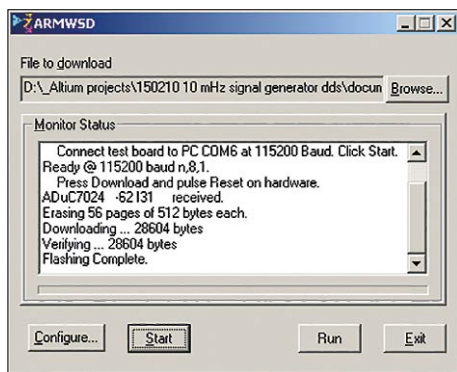


Figure 9. Capture d'écran de l'outil de programmation ARMWSD en plein travail.

La première solution, qui nécessite un adaptateur JTAG (p. ex. le J-LINK de Segger ou le ULINK de Keil), fournit une interface de débogage. L'interface JTAG standard a 20 broches mais fonctionne également avec seulement 6 broches (K3). L'autre solution est d'utiliser l'interface du port série (K2). Seule la programmation est possible ici. Autrement dit : pas de

Matériel disponible sur www.elektor.fr

- 150210-1 – circuit imprimé de la carte principale, sans composants
 - 150210-2 – circuit imprimé de la carte d'alimentation, sans composants
 - 150210-91 – carte principale, préassemblée
- Hélas, nous ne pouvons pas proposer le microcontrôleur IC1, ni vierge, ni programmé.

débogage. Analog Devices vend le câble nécessaire, mais vous pouvez fabriquer le vôtre avec un câble série TTL-vers-USB. Le site web d'Analog Devices propose un outil de programmation in situ gratuit (ARMWSD.exe). Sélectionnez d'abord le port série USB approprié, puis chargez le fichier hex. Le programme vous invitera à *Press Download and pulse Reset on hardware* (fig. 9). C'est la raison pour laquelle les deux cavaliers JP1 et JP2 portent les mentions *Download* et *Reset* sur la carte. Pensez à retirer le cavalier de *Download* après la programmation, sinon à la prochaine mise sous tension le contrôleur sera encore en mode *bootloader* et l'écran restera noir.

Et là c'est moi...

... qui vous envoie des ondes positives : avec ce bel instrument vous prendrez plaisir à réparer des amplificateurs, vous découvrirez que tracer la courbe de réponse d'un filtre est un jeu d'enfant, et le langage des ondes n'aura plus de secret pour vous. Bienvenue dans le monde merveilleux des électroniciens bien équipés ! ◀

(150210-I – version française : Hervé Moreau)

Liens

- [1] www.elektormagazine.fr/150210
- [2] www.elektor-labs.com/150210

Manuel de l'utilisateur

Bouton	Fonction	Bouton	Fonction
S1	Set	S5	-Down
S2	Right	S6	Mode
S3	Left	S7	Sweep
S4	+Up	S8	Calibration

- **Forme d'onde** – Appuyez sur *Mode* pour basculer entre sinus, carré et triangle.
- **Rapport cyclique** – Il ne peut être défini qu'en mode onde carrée. Appuyez sur *Mode* pour activer la sortie onde carrée. La valeur du rapport est indiquée en bas de l'écran. Ajustez-le en pressant *+Up* et *-Down* (les chiffres ne doivent pas clignoter).
- **Fréquence** – Appuyez sur *Set*. Un chiffre clignote ; utilisez *+Up* et *-Down* pour modifier sa valeur, et *Left* et *Right* pour passer aux autres chiffres. Appuyez sur *Set* pour valider.
- **Amplitude** – Ajustez *P2*. Notez que le réglage de l'amplitude affecte la tension de décalage. Cf. [2] pour le détail des mesures.
- **Décalage CC** – Ajustez *P1*.
- **Balayage de fréquence** – Appuyez sur *Sweep* pour ouvrir le menu *Sweep*. Le chiffre le moins significatif de la fréquence de départ clignote ; utilisez *+Up* et *-Down* pour modifier sa valeur, *Left* et *Right* pour passer aux autres chiffres. Appuyez sur *Set* pour régler le paramètre suivant. Définissez la fréquence d'arrêt, la durée du balayage (appelée *msec*) et le mode (linéaire

ou logarithmique). *Set* lance le balayage : la première ligne du menu *Sweep* indique *sweep run*. Appuyez de nouveau sur *Set* pour l'arrêter (*sweep stop* s'affiche). Vous pouvez définir de nouvelles valeurs. Pressez *Sweep* pour revenir au menu principal.

- **Contraste** – Appuyez sur *Calibration* pour accéder au menu permettant de régler le contraste du LCD. Utilisez *Set* pour naviguer entre les options, *+Up* et *-Down* pour modifier le niveau de contraste. Appuyez sur *Calibration* pour revenir au menu principal.
- **Niveaux de tension de calibrage** – Reliez un oscilloscope à la sortie du générateur et réglez le niveau de sortie sur 5 V_{cc}. Appuyez sur *Calibration* pour ouvrir le menu *Calibration*. Sélectionnez *Measurements* pour lancer la procédure (si vous y entrez par erreur, coupez l'alimentation pour en sortir). Ajustez *P1* pour définir la valeur minimale de sortie sur 0,00 V ; appuyez sur *Set* pour valider. Ajustez *P1* pour mettre la valeur maximale de sortie sur 12,00 V ; *Set* pour valider. Un message signale la fin du calibrage. Appuyez sur *Calibration* pour revenir au menu principal.
- **Fréquence de calibrage** – Reliez un fréquencemètre de précision à la sortie du générateur. Appuyez sur *Calibration* pour ouvrir le menu *Calibration*. Sélectionnez *Frequency* pour lancer le calibrage (si vous y entrez par erreur, coupez l'alimentation pour en sortir). Réglez la fréquence de sortie sur 100 kHz avec *+Up* et *-Down*. Bouton *Set* pour valider. Un message signale la fin du calibrage. Appuyez sur *Calibration* pour revenir au menu principal.

robot analogique de l'idée au kit

Tino Werner (Allemagne)

Le but du kit de construction du robot « tibo » est que les jeunes se passionnent pour la technique et le « je l'ai fait moi-même ». Leurs aînés y trouveront aussi leur compte : le cœur de ce robot insolite est un amplificateur opérationnel ; il se passe donc de microcontrôleur et de logiciel. Des résistances et des condensateurs enfichables servent à réaliser les circuits de commande simples, qui définissent le comportement — étonnamment complexe — du robot. Comme l'inventeur nous le relate ici, le chemin, de l'idée au kit fini, fut long et caillouteux.

Tout jeune, la construction d'automates, en particulier de robots mobiles qui réagissent à leur environnement de manière autonome, m'a enthousiasmé. À cette époque, je construisais mes projets avec des éléments Fischertechnik. En 1993, j'ai voulu réaliser une araignée-robot, aux mouvements aussi naturels que possible, mais avec un minimum de mécatronique, toutefois j'ai rapidement atteint, en termes de stabilité, les limites de ce système modulaire. J'ai donc fraisé le corps de l'araignée dans du bois et lui ai donné des pattes en aluminium. La transmission des entraînements était de l'autoconstruction elle aussi.

Je possédais bien des connaissances de base sur les microcontrôleurs, mais comme il n'existait ni Arduino, ni Raspberry Pi ni d'autre système similaire à l'époque, l'utilisation d'un circuit de logique câblée me parut le plus facile. L'unité de commande, je pouvais la développer à l'école dans le cadre d'un projet. À l'aide d'un compteur binaire, je produisais les mouvements pseudo-aléatoires de l'araignée. Sans qu'on puisse le prédire, elle changeait de direction ou restait immobile pendant plus ou moins longtemps. De plus, elle bougeait ses péripalpes et son abdomen — tout cela de façon apparemment aléatoire. Un système de capteurs à barrières de lumière infrarouge détectait les obstacles : avant que l'araignée n'entre en collision avec un objet, elle s'arrêtait un court instant, reculait et se tournait pour repartir dans une autre direction. La combinaison de fonctions et « décisions aléatoires » donnait (presque) l'impression que l'on avait affaire à une créature intelligente. Ma mère m'a aidé à coudre une « peau » qui transforma le robot en une tarentule géante, de 60 cm (**fig. 1**) !

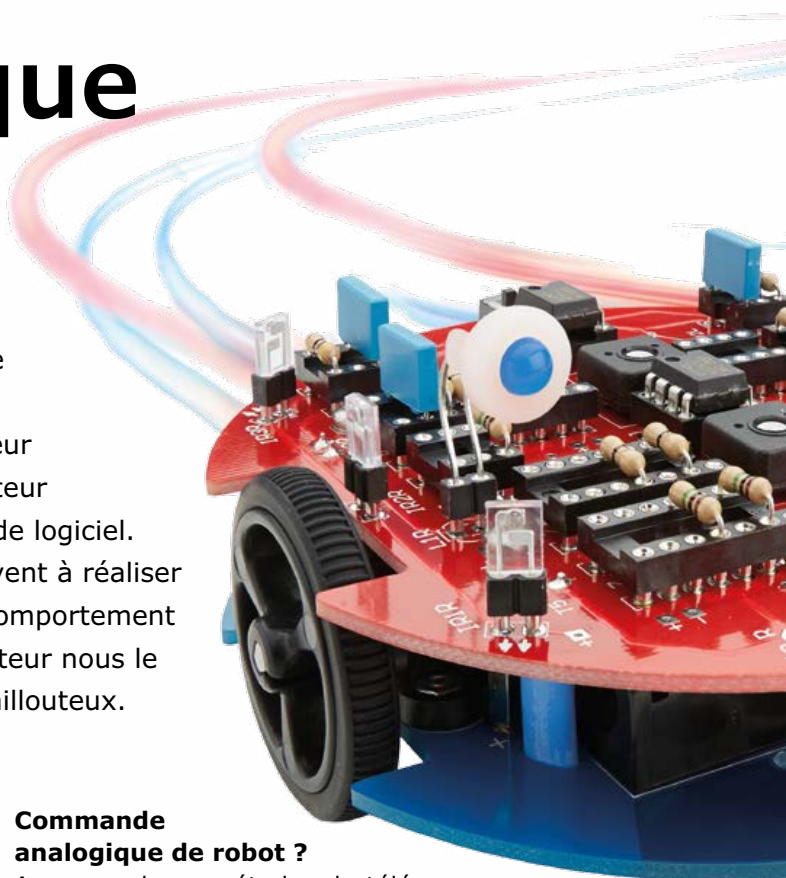
Avec ce projet et ceux qui suivirent, je ne cherchais pas à réaliser des fonctions clairement définies, comme c'est généralement le but en robotique. J'étais bien plus intrigué par la question de savoir quand nous, êtres humains, percevions le comportement biologique d'un être comme vivant et comment pouvait-on modéliser un tel comportement de manière technique et efficace.

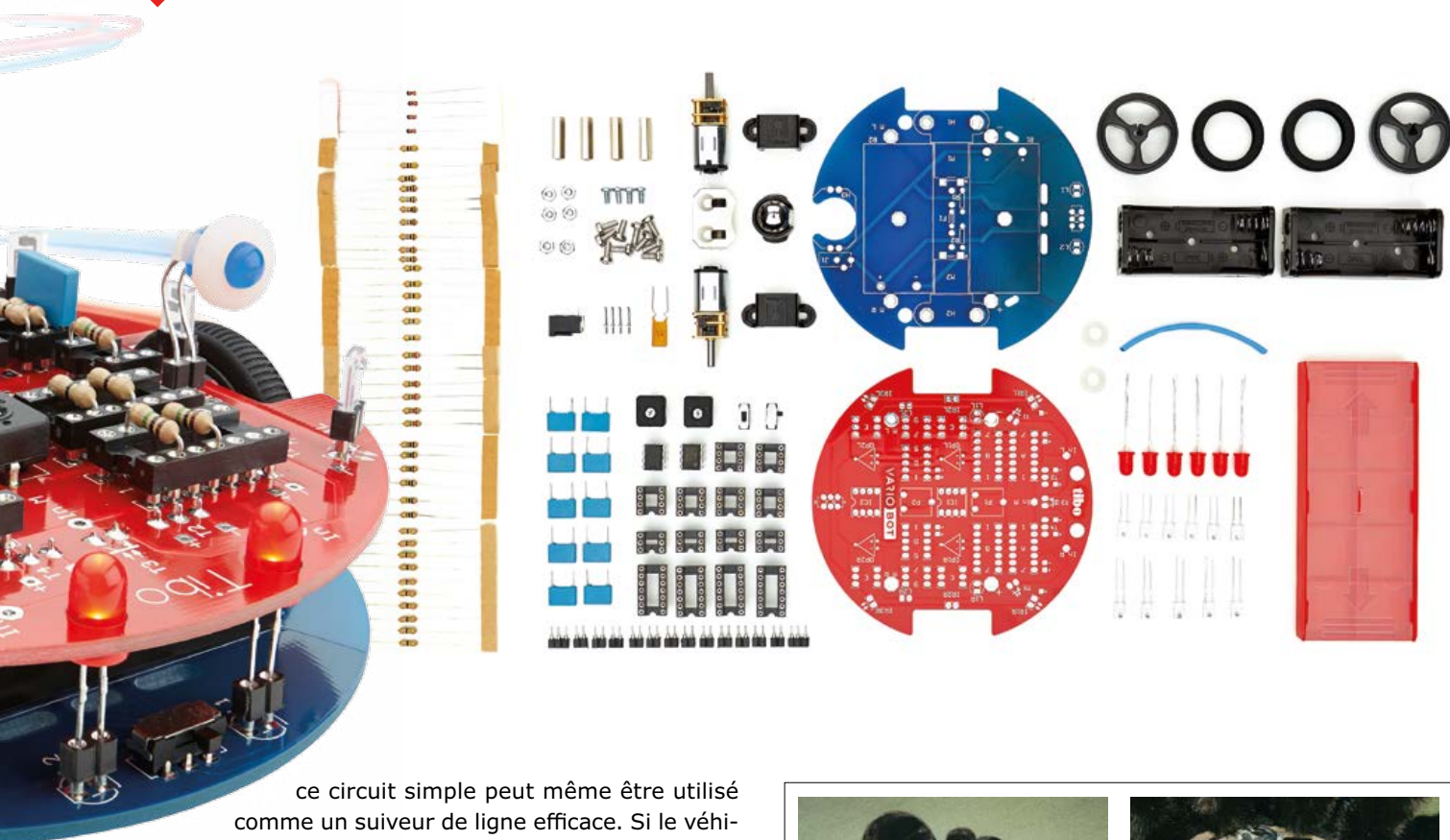
Commande analogique de robot ?

Au cours de mes études de télématique, j'ai découvert en 1999 le véhicule de Braitenberg [1]. Ces expériences décrivent comment des véhicules simples peuvent avoir des comportements étonnamment complexes par l'interconnexion directe des capteurs et des moteurs d'entraînement. Fasciné par la simplicité et l'efficacité de cette approche, je me suis mis à développer et optimiser de nombreuses variantes d'intégration de capteurs et de traitement des signaux analogiques sur la base de robots mobiles simples dotés de photocapteurs.

L'innovation majeure fut alors la combinaison, par groupe, de plusieurs capteurs montés en série. Ceux-ci ne fournissent pas comme d'habitude les mesures absolues dans une zone de travail limitée, mais donnent plutôt, à partir de n (2 ou plus) photocapteurs, jusqu'à $n - 1$ valeurs de luminosité relatives, utilisables pour la navigation et la détection d'obstacle. Notre propre organe visuel connaît lui aussi une perception relative de la luminosité : la perception d'un objet se réfère toujours à la luminosité de l'environnement. Nombre d'illusions d'optique permettent de le prouver de façon impressionnante.

Le circuit à transistors simple de la **figure 2** montre le principe d'une telle commande pour un robot à commande différentielle sur lequel le capteur S1 est orienté vers la droite et S2 vers la gauche. Un tel robot navigue « intelligemment » en fonction de la luminosité relative de l'environnement sans être limité à une certaine plage de luminosité. Si on positionne les capteurs de façon à ce qu'aucune lumière directe ne les frappe, le véhicule esquive également des obstacles, sachant qu'ils réduisent la quantité de lumière qui atteint le capteur concerné. Si on intercale un autre capteur entre S1 et S2, le véhicule s'arrête de façon fiable devant des obstacles frontaux. Si les capteurs regardent à distance convenable vers le sol,





ce circuit simple peut même être utilisé comme un suiveur de ligne efficace. Si le véhicule doit suivre une ligne foncée, il suffit d'inverser les seuls capteurs S1 et S2. Pour la réalisation de ce circuit, on pourra utiliser des transistors Darlington ; ils ont un gain suffisant pour piloter directement de petits moteurs à courant continu.

Amplificateurs opérationnels transformés en cellules cérébrales

Dès le début j'ai choisi d'utiliser des amplis op comme « cellules cérébrales » analogiques. Les amplis op de puissance peuvent traiter les signaux analogiques haute impédance de capteurs, mais aussi commander des moteurs en bidirectionnel. J'utilise le double ampli op de puissance L272 ; il peut fournir un courant de sortie jusqu'à 1 A et se satisfait d'une tension d'alimentation de 4 V. Cette faible tension entraîne cependant une plage dynamique limitée.

La « liaison directe » entre capteurs et moteurs la plus simple est réalisée sous forme de convertisseur d'impédance (suiveur de tension), où le signal de sortie est réinjecté dans l'entrée inverseuse. Avec une alimentation symétrique, les moteurs sont, avec un seul ampli op chacun, commandés dans les deux sens, ces véhicules de Braitenberg font donc aussi de la marche arrière.

Un ampli op peut amplifier de façon linéaire, mais aussi inverser, additionner ou soustraire (en pondéré), calculer des logarithmes ou intégrer, travailler en comparateur de plusieurs signaux, voire servir de commutateur de seuil. Comme, sur les anciens ordinateurs analogiques, ces opérations arithmétiques étaient programmées via des panneaux de connexion, j'ai utilisé d'emblée des dispositifs d'interconnexion sous la forme de supports de circuits intégrés, pour câbler les amplificateurs opérationnels à la demande et adapter ainsi les fonctions et les paramètres des circuits de commande. Les premiers petits



Figure 1. Aspect extérieur et entrailles de ma première araignée-robot.

véhicules ne comportaient rien de plus que deux entraînements, une petite roue pivotante, les batteries et un support DIP, sur lequel trouvaient place un ampli op double et une paire de résistances ou de condensateurs interchangeables (**fig. 3**).

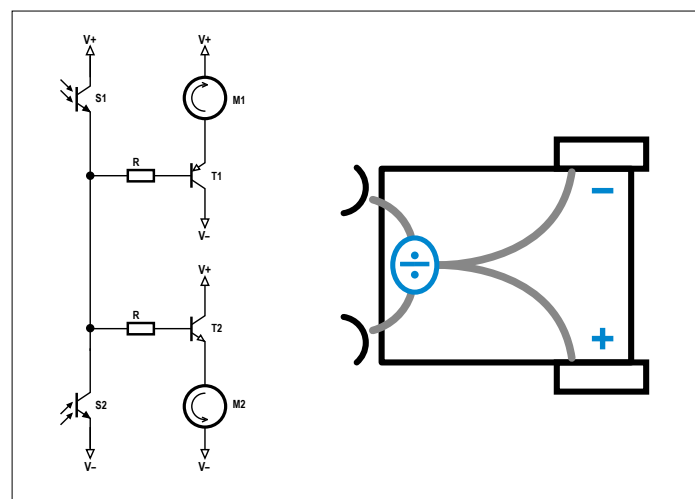


Figure 2. Simple circuit à transistors pour un robot à entraînement différentiel.

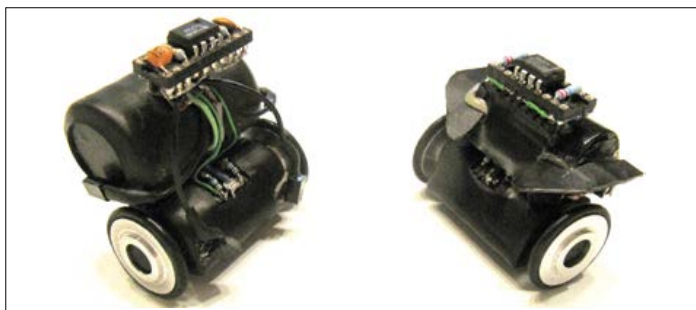


Figure 3. Les premiers modèles simples de robot analogique.

Le défi de production de série

Lorsqu'en 2013, j'ai décidé d'utiliser mon hobby pour devenir indépendant, de breveter l'idée et de créer une nouvelle génération de robots, je n'avais pas la moindre idée de ce qui m'attendait. L'un des défis majeurs fut de rendre la plate-forme d'expérimentation analogique la plus polyvalente possible tout en la gardant aussi simple que possible. Après nombre d'essais, je réussis à représenter de façon claire le système symétrique de connexion de la platine rouge (fig. 4, au-dessus), pour que chaque schéma de câblage corresponde avec le tracé de la couche supérieure, ce qui permet donc d'expérimenter même sans manuel. Une seconde carte sous-jacente, bleue (en des-

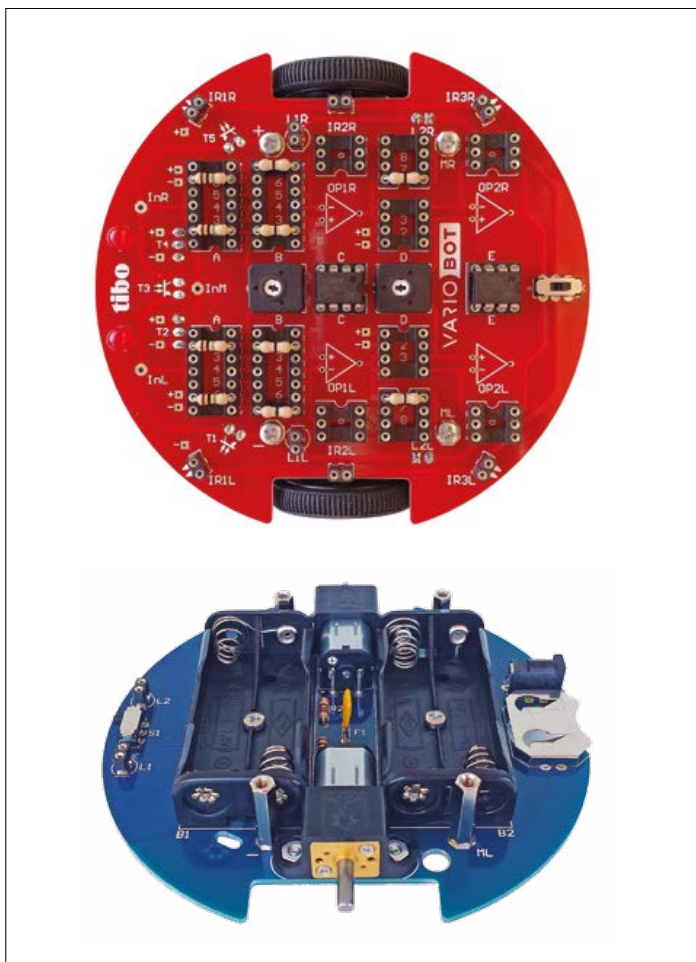


Figure 4. La carte de commande (en haut) et la carte de puissance (en bas) du robot tibo.

sous), reçoit l'alimentation et les entraînements du robot tibo. La mise au point d'un manuel de montage facile à comprendre et la définition d'un nouveau concept didactique pour le manuel d'expérimentation de tibo prirent du temps. Des graphiques illustrent les relations entre les signaux des capteurs et les grandeurs de sortie, ce qui permet de comprendre les fonctions élémentaires, sans formules. On voit en figure 5 un extrait du manuel d'expérimentation pour un suiveur de ligne.

Lors de la sélection des composants, je mis l'accent sur les aspects robustesse, fiabilité et confort de réalisation du kit. Il était hors de question d'utiliser des serre-câbles ou des fixations par collage tels que les utilisent souvent les kits bon marché. J'avais cependant bien sous-estimé la complexité de la sélection et de l'achat au meilleur prix de tous les composants auprès de divers revendeurs.

La recherche et l'achat d'un ampli op adéquat ont, à eux seuls, pris plusieurs semaines. Pour le kit à souder, comme il fallait absolument un circuit intégré en boîtier DIP, le choix s'est sévèrement restreint. L'ampli op devait non seulement piloter les moteurs, mais aussi commander les LED infrarouges. Pour ne pas le surcharger, il fallait limiter le courant. Comme tibo puise son énergie dans quatre cellules AAA (piles ou accus) avec une tension d'alimentation variable de 4 à 6 V (symétrique ± 2 à ± 3 V), la solution simple à base de résistance série n'entraînait pas en ligne de compte. Pour que le kit à souder reste le plus simple possible et parfaitement compréhensible, pas question

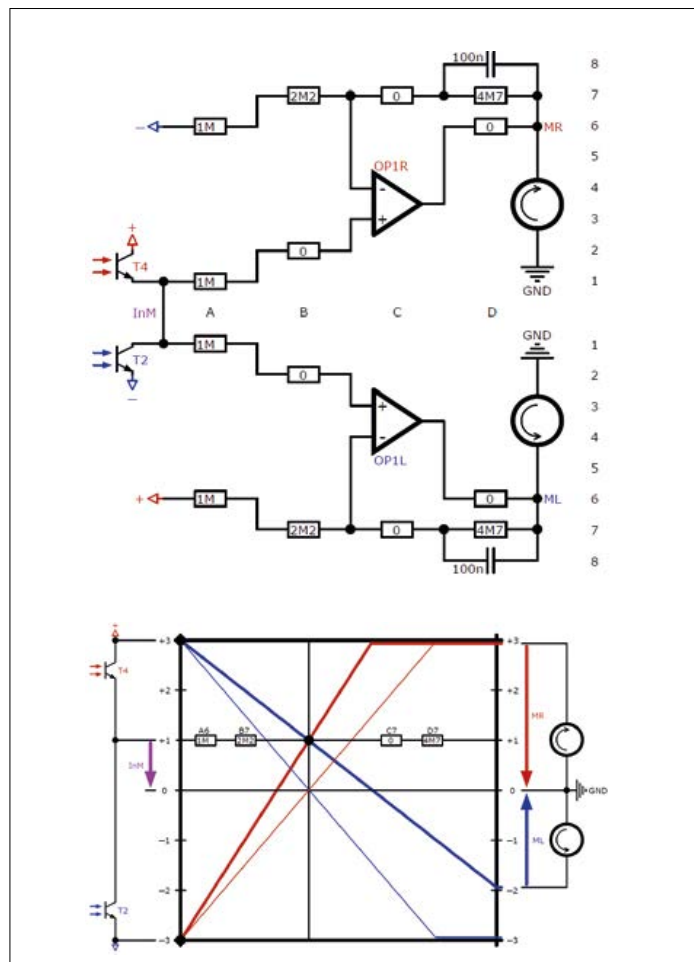


Figure 5. Extrait du manuel d'expérimentation.

d'utiliser des circuits intégrés additionnels tels qu'un régulateur de tension. Finalement, je choisis le double ampli op TS922, qui présente un énorme avantage : une limitation du courant interne. Avec cet ampli op, on peut attaquer directement les LED IR (IR908-7C-F), sans le surcharger. De plus, la limitation de courant rend également d'éminents services, moteurs en fonction. Si tibo devait se trouver coincé sur un obstacle sans qu'on s'en aperçoive ou si ses roues devaient bloquer, les moteurs ne courent pas le risque d'être endommagés.

À la recherche des composants !

Ce n'est qu'une fois le projet terminé et la commande d'une grande partie des amplis op passée que j'apprends que STMicroelectronics, le fabricant du TS922IN en boîtier DIP8, en avait arrêté la production. Mettre la main sur les 300 pièces manquantes pour la première série releva d'une odyssée sans fin. Les circuits intégrés commandés en Chine furent des contre-façons ne fournissant que 12 mA au lieu des 80 mA prévus. J'ai demandé trente devis avec désignation exacte du type et photos des composants et ai passé plus de dix commandes d'échantillons — le tout sans succès. Tous les échantillons présentaient le même défaut. Après de longues recherches, la société allemande TronicPool spécialisée dans l'achat de composants obsolètes réussit à me fournir la quantité requise. L'acquisition des motoréducteurs métalliques (PGM-12F) pris longtemps elle aussi. Il existe, pour ces moteurs très compacts de 25 x 12 x 10 mm³, une riche gamme de vitesses de rotation, de rapports de transmission et de puissances. Conçus pour une tension de 6 V, il en existe aussi des versions 3 V (comme dans tibo). La commande auprès d'un revendeur européen n'aboutit pas en raison des prix (trop) élevés. En quantités « domestiques », ces moteurs devraient coûter de l'ordre de 15 €. Sur eBay, Alibaba et Aliexpress, on trouve de nombreux revendeurs où ces moteurs coûtent moins cher. Attention, prudence conseillée. La réception d'un échantillon satisfaisant ne garantit en rien de recevoir la même qualité ensuite. Si l'on veut éviter des coûts de transport élevés, il faut donc prévoir un délai de livraison de plusieurs semaines. Pour le kit de robot, j'ai testé sept types de moteurs. Mes critères de décision furent une consommation d'énergie faible, une faible tension de démarrage et un fonctionnement souple. Certains des moteurs restèrent sur le carreau en raison d'un arbre « montant une mayonnaise » ou d'un non-démarrage pur et simple. Pour des applications personnelles et de petites quantités, je recommande donc d'acheter ces moteurs auprès des marchands européens pour éviter tout risque en termes de qualité et s'épargner des soucis en cas de réclamations.

Et encore des problèmes...

S'il vous venait à l'idée de développer un produit sur la base d'un projet personnel pour le commercialiser ensuite, sachez qu'il existe, parallèlement aux aspects techniques, d'autres défis, qui coûtent cher en temps et en argent. J'avais décidé au départ de breveter et la commande et la cinématique de l'araignée-robot. Il fut très instructif d'analyser des approches alternatives similaires et d'identifier le cœur de chaque invention. Si le droit de la propriété intellectuelle n'est évidemment pas une condition préalable pour la commercialisation d'un produit novateur, je conseille cependant de demander une recherche de brevets pour ne pas courir le risque de

violer les droits d'autrui et risquer un procès. Ceci vaut aussi pour un éventuel dépôt de la marque. Renseignez-vous auprès de l'Institut national de la propriété industrielle (INPI) [2]. Enfin, on se trouve inévitablement confronté aux obligations légales telles que la certification CE, la directive sur les emballages et la directive sur les déchets d'équipements électriques et électroniques (DEEE). En cas de doute, il est conseillé de chercher de l'aide auprès de professionnels, même si l'on pense au départ pouvoir tout faire soi-même. Pour me faire la main j'avais, par exemple, produit moi-même les cent premiers kits, mais j'ai rapidement réalisé qu'il est préférable de laisser ce genre de travaux que l'on ne fait pas bien ou que l'on n'aime pas faire, à des professionnels. Les kits suivants sont, depuis lors, vérifiés et emballés par un atelier protégé (pour handicapés) tout proche.

Pour de nombreux inventeurs, la meilleure approche consistera probablement à convaincre un fabricant établi de leur idée et de définir la coopération sous forme de licence. On évite ainsi d'avoir à assumer les risques que comporte toujours la création de sa propre entreprise. On pourra alors se lancer dans le développement de son prochain projet, sans avoir à se soucier de la distribution des produits qui de toute façon prend beaucoup de temps ! ◀

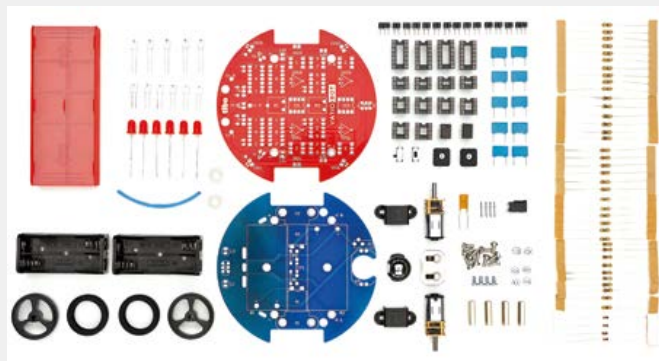
(140510 – version française : Guy Raedersdorf)

Liens

- [1] [Vehicles : Experiments in Synthetic Psychology, Braitenberg, Valentino, MIT Press 1986](#)
- [2] [INPI : www.inpi.fr](#)

tibo dans l'E-choppe d'Elektor

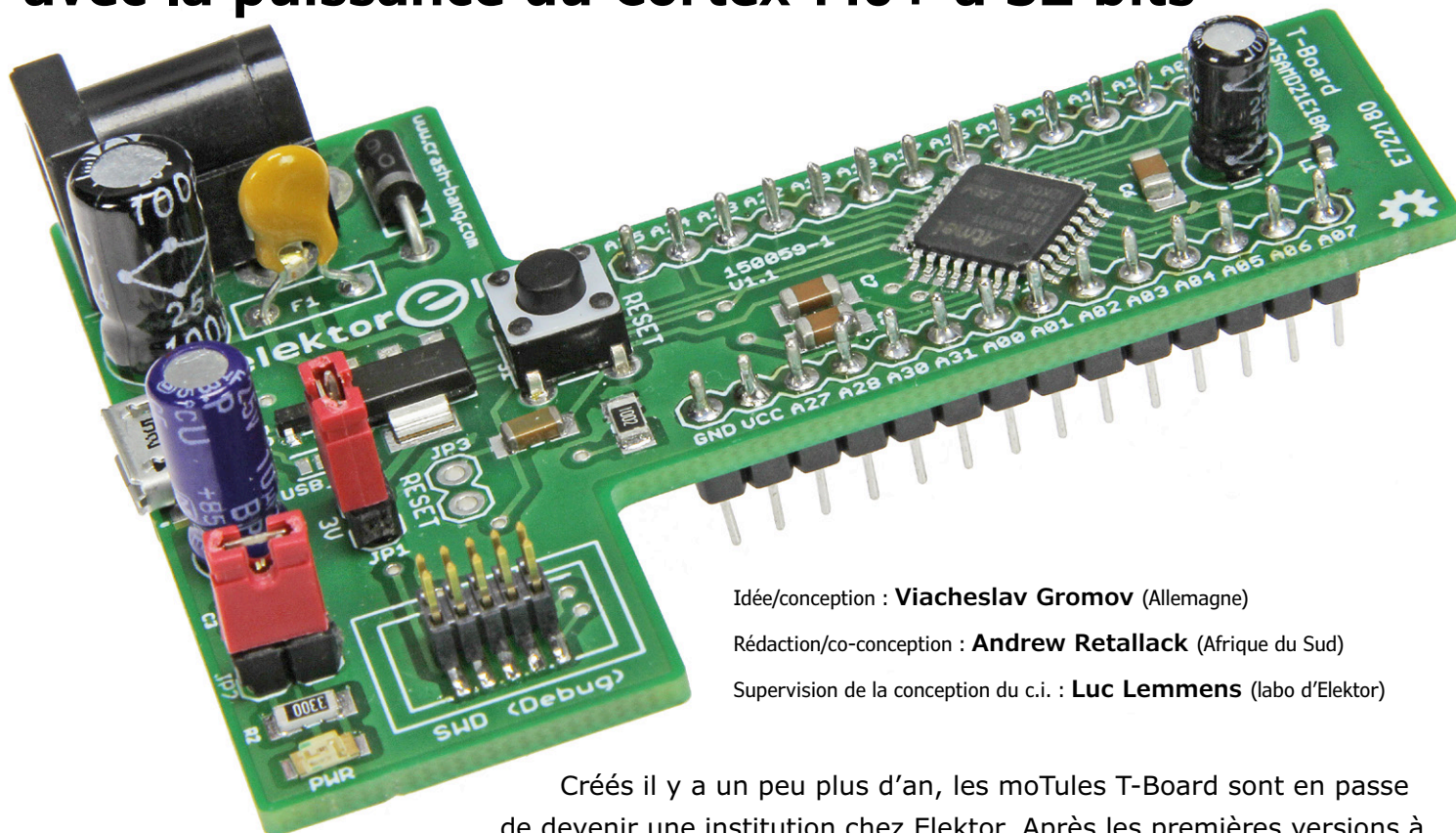
L'e-choppe d'Elektor est une excellente plate-forme pour promouvoir partout dans le monde de nouveaux produits comme le robot tibo. Le kit contient tous les composants électroniques et mécaniques nécessaires pour réaliser un robot tibo. Sur la page www.elektor.fr/tibo-robot-kit, vous trouverez de plus amples informations sur le kit mais aussi un manuel de construction et d'expérimentation de 40 pages (en anglais).



Le kit coûte 89,95 € ; les membres d'Elektor bénéficient d'une réduction de 10% !

T-Board bien ARMée

avec la puissance du Cortex-M0+ à 32 bits



Idée/conception : **Viacheslav Gromov** (Allemagne)

Rédaction/co-conception : **Andrew Retallack** (Afrique du Sud)

Supervision de la conception du c.i. : **Luc Lemmens** (labo d'Elektor)

Créés il y a un peu plus d'an, les moTules T-Board sont en passe de devenir une institution chez Elektor. Après les premières versions à AVR à 8 bits, la T-Board sans fil et la T-Board « ampli audio », voici la dernière (et la plus puissante) de la famille : la T-Board à ARM à 32 bits.

Bien qu'amateurs et *makers* utilisent les microcontrôleurs à 8 bits aussi naturellement que le menuisier utilise le bois, ils recourent de plus en plus souvent aux versions à 16 et 32 bits. L'architecture la plus en vogue est sans conteste celle de la famille Cortex-M d'ARM, des cœurs de processeur que la plupart des grands fabricants incluent dans leurs microcontrôleurs. Leur popularité vient à la fois de leur petit prix (un petit Cortex M0 coûte désormais moins de 1 €) et de ce qu'ils ont à offrir à l'heure de l'IdO : plus de puissance de calcul, de fonctions et de broches d'E/S. Avec le cours *µC ARM pour néophytes, pour passer de 8 à 32 bits*, Elektor favorise aussi leur adoption.

Le hic avec les puces ARM, c'est qu'elles sont en boîtier à pattes très rapprochées et nécessitent un cortège de composants. Pour le débutant, un montage de base

sur plaque d'essai peut donc très vite se corser ou virer à l'embrouillamini de fils et de composants. Il existe bien des cartes de développement pour ces puces, mais elles ne sont hélas pas faites pour les plaques d'essai, ce qui peut rendre leur utilisation difficile pendant la phase de prototypage et conduire à un montage « lourd ».

Aucun de ces inconvénients avec les T-Boards : elles s'enfichent directement dans une plaque d'essai et donnent accès à tous les signaux des µC. Si vous en avez déjà utilisé une, vous savez combien leur conception facilite et accélère le prototypage ! La T-Board présentée ici nous semble donc idéale pour exploiter un ARM Cortex M0+.

ARMement

Notre T-Board accueille le microcontrôleur ATSAMD21E18A d'Atmel, un ARM

Cortex M0+. Pour ceux qui le découvriraient, voici quelques-unes de ses caractéristiques :

- Temporisateur/compteur à 8/16/32 bits
- 3 temporisateurs/compteurs à 24 bits
- Horloge en temps réel (RTC) à 32 bits avec calendrier
- Interface USB 2.0 (fonctions hôte et périphérique)
- 4 interfaces sérielles (USART, I²C, SPI, LIN)
- CN/A à 12 bits avec 20 canaux (entrée différentielle et asymétrique ; support matériel du sur-échantillonnage)
- CN/A à 10 bits
- Interface I²S (*Inter IC Sound*)
- 2 comparateurs analogiques
- 16 interruptions externes
- 26 broches GPIO
- Contrôleur de périphériques tactiles

à 10 x 6 lignes

- Fréquence maximale de 48 MHz
- Alimentation de 1,62 V à 3,63 V.

C'est plus que n'en peut et offre le « vieux 8 bits ». Ajoutez un nombre de broches adapté à une « carte pour plaque d'essai », un honorable 256 Ko pour sa mémoire flash, 32 Ko de SRAM, un support USB, un prix acceptable, et vous comprendrez que nous ayons vu en l'ATSAMD21E18A le candidat idéal pour notre plateforme T-Board.

Après cette tirade justificative, attachons-nous sur la conception de la T-Board à ARM.

Conception

Le schéma de principe de notre T-Board à ARM est reproduit sur la **figure 1**. L'alimentation arrive par un jack standard de 2,1 mm avec positif au centre (K1), comme sur les cartes sœurs à AVR. Une diode 1N4007 (D1) protège la carte contre les inversions de polarité, et un fusible réarmable à CTP spécifié pour 0,5 A (F1) la protège des surintensités – étant donné que chaque broche V_{DD} absorbe au plus 92 mA, 0,5 A est plus que suffisant pour alimenter la carte et quelques périphériques. Les condensateurs C1, C2 et C3 lissent et filtrent le courant d'alimentation traversant IC2, un régulateur de 3,3 V à faible tension de déchet. Les ARM Cortex,

contrairement aux AVR, tolèrent jusqu'à 3,8 V, d'où notre décision d'opter pour une tension commune de 3,3 V. Aidée de R2, LED1 sert de témoin de mise sous tension. Les cavaliers JP1 et JP2 participent à la réduction de la consommation – nous y reviendrons.

Notez le connecteur micro USB K5. Les 5 V de cette connexion USB sont bien sûr acheminés vers le régulateur de tension pour y être filtrés et abaissés à 3,3 V. Nous avons ajouté un réseau de diodes (D2) afin de protéger les lignes D⁻ et D⁺ contre les risques de décharges électrostatiques, toujours possibles au moment où nous connectons des périphériques. D2 est plus précisément le PRTR5V0U2X de

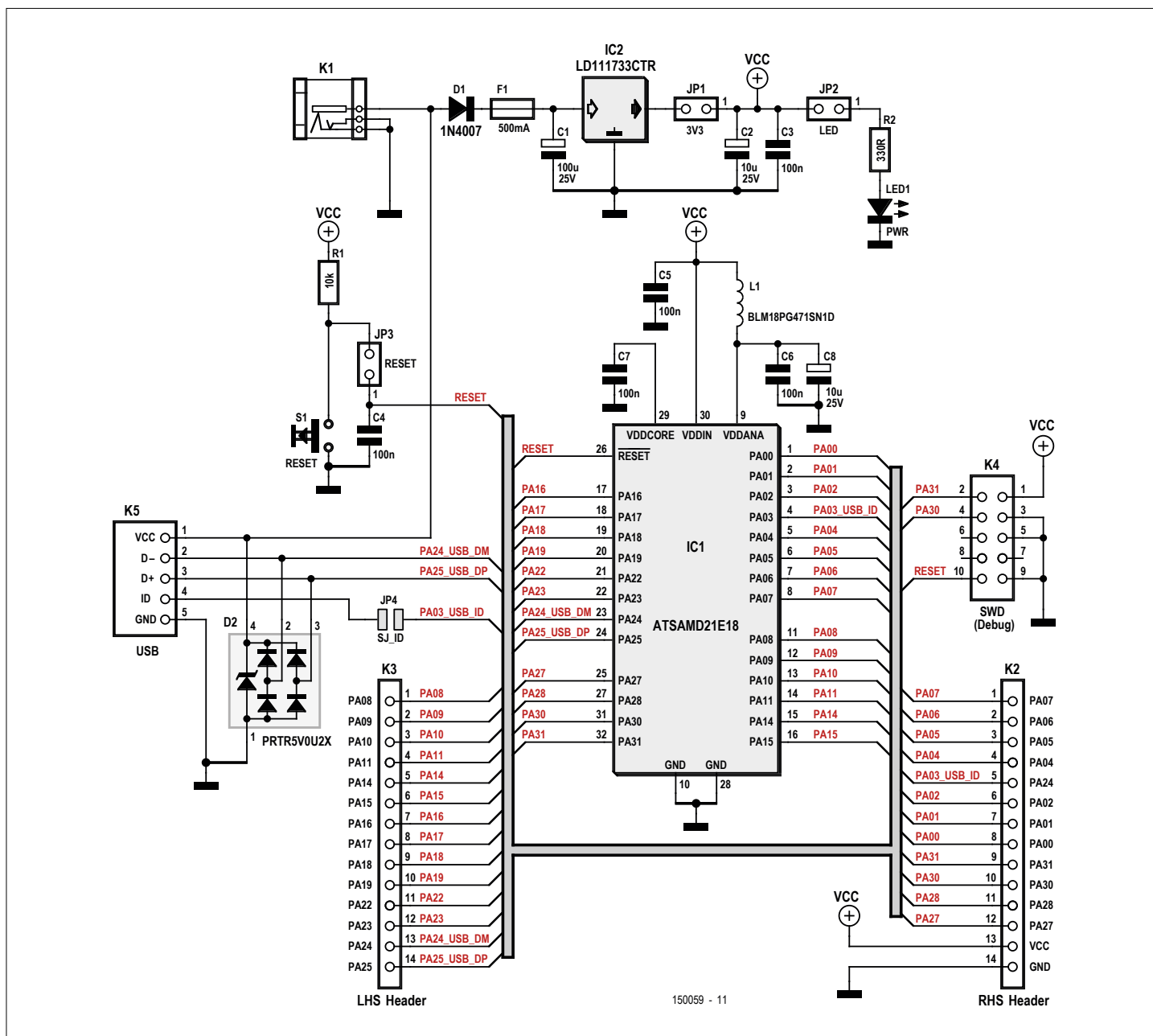
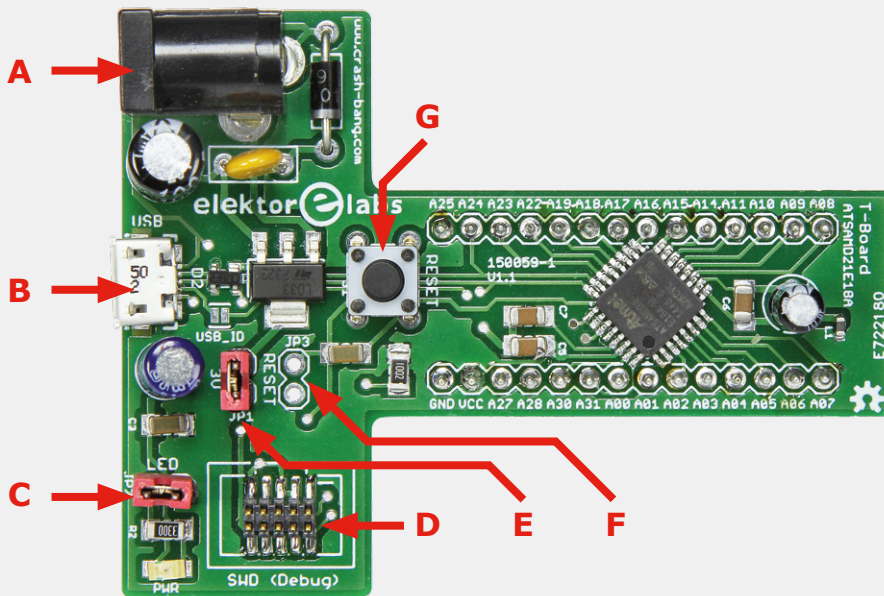


Figure 1. Schéma de la T-Board ARMée.

Disposition physique de la T-Board

- A. Connecteur d'alimentation : jack 2,1 mm avec positif au centre (9 V max.)
- B. Connecteur micro USB
- C. Cavalier : connecte/déconnecte la LED témoin de mise sous tension
- D. Connecteur SWD : embase de programmation à 10 broches (*Serial Wire Debug*)
- E. Mesure d'intensité : retirer le cavalier pour mesurer la consommation de la carte
- F. Embase Reset : permet une remise à zéro distante de la carte
- G. Bouton de remise à zéro



Cette T-Board est une plateforme idéale pour un microcontrôleur ARM Cortex M0+

NXP, un composant en mini-boîtier CMS conçu spécialement pour la protection des lignes de données à haut débit. Son nom officiel (pour ne pas dire son titre de majesté) est : *Ultra low capacitance double rail-to-rail ESD protection diode*. La broche ID de K5 n'est par défaut pas reliée au μC (pour avoir plus de broches disponibles), mais vous pouvez la connecter à la broche PA03 en soudant le cavalier JP4. La broche ID sert aux connexions USB On-The-Go (OTG) ; vous trouverez plus de détails sur ce point dans la note d'application d'Atmel [1].

La barrette à deux broches placée sur la ligne *Reset* permet une remise à zéro distante du microcontrôleur. La programmation et le débogage se font via l'embase SWD à 10 broches et au pas de 1,27 mm. L'interface SWD (*Serial Wire Debug*) est un sous-ensemble de l'interface JTAG qui n'utilise que six fils. S'il y a dix broches ici, c'est pour maintenir la compatibilité avec les programmeurs/débogueurs d'Atmel comme l'ICE.

Le reste de la carte est composé des embases d'accès aux broches d'E/S du μC et des composants auxiliaires : les condensateurs de découplage prescrits par la fiche technique, ainsi qu'une perle de ferrite (L1) pour éviter que le bruit V_{DD} n'interfère avec l'alimentation analogique VDDANA.

Nous n'avons placé aucun quartz sur la carte afin que vous puissiez sélectionner le quartz externe le mieux adapté à votre circuit : soit un oscillateur à quartz de 0,4 MHz à 32 MHz sur PA14/PA15, soit un quartz

Liste des composants

Résistances

CMS 1206, 1 %, 0,125 W
R1 = 10 k Ω
R2 = 330 Ω

Condensateurs

C1 = 100 μF , 25 V, 20 %, radial
C2, C8 = 10 μF , 25 V, 20 %, radial
C3, C4, C5, C6, C7 = 100 nF, 50 V, X7R, 1206

Inductance

L1 = BLM18PG471SN1D, perle de ferrite, 0,2 Ω / 1 A, 470 Ω @100 MHz

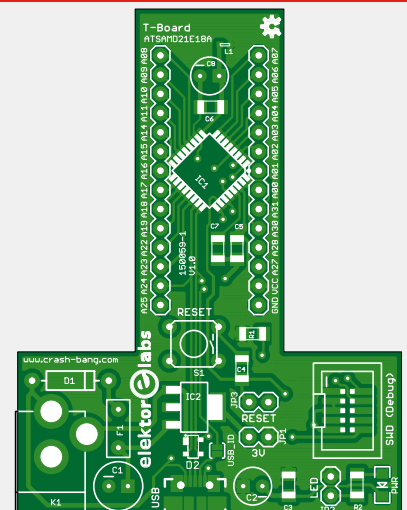
Semi-conducteurs

D1 = 1N4007 (1000 V, 1 A)
LED1 = LED, rouge, CMS 1206
IC1 = ATSAM21E18A-AU
IC2 = LD1117S33CTR (régulateur de tension 3,3 V)
IC3 = PRTR5V0U2X (diode de protection contre les pointes de tension)

Divers

S1 = poussoir tactile, 24 V, 50 mA, 6x6 mm
JP1, JP2, JP3 = cavalier, 1x2, vertical, pas de 2,54 mm
F1 = fusible réamorçable à CTP, 500 mA
K1 = jack d'alimentation CC 2,1 mm
K2, K3 = barrette mâle 14 voies, SIL, pas de 2,54 mm
K4 = barrette mâle 10 voies (2x5), pas de 1,27 mm
K5 = fiche micro-USB 2.0, type B, CMS embasse cavalier au pas de 2,54 mm circuit imprimé nu réf. 150059-1 (www.elektor.fr)
carte assemblée réf. 150059-91 (www.elektor.fr)

Figure 2. Sérigraphie de la T-Board ARMée. Pas de panique, la carte assemblée est disponible sur www.elektor.fr.



de montre de 32,768 Hz sur PA00/PA01 si vous utilisez une horloge RTC.

Optimiser la consommation

Dans son article *T-Board 28 : consommation minimale* (décembre 2014), l'auteur avait montré comment optimiser la consommation d'une T-Board 28 équipée d'un ATmega328p en y apportant de petites modifications et en mesurant les effets produits. C'est en pensant à ce genre de flexibilité et à la possibilité d'agir sur plusieurs leviers que nous avons conçu la T-Board, et il en a bien sûr été de même avec notre T-Board à ARM. Le cavalier JP1 permet de brancher un circuit détecteur de courant afin de mesurer la consommation de la carte, donc d'évaluer les effets apportés par une modification du code ou par une optimisation matérielle. Les premières T-Boards à AVR le permettaient déjà, mais le nouveau cavalier JP2 permet en outre de déconnecter la LED1 afin que sa consommation ne fausse pas les mesures.

Conception mécanique

Si vous n'avez pas encore utilisé de T-Board, la barre transversale du « T » sert à placer le plus de composants possible hors de la plaque d'essai, et le « pied » vertical à orienter les broches du contrôleur de façon à ce qu'elles s'enfichent directement dans la carte et soient ainsi facilement accessibles.

La T-Board est bien sûr à source ouverte puisqu'elle a été conçue dans l'esprit de partage d'Elektor. Fichiers de conception [3], liste des composants et dessin du circuit imprimé (**fig. 2**) sont à votre disposition si vous avez les ressources et le matériel nécessaires à l'assemblage de votre propre moTule. Si la perspective de vous attaquer aux pattes rapprochées (0,8 mm) du boîtier TQFP de l'ARM ou aux broches invisibles du connecteur USB vous refroidit, l'e-shoppe vous propose la T-Board assemblée et testée, bref prête à l'emploi.

Prêt et ARMé ?

Après ce déballage de présentations et d'explications, il est temps de passer à l'action ! Nous avons choisi un projet simple pour illustrer l'utilisation de la T-Board. Si vous souhaitez vous frotter à plus complexe, allez voir du côté de la série *µC ARM pour néophytes* qui a débuté dans le numéro de janvier & février 2015. Une petite mise en garde toutefois avant

de démarrer : les microcontrôleurs ARM sont moins simples à mettre en œuvre que les AVR, autrement dit il faut consacrer quelque temps à l'étude de leurs systèmes et architectures avant de maîtriser leur complexité. L'objectif ici n'étant pas de déshabiller les ARM jusqu'au sili-cium mais juste de vous faire découvrir la T-Board, référez-vous là encore au cours *µC ARM pour néophytes* pour une tonique initiation à ces puces. Viacheslav Gromov, qui a lancé ce projet, prépare par ailleurs un article consacré à la programmation de cette T-Board.

Étape 1 : câblage

Commencez par positionner la T-Board sur une plaque d'essai, comme sur la **figure 3** : barre horizontale du « T » hors de la plaque et contre son bord, et broches latérales du « pied » chevauchant le canal central de la plaque. Ensuite :

1. câblez un cavalier entre la broche GND et le rail négatif de la plaque.
2. Reliez l'anode d'une LED à la broche A07 et placez sa cathode sur une rangée libre de la plaque.
3. Ajoutez une résistance en plaçant une

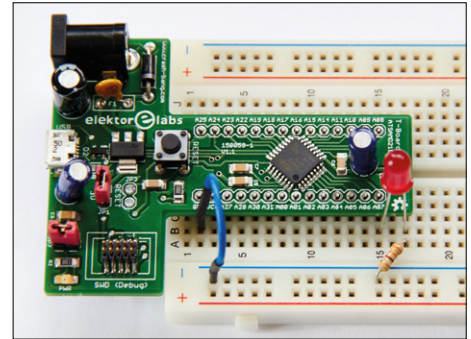


Figure 3. Un montage pour le moins dépouillé grâce à la T-Board.

de ses pattes sur la même rangée que la cathode, et insérez l'autre patte dans le rail négatif.

La figure 3 montre le projet assemblé : admettez qu'il est impressionnant de n'y voir qu'un seul cavalier !

Étape 2 : création du projet

Nous allons œuvrer depuis Atmel Studio 6.2 (voir l'**encadré D'autres EDI**) et utiliser les bibliothèques ASF (*Atmel Software Framework*). Le framework ASF est à nos yeux le moyen le plus rapide de lancer un

D'autres EDI

Les ARM font en quelque sorte partie du haut du panier des µC et nécessitent un environnement de développement intégré approprié. Ces programmes sont complexes et la place nous manque pour vous proposer un comparatif complet de tout ce qui existe. Nous nous contenterons donc d'un bref aperçu des trois EDI les plus populaires.

Atmel Studio 6.2

Premier de la liste car à nos yeux le mieux adapté au travail avec les ARM dans un cadre non-professionnel. Il est gratuit, non bridé, et prend en charge non seulement les µC d'Atmel mais aussi leurs programmeurs et débogueurs, puisque c'est bien sûr Atmel qui a conçu Atmel Studio. www.atmel.com/atmelstudio

IAR Embedded Workbench pour ARM

IAR propose plusieurs EDI de niveaux (et prix !) professionnels. Tous prennent en charge les µC AVR et ARM d'Atmel ainsi que ceux d'autres fabricants. Ils représentent un bon choix si vous souhaitez exploiter des µC de fabricants différents avec un même outil. Seul frein : leurs prix. Les versions gratuites ont une taille de code si limitée qu'elles ne sont vraiment intéressantes que pour évaluer le produit. www.iar.com/iar-embedded-workbench/arm

Keil

L'EDI Keil MDK-ARM fait partie des outils conçus par ARM pour la conception de projets à base d'ARM. Bien que cet EDI cible avant tout le marché professionnel, Keil offre une version appelée *MDK-lite* pour un code limité à 32 Ko, valeur qui devrait suffire à la plupart des projets amateurs. Si votre emploi du temps vous le permet et si vous souhaitez explorer les µC ARM d'autres fabricants, cet EDI peut être intéressant. www.keil.com

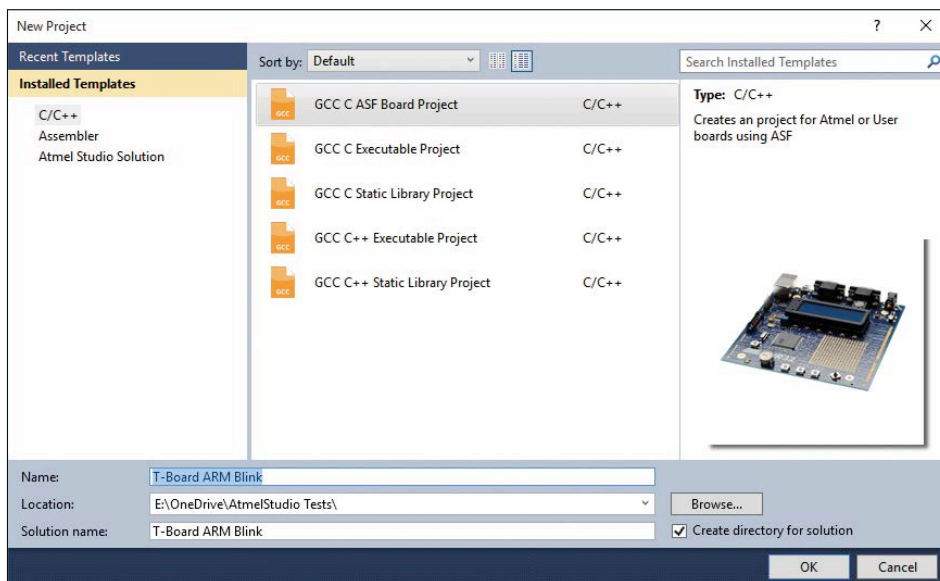
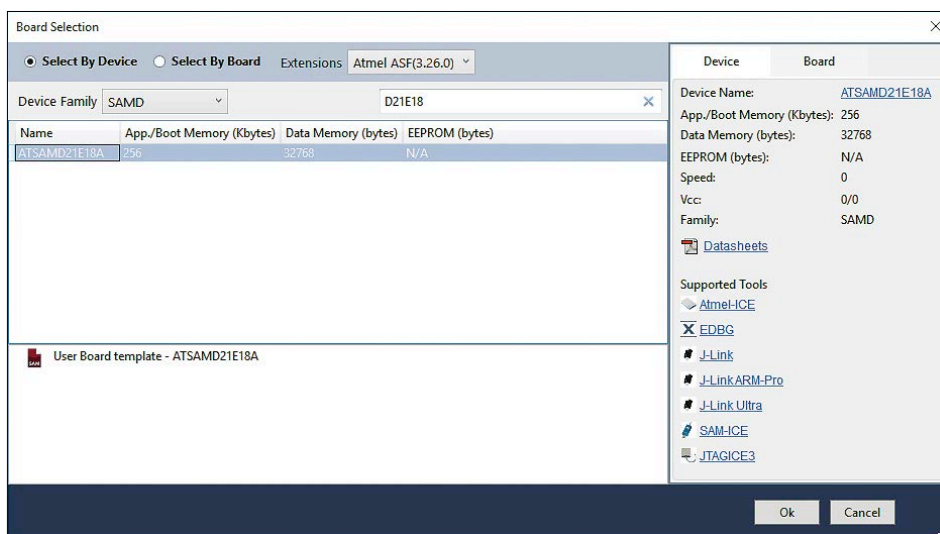
Figure 4. Fenêtre de sélection du *GCC ASF Board Project*.

Figure 5. Sélection du µC de la T-Board.

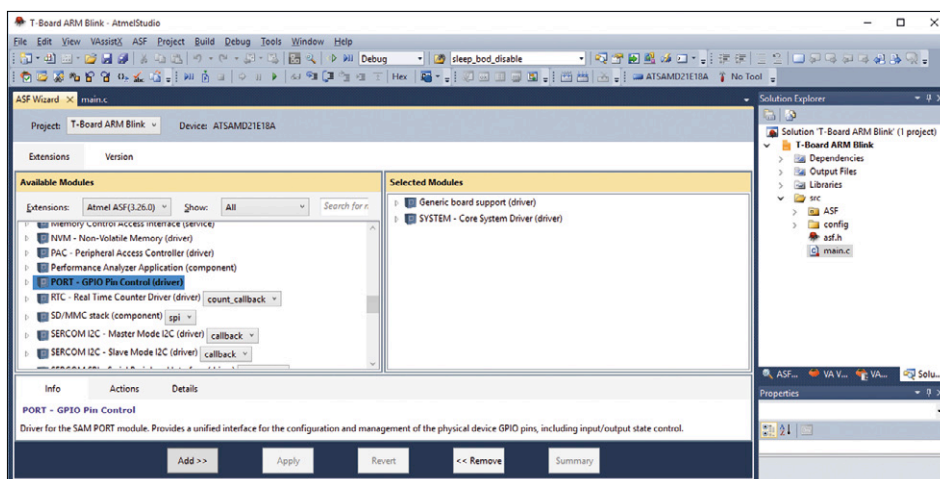


Figure 6. Ajout du pilote PORT.

projet au vu de la complexité d'un contrôleur ARM. La série μC ARM pour néophytes l'utilise également, mais avec une carte Atmel SAMD20 XPlained Pro ; bien que le paramétrage du projet soit légèrement différent ici, ASF nous permettra une nouvelle fois d'aller sans détour à l'essentiel.

Opérez comme suit :

1. Ouvrez Atmel Studio et créez un nouveau projet *GCC ASF Board Project* – entrez un nom et l'emplacement du dossier de sauvegarde (**fig. 4**).
2. Sélectionnez le μC de la carte, ici un ATSAM21E18A. Comme il n'existe que peu de cartes de développement pour ce μC , il faut sélectionner *User Board Template*. Cliquez sur OK (**fig. 5**).
3. L'EDI a créé un projet vide dont le fichier *main.c* est visible dans le dossier *src* de l'explorateur.
4. Ajoutons au projet le pilote ASF *PORT* ; il fournit un ensemble de fonctions qui facilitent l'accès aux broches. Lancez l'assistant *ASF Wizard* depuis le menu ASF.
5. La fenêtre qui s'ouvre (**fig. 6**) affiche à gauche les modules disponibles, à droite les modules sélectionnés. Faites défiler les modules disponibles jusqu'à *PORT – GPIO Pin Control (driver)*, sélectionnez-le, puis cliquez sur *Add* et *Apply*. Le pilote *PORT* fait maintenant partie du projet.
6. Copiez le code du **listage 1** dans le module *main.c*.

Étape 3 :

programmation de la T-Board

Compilons et chargeons le projet dans la T-Board :

1. Compilez le projet (F7) ; il ne doit bien sûr y avoir aucune erreur.
2. Alimentez la carte avec le jack CC (en supposant que votre programmeur n'alimente pas le projet). L'ICE d'Atmel que j'utilise fait partie des programmeurs qui n'alimentent pas la cible. Si vous utilisez un JTAGICE3, référez-vous à sa documentation pour son utilisation.
3. Reliez le programmeur à l'embase SWD de la carte et l'alimentation au port USB du PC.
4. Sélectionnez le programmeur depuis le menu *Tools*, sous *Device Programming*.
5. Chargez le programme : choisissez *Start without Debugging* depuis le menu *Debug*.

Liens

- [1] Atmel USB 2 OTG : www.atmel.com/Images/Atmel_11201_USB-OTG-Like-Connector-Implementation_SAM9G-SAM9X-SAMA5D3_Application-Note.pdf
- [2] Consommation minimale des T-Boards : www.elektormagazine.fr/140413
- [3] Ressources pour le projet : www.elektormagazine.fr/150059
- [4] Atmel Studio : www.atmel.com/tools/atmelstudio.aspx
- [5] www.der-hammer.info/terminal/

6. Admirez la LED qui clignote !

Aller plus loin

Maintenant que vous avez vu la T-Board à l'œuvre, vous avez probablement hâte de vous attaquer à d'autres projets. Mais quels projets ? Eh bien si vous débutez avec les ARM Cortex, voici un défi intéressant : reprendre les projets de la série μC

ARM pour néophytes en remplaçant la carte Xplained Pro par la T-Board. Vous pouvez réduire le nombre de composants utilisés pour simplifier le câblage et le débogage. V. Gromov, co-auteur de ce projet, a également écrit un programme (**encadré Premier projet** et **listage A**) qui vous familiarisera un peu plus avec cette T-Board ARMée.

Nous espérons que vous apprécierez ses atouts : un formidable outil d'apprentissage et un moyen rapide de mettre sur pied vos projets ! ◀

(150059 – version française : Hervé Moreau)

Listage 1. Faites coucou à la LED qui clignote

```
#include <asf.h>

#define LED_PIN PIN_PA07 //LED reliée à PA07

//prototypes de fonction
void configure_port_pins(void);

int main (void)
{
    system_init(); //routine pour initialiser le système, les horloges, etc.

    configure_port_pins(); //configuration des broches GPIO

    SysTick_Config(system_gclk_gen_get_hz(GCLK_GENERATOR_0)); //active l'interruption SysTick :
                                                                //lit la fréquence pour définir l'interruption sur 1 s

    while (1)
    {
        //besoin d'aucun code ici puisque nous utilisons une interruption pour changer le niveau de la broche de la LED
    }
}

//fonction de configuration des broches de port
void configure_port_pins(void)
{
    struct port_config config_port_pin; //structure utilisée pour stocker les paramètres
    port_get_config_defaults(&config_port_pin); //lit la configuration actuelle des broches dans la structure
    config_port_pin.direction = PORT_PIN_DIR_OUTPUT; //paramètre la structure pour indiquer que la broche est une sortie
    port_pin_set_config(LED_PIN, &config_port_pin); //utilise la structure pour définir la direction de la broche de la LED
}

//gestionnaire d'interruption de l'interruption SysTick
void SysTick_Handler(void)
{
    port_pin_toggle_output_level(LED_PIN); //change le niveau de la broche de la LED
}
```

Premier Projet

Viacheslav Gromov

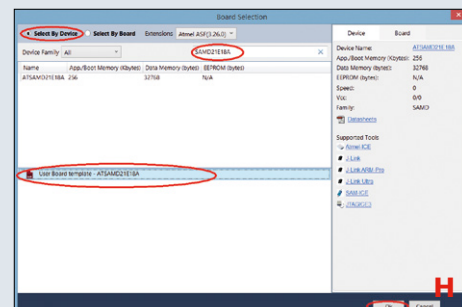
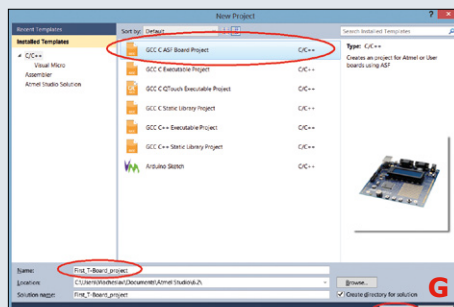
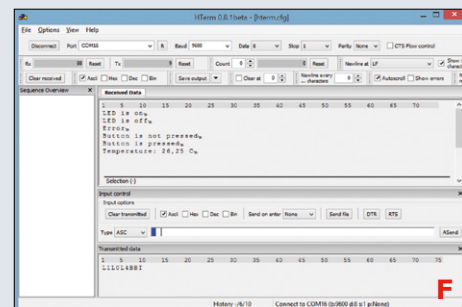
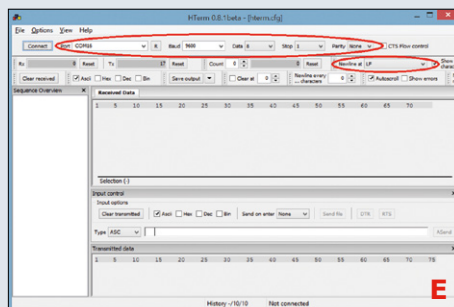
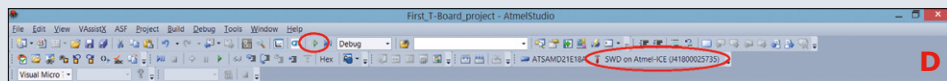
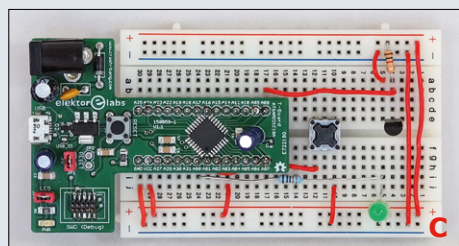
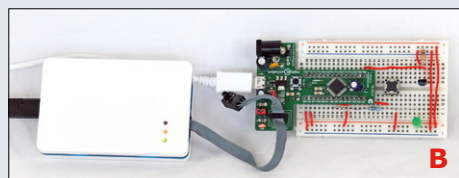
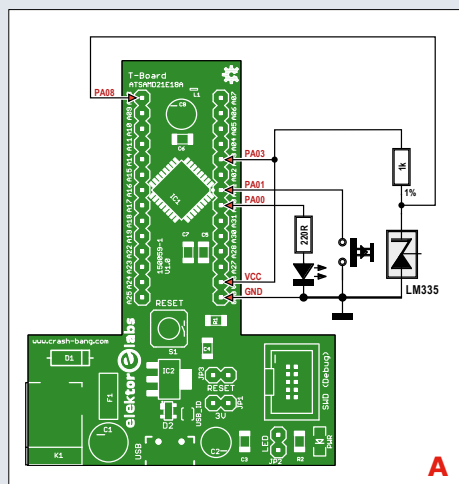
Le projet est volontairement simple : mise sous et hors tension d'une LED, scrutation d'un poussoir, et mesure de température avec un LM335 et un CAN. Le circuit est commandé via USB depuis un émulateur de terminal.

1. Matériel

La **figure A** montre comment relier les composants à la T-Board, la **figure B** montre le montage terminé. Une fois tout en place, utilisez un câble USB A – USB micro-B pour relier la T-Board à votre PC ; vous devriez en avoir un sous la main, ce type de câble sert souvent à recharger les smartphones ou les tablettes. Il vous faut aussi un débogueur comme l'ICE d'Atmel. Reliez-le au connecteur SWD (*Debug*) de la T-Board, et bien sûr à votre PC (**fig. C**).

2. Logiciel

Commencez par télécharger depuis [3] le projet Atmel Studio appelé *First_T-*



Board_project. Ouvrez-le depuis Atmel Studio 6 ; si vous ne l'avez pas, vous pouvez le télécharger depuis [4] en tant qu'invité ou en créant un compte. Si vous découvrez aujourd'hui cet EDI et la famille SAM D d'Atmel, étudiez notre cours *µC ARM pour néophytes*. Le code débute par la déclaration des variables, tableaux, prototypes de fonction et fonctions de configuration pour les périphériques. Dans la fonction *main* sont initialisés le système, les interruptions et les fonctions de temporisation. Ensuite toutes les fonctions de configuration sont appelées. Le **liste A** montre que le corps de la boucle sans fin est constitué d'une seule instruction switch. Elle évalue l'expression `udi_cdc_getc()`, autrement dit le caractère ASCII reçu, puis réalise le branchement conditionnel correspondant. Le **tableau A** indique les commandes

(caractères) qu'il est possible d'envoyer depuis le terminal. Chaque caractère est comparé dans un case du switch en indiquant le code ASCII correspondant : 76 pour « L », 66 pour « B », et 84 pour « T » (notez qu'il s'agit de lettres majuscules).

Le premier case (76) contient une instruction `if-else` qui allume la LED lorsque la variable `number` représentant le second caractère contient « 1 » (commande L1), et l'éteint lorsque `number` vaut 0 (commande L0). Le contenu de `number` est récupéré avec `number = udi_cdc_getc()`. Selon celui-ci, la fonction `udi_cdc_write_buf(&buffer, x)` envoie via USB le message *LED is on* ou *LED is off* ; ses arguments sont normalement un pointeur (&buffer) vers la chaîne contenant la donnée à envoyer, et le nombre de caractères à transmettre (x). Toutefois,

Tableau A. Mini jeu de commandes

L	1	L'envoi de L1 (L pour LED) allume la LED ; la carte envoie le message <i>LED is on</i> . L'envoi de L0 éteint la LED ; la carte envoie <i>LED is off</i> . Si L est suivi d'un chiffre différent de 0 ou 1, la carte retourne le message <i>Error</i> .
	0	
B		À l'envoi de B (B pour Bouton), la T-Board répond <i>Button is pressed</i> ou <i>Button is not pressed</i> .
T		Le caractère T (T pour Température) renvoie la chaîne « <i>Temperature: x C</i> », où x est la température actuellement mesurée.

par souci de simplicité, nous n'employons pas de pointeur ici mais passons directement la chaîne : `udi_cdc_write_buf(«LED is on\n », 10);`. « \n » est le caractère de nouvelle ligne. Le message *Error* est envoyé au PC si number est différent de 1 ou 0.

Le case suivant (66) scrute l'état de la broche PA01 du poussoir à l'aide de `port_pin_get_input_level(PIN_PA01)`, et en fonction de cet état renvoie *Button is pressed* ou *Button is not pressed*. Le dernier case (84) mesure la tension envoyée par le LM335 sur le canal 16 du CAN (PA08), et calcule la température à partir de cette valeur. La chaîne « *Temperature: x C* » envoyée au PC pour indiquer la température est formatée selon une approche semblable à celle utilisée dans la cinquième partie du cours ARM.

3. Test

Sélectionnez votre débogueur (au-dessus à droite), et lancez la compilation et le transfert en cliquant sur le bouton vert *Start Without Debugging* (fig. D). Une fois le µC programmé, la T-Board sera enregistrée sur votre ordinateur comme un *Communication Device Class ASF example* – une interface série virtuelle. L'installation des bons pilotes peut demander un certain temps. Cette étape franchie, ouvrez le gestionnaire de périphériques du PC et cherchez sous *Ports (COM & LPT)* le numéro de port assigné à votre T-Board. Ouvrez un émulateur de terminal, p. ex. HTerm [5], et entrez les paramètres suivants (fig. E) :

- numéro de port COM (d'après votre gestionnaire de périphériques)
- 9600 bauds
- 8 bits de données
- 1 bit d'arrêt
- pas de parité
- paramètre *Newline* sur LF

Avec *DTR on*, connectez-vous à la carte et testez les commandes du tableau A (fig. F). Ça marche ? Bravo, vous venez de réussir votre premier projet avec la carte T-Board à ARM !

Cela dit je vous tenais la main. Jetez-vous à l'eau en créant seul un nouveau projet !

4. Créer un nouveau projet avec la T-Board ARMée & Atmel Studio

La création d'un nouveau projet est

expliquée dans la première partie du cours ARM. Avec la T-Board, vous devez sélectionner un projet *GCC C ASF Board Project* depuis *File/New/Project...* (fig. G), valider avec OK, puis utiliser *Select By Device* pour sélectionner

l'ATSAMD21E18A (fig. H). Validez avec OK pour qu'Atmel Studio crée votre projet C. Voilà, vous pouvez importer des bibliothèques ASF depuis l'assistant ASF et écrire votre code. À vous de jouer !

Listage A. Corps de la boucle sans fin du programme

```
while(1){
switch (udi_cdc_getc())
{
    case 76: //if received "L"
        number = udi_cdc_getc(); //get the second character (a number)
        if (number == 49) //if received a one
        {
            port_pin_set_output_level(PIN_PA00, 1); //put the LED on
            //send "LED is on" with a "new line" on USB
            udi_cdc_write_buf("LED is on\n", 10);
        }
    else if(number == 48) //if received a zero
    {
        port_pin_set_output_level(PIN_PA00, 0); //put the LED off
        //send "LED is off" with a "new line" on USB
        udi_cdc_write_buf("LED is off\n", 11);
    }
    else //if received a wrong number (not 1 or 0)
    {
        udi_cdc_write_buf("Error\n", 6); //send "Error" and a "new line" on USB
    }
    break;
    case 66: //if received "B"
    if(!port_pin_get_input_level(PIN_PA01)) //get the level on PA01 (button)
    {
        //if pressed, send "Button is pressed" and a "new line" on USB
        udi_cdc_write_buf("Button is pressed\n", 18);
    }
    else
    {
        //if not pressed, send "Button is not pressed" and a "new line" on USB
        udi_cdc_write_buf("Button is not pressed\n", 22);
    }
    break;
    case 84: //if received "T"
    adc_start_conversion(&adc_instance); //start ADC-conversion
    //read and save the conversion result
    while(adc_read(&adc_instance, &data) == STATUS_BUSY){}
    //calculate the temperature
    temperature = (25 + (data * 0.000805 - 2.945) / 0.01) * 100;
    //reformat the result in a buffer
    sprintf(temperature_string, "%i", temperature);
    //send "Temperature:" with USB
    udi_cdc_write_buf("Temperature: ", 13);
    //send the temperature on USB
    for(i = 0; i < 4; i++){
        if((temperature >= 1000) && (i == 2)) udi_cdc_putc(44); //make a comma
        if((temperature < 1000) && (i == 1)) udi_cdc_putc(44); //make a comma
        udi_cdc_putc(temperature_string[i]); //send a digit on USB
        //send "C" and "new line" on USB at the end of transmission
        if(i == 3) udi_cdc_write_buf(" C\n", 3);
    }
    break;
}
}
```


hors-circuits

ampli de classe

A, B, C, D, E, F, G, H : quesako ?

Robert Lacoste (Chaville)

Dans mon précédent article, je vous ai expliqué comment réaliser un amplificateur avec un banal transistor bipolaire [1]. Même si je ne l'avais qu'évoqué, cet amplificateur est dit de classe A. Si vous êtes un audiophile aguerri alors pour vous « classe A » rime avec « qualité supérieure ». Vous savez également qu'il y a une ribambelle d'autres variantes ; A, AB, B et D sont les plus connues. Toutefois on trouve aussi des amplificateurs de classe G, H, voire T, et même des choses plus exotiques pour les aficionados des radiofréquences : C, E, F... La liste n'en finit pas. Un peu effrayant non ? Qu'est-ce que tout cela veut dire ? Dans cet article, je lève le voile sur ces différentes classes avec des mots simples.

Classe A !

Commençons par quelques hypothèses simplificatrices. Supposons que vous deviez concevoir un amplificateur dont les entrées et sorties sont couplées en AC, et que son alimentation puisse être soit unipolaire (+V), soit bipolaire ($\pm V$). Supposons aussi qu'une éventuelle inversion de la phase du signal ne soit pas un problème, et que vous vous intéressiez surtout à obtenir un gain en tension, c'est-à-dire un signal AC en sortie avec une amplitude supérieure à celle en entrée. Je vous l'accorde, ceci est assez réducteur ; il faudrait prendre en compte d'autres facteurs (impédances d'entrée et de sortie, d'où gain en courant et en puissance, etc.), mais cela suffira pour mon propos.

Quelles sont les solutions possibles ? La première, la plus simple, serait d'utiliser un seul transistor comme présenté dans mon précédent article. La **figure 1** vous rappelle la structure d'un tel ampli. Fondamentalement le transistor doit être polarisé pour avoir au repos une tension de collecteur proche de $V_{cc}/2$. Réfléchissez, c'est la seule solution pour que le signal de sortie puisse osciller au-dessus ou en dessous de cette tension constante, en fonction de la polarité de la tension d'entrée. Si vous regardez à nouveau le schéma, vous comprendrez qu'alors un courant doit circuler en permanence à travers le transistor. Vous pouvez remplacer le transistor bipolaire par un MOSFET ou une paire de transistors, le concept reste le même : une telle structure d'amplificateur ne peut fonctionner correctement, c'est-à-dire avec une distorsion faible, que si le courant à travers le ou les transistors de sortie ne s'annule jamais. C'est exactement la définition d'un amplificateur de classe A. Ses avantages ? Simplicité, pas besoin d'une alimentation bipolaire et excellente linéarité (au moins tant que la tension de sortie ne s'approche pas trop des rails d'alimentation). Un ampli de classe A présente toutefois un inconvénient majeur : comme un courant continu doit circuler à travers le transistor, son efficacité énergétique est très faible. Elle ne dépasse pas 30%, voire 40% en remplaçant la résistance de collecteur par

un transformateur. Supposons que vous souhaitiez construire un ampli stéréo de classe A de 2x100 W. Faites le calcul, vous verrez que 300 W seraient dissipés en continu dans les transistors, qu'il y ait ou pas un signal audio à amplifier. Un excellent chauffage d'appoint !

Classe B

Comment améliorer l'efficacité de l'amplificateur ? Il faut bien sûr éviter autant que possible ce courant permanent à travers les transistors de sortie. Ce qui nous amène au concept d'amplificateur de classe B, illustré à la **figure 2**. Un tel amplificateur utilise une paire de transistors en sortie (en général NPN et PNP), en configuration *push-pull*. Ces transistors sont polarisés de telle sorte que l'un d'eux soit conducteur lorsque le signal d'entrée est positif, l'autre lorsqu'il est négatif. Comme les deux transistors ne sont jamais conducteurs simultanément, il y a très peu de pertes : toute l'énergie consommée est transmise à la sortie. Beaucoup de variantes existent, mais retenez que classe B signifie toujours que chaque transistor est conducteur la moitié du temps. Évidemment la grande qualité d'un amplificateur de classe B est son rendement, en théorie proche de 100% (en pratique de l'ordre de 70% avec des transistors bipolaires, et près de 90% avec des MOSFET). Comme rien n'est gratuit, vous imaginez bien qu'un tel ampli doit présenter des inconvénients. Ici le problème est qu'il est impossible de garantir que l'un des transistors commence à conduire exactement lorsque le second se bloque. Il y a toujours un peu d'incertitude, en particulier liée à la dispersion des caractéristiques des composants, à leur échauffement, à la fréquence du signal d'entrée, etc. De là un amplificateur de classe B induit une distorsion dite de croisement à chaque passage par zéro du signal. Faible s'il est bien conçu, mais toujours présente.

Et la classe AB ?

Comme son nom l'indique, la classe AB est une sorte d'intermédiaire entre les classes A et B. Prenez le schéma d'un ampli

de classe B (figure 2), et modifiez un peu la polarisation des transistors pour qu'un petit courant permanent les traverse. Dans une telle approche, un seul transistor est passant lorsque le signal d'entrée a une tension assez élevée (positive ou négative), mais les deux conduisent un peu lorsque celle-ci est voisine de 0 V. C'est pourquoi les caractéristiques d'un ampli de classe AB sont intermédiaires entre les classes A et B, tant en linéarité qu'en efficacité. Ce « juste milieu » fait de la classe AB un choix judicieux pour la majorité des applications à faible coût : la plupart des amplis audio étaient des amplis AB, au moins jusqu'à la prolifération de la classe D que nous verrons dans quelques minutes.

Un petit tour en classe C

Plutôt que de continuer à parler d'audio avec la classe D, suivons donc l'alphabet. Il n'y a aucun ampli audio de classe C, cette classe est exclusivement utilisée pour des amplificateurs RF. Pourquoi ? Tout simplement parce qu'un ampli de classe C est fortement non linéaire, tout le contraire de ce que l'on cherche en audio. Alors peut-il avoir une quelconque utilité ? Un signal radio est composé d'une porteuse HF sur laquelle est appliquée une modulation. Beaucoup de ces modulations ne modifient pas l'amplitude du signal. C'est le cas des modulations de fréquence ou de phase. Dans de tels cas, la linéarité de l'amplificateur n'est pas trop critique, d'où le concept d'amplificateur classe C (**fig. 3**). Comment ça marche ? Rappelez-vous que le transistor conduit tout le temps en classe A et la moitié du temps en classe B. En classe C, il conduit encore moins longtemps, typiquement seulement lorsque la tension d'entrée est proche de son pic. Cette brève conduction provoque une brutale chute de tension sur le collecteur, illustrée sur la figure. Imaginons qu'on applique une sinusoïde en entrée. Le signal de sortie aura la même fréquence et une phase inversée. Ce sera loin d'être une sinusoïde, mais son amplitude sera grande. Il ne reste qu'à ajouter un bon filtre passe-bande et on retrouve une belle sinusoïde amplifiée, qui suivra la modulation en fréquence ou en phase du signal d'entrée !

Comme vous pouvez l'imaginer, concevoir correctement un amplificateur de classe C est un peu plus délicat que les précédentes variantes, mais l'avantage est son efficacité : un amplificateur de classe C n'a que très peu de pertes et peut afficher un rendement de 60% ou même 70%, ce qui est très bon en RF. De plus un seul transistor est nécessaire. Il y a donc de bonnes chances pour que votre télécommande de porte de garage soit équipée d'un ampli RF de classe C.

Qu'en est-il de la classe D ?

Ah, la classe D ! En un mot, c'est la solution actuellement la plus utilisée en basses fréquences, en particulier en audio. Le concept, illustré à la **figure 4**, est simple. Tout d'abord le signal d'entrée analogique est converti en numérique via un modulateur de largeur d'impulsion (MLI, ou PWM en anglais). Une méthode simple pour cela est d'utiliser un générateur en dents de scie et un comparateur de tension, comme illustré sur la figure. On obtient ainsi un signal numérique « un bit » dont le rapport cyclique est proportionnel à la tension d'entrée. Bien sûr ce signal MLI sera produit de façon purement numérique si le signal d'entrée est une source numérique (lecteur CD, radio numérique, carte audio d'ordinateur, etc.), ce qui permet d'éviter toute perte de qualité.

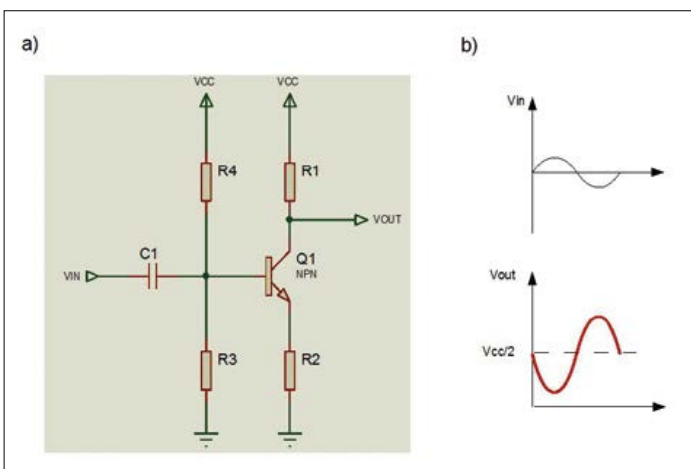


Figure 1. Un amplificateur de classe A peut être construit autour d'un simple transistor (a). Sa sortie est polarisée de manière à ce qu'il soit toujours conducteur (b).

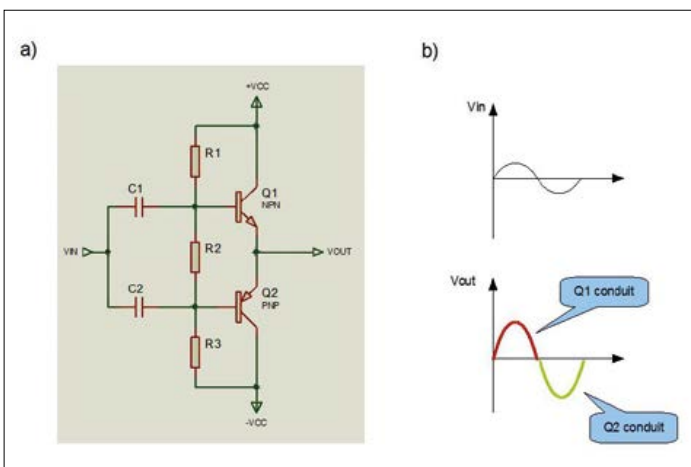


Figure 2. Un amplificateur de classe B est généralement construit autour d'une paire de transistors complémentaires (a). Chaque transistor est conducteur 50% du temps, ce qui minimise les pertes, mais introduit une distorsion de croisement (b).

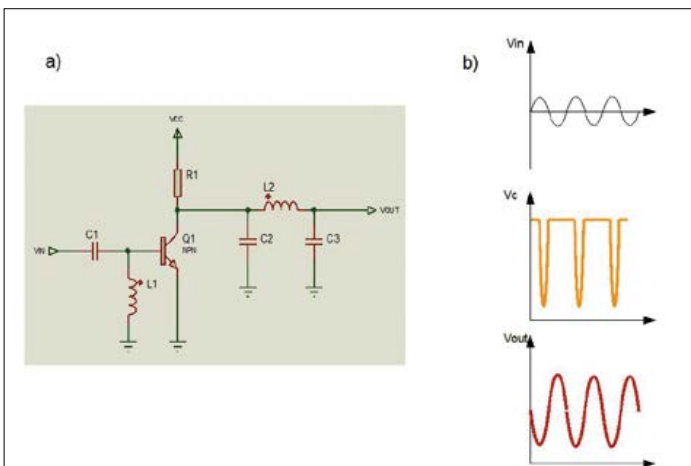


Figure 3. Un amplificateur RF de classe C n'a souvent pas de polarisation de la base (a). Le transistor est conducteur uniquement pendant une petite partie du temps, le signal de sortie est reconstruit grâce à un filtre passe-bas (b).

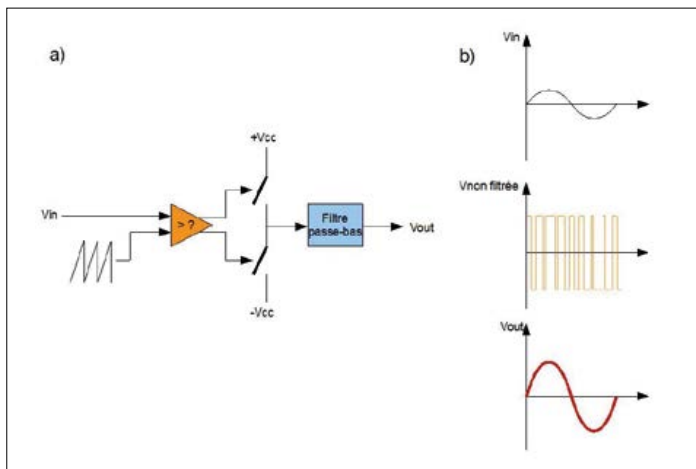


Figure 4. Un amplificateur de classe D est principalement numérique (a). Le signal audio est converti en MLI, amplifié, puis reconverti en analogique par un filtre de sortie (b).

Ce signal MLI est ensuite amplifié. La beauté de l'architecture classe D est ici : l'amplification d'un signal numérique peut être réalisée avec un rendement très élevé, car les transistors sont utilisés comme des interrupteurs marche/arrêt. La dissipation de puissance n'a lieu que durant les transitions d'un état logique à l'autre, et peut être minimisée avec des transistors assez rapides. Après cette amplification, on a évidemment toujours un signal MLI, mais de forte puissance. Comment en extraire un signal analogique ? Simplement avec un filtre passe-bas ! En audio, le haut-parleur lui-même peut faire partie de ce filtre, vu que c'est une inductance...

Évidemment il y a des difficultés. Tout d'abord, comme pour tout système numérique, monsieur Nyquist nous dit que la fréquence de la MLI doit être plus que deux fois supérieure à la fréquence la plus élevée présente dans le signal d'entrée. En pratique, afin de réduire le coût du filtre de sortie, celle-ci doit même être dix fois supérieure. Pour une bande passante de 20 kHz, on a donc typiquement des fréquences MLI de l'ordre de 200 kHz, soit des périodes de 5 µs. Ok, mais supposons maintenant que vous vouliez une précision d'amplitude de 16 bits. Cela implique que la position temporelle des fronts du signal MLI puisse être réglée avec une résolution de 1/65536, soit 5 µs/65536 = 76 ps... Oui, picosecondes. Cela explique pourquoi la construction d'un bon amplificateur de classe D n'est pas si simple.

L'autre souci avec la classe D est lié à la compatibilité électromagnétique. Commuter des signaux numériques de puissance à haute fréquence est la meilleure manière de récolter des soucis de type CEM ; or c'est exactement ce que fait un ampli classe D, c'est pourquoi il faut prendre quelques précautions lors de la conception des filtres et des circuits imprimés. Quoi qu'il en soit, les amplificateurs de classe D sont partout, et de très nombreux circuits intégrés dédiés facilitent le travail du concepteur (voir Texas, Maxim ou Linear par exemple).

Retour aux radiofréquences avec les classes E et F

Souvenez-vous que classe C rime avec RF. La classe E est une amélioration de cette dernière, qui permet d'augmenter l'efficacité jusqu'à 80 ou 90%. Comment ? Vous rappelez-vous

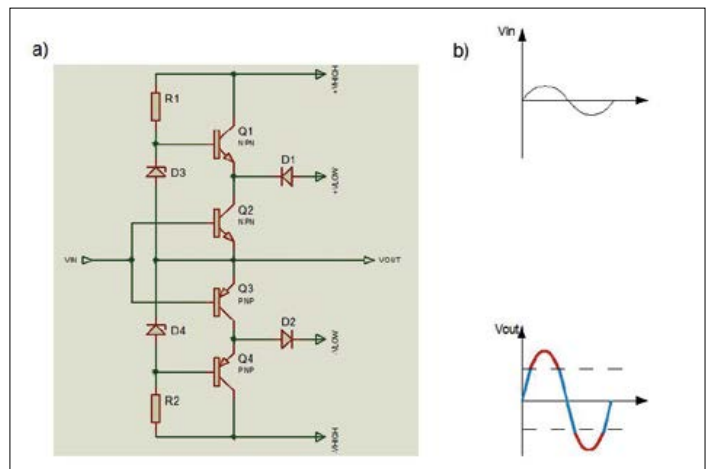


Figure 5. Un amplificateur de classe G utilise deux paires de rails d'alimentation (a). L'un est utilisé lorsque le signal a une faible puissance (bleu), l'autre entre en action pour des puissances élevées (rouge) (b).

que $P = U \times I$? Donc la puissance dissipée dans le transistor de sortie est nulle si la tension à ses bornes est nulle ou si le courant le traversant est nul. C'est l'idée à la base de la classe E : faire en sorte qu'on n'ait jamais simultanément une tension élevée et un courant intense sur le transistor. Cela peut sembler naïf, voire impossible, mais un amplificateur RF est en général utilisé seulement dans une bande de fréquence très étroite autour de la porteuse. On a recours à deux astuces pour réaliser un ampli de classe E : d'abord, comme pour la classe D, on utilise un transistor très rapide, ce qui évite de rester trop longtemps dans une zone linéaire. Ensuite une conception fine du réseau d'adaptation d'impédance de sortie est réalisée pour déphaser de 90° la tension et le courant à travers le transistor. Il faudrait un article complet sur le sujet, mais si vous êtes intéressé, je vous recommande le très bon article publié par Nathan O. Sokal dans le magazine QEX (voir [8]). Enfin juste un mot sur la classe F. C'est une version améliorée de la classe E ; elle optimise encore un peu les adaptations d'impédances pour réduire les effets des harmoniques, elle est donc toujours réservée à la HF.

Retour à l'audio : G, H...

Les fans d'audio ne pouvaient pas laisser les gars des RF prendre les dernières lettres de l'alphabet, ils ont donc ajouté quelques variantes à la liste. Les classes G et H sont des variantes d'une efficacité supérieure à celle de la classe AB classique. Dans les deux cas, l'amélioration est liée à l'alimentation. L'idée est simple : pour avoir une puissance de sortie élevée, il faut alimenter les transistors de sortie avec une tension assez élevée, mais à faible puissance cette tension d'alimentation élevée augmente les pertes et la dissipation dans l'étage de sortie. D'où l'idée de moduler la tension d'alimentation de l'étage de sortie en fonction de la puissance instantanée nécessaire. L'idée est astucieuse, en particulier en audio, car la plupart des morceaux ne nécessitent des puissances élevées que pendant certains fortissimo. Ok, ce n'est peut-être pas le cas avec les musiques écoutées par certaines tranches d'âge, mais vous avez compris l'idée...

Un ampli de classe G réalise cela en utilisant plusieurs tensions

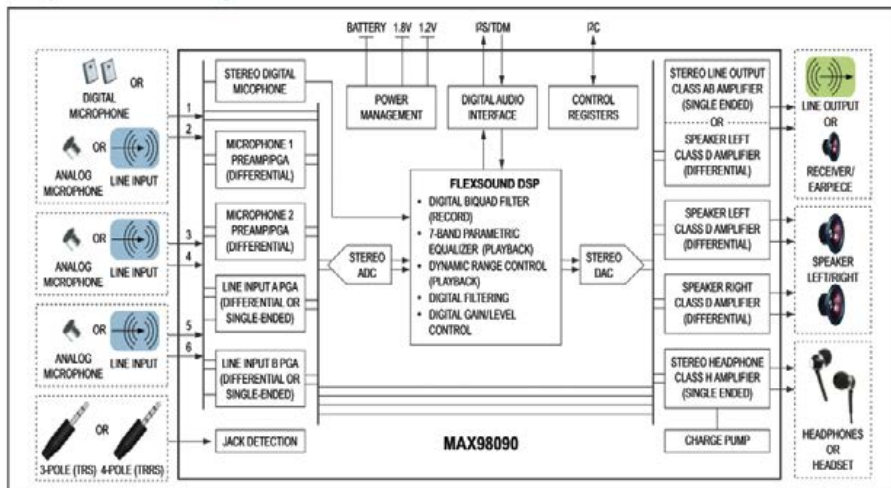


Figure 6. Un exemple d'ampli intégré multiclasses : le MAX98090. Source Maxim Integrated

d'alimentation, en général deux, et en commutant dynamiquement entre les deux selon le niveau du signal à amplifier (fig. 5). Tant que le signal d'entrée est suffisamment faible, l'amplificateur est de classe AB classique et utilise une alimentation basse. Quand l'amplitude du signal est plus élevée, un autre couple de transistors de sortie entre en action, alimenté par une tension plus élevée. La conception détaillée d'un tel ampli est plus complexe, en particulier pour réduire la distortion pendant les transitions entre chaque mode, mais vous avez saisi l'idée. Ah, un dernier mot : il n'y a de gain énergétique que si les deux tensions d'alimentation sont délivrées par deux transformateurs indépendants ou par un convertisseur DC/DC à haut rendement, sinon c'est l'alimentation qui chauffera au lieu des transistors... Dans le même esprit, un ampli de classe H va encore plus loin en utilisant une alimentation DC continument variable.

Pour conclure

La liste semble sans fin. J'ai même trouvé quelques références à un ampli audio de classe T (T pour *Tripath*, la société qui

visiblement a développé la chose), mais sans voir beaucoup de différences par rapport à la classe D.

Comme vous l'avez compris, le Graal des amplificateurs est une bonne efficacité énergétique combinée à une faible distortion et bien sûr un coût réduit. Le concepteur doit trouver la meilleure combinaison pour un projet donné. Les concepteurs de circuits intégrés ont aussi une palette de solutions impressionnantes à vous proposer. À titre d'exemple, regardez le schéma de principe de l'ampli audio MAX98090 de Maxim (fig. 6). Il accueille dans un seul et même circuit intégré un amplificateur stéréo (pouvant être au choix un ampli de classe D de 3 W ou un ampli de classe AB pour une sortie ligne), mais aussi un ampli de classe H pour une sortie casque indépendante. Et pour faire bonne mesure, il offre aussi une poignée de préamplificateurs, un CA/N, un CN/A et un DSP, le tout pour 3 \$ si vous en achetez une caisse... De tels circuits intégrés ou leurs petits frères plus simples répondent aux besoins de 99% des applications. Ceci dit, les électroniciens débattent depuis des lustres sur les avantages et inconvénients des différentes classes d'amplificateurs. J'espère que ce petit article vous permettra de mieux vous y retrouver et de choisir la solution la plus adaptée à votre prochain projet. Maintenant, si le sujet vous intéresse, allez fureter du côté des articles et ouvrages indiqués dans la section « références », vous y trouverez de quoi occuper vos soirées !

Cet article a été publié dans la revue *Circuit Cellar* (n°281, décembre 2013).

(150527-1)

Références

- [1] *Hors-circuits : comprendre l'amplificateur à transistors*, Robert Lacoste, Elektor, 09/2015
- [2] <https://en.wikipedia.org/wiki/Amplifier>
- [3] *Audio Power Amplifier Design Handbook*, Douglas Self, ISBN 978-0240521626
- [4] *RF Power Amplifiers for Wireless Communications*, Second Edition, Steve C. Cripps, ISBN 978-1596930186
- [5] *Class D Audio Amplifiers: What, Why, and How*, Eric Gaalaas, Analog Devices Inc, www.analog.com/library/analogdialogue/archives/40-06/class_d.pdf
- [6] Note d'application AN-1071 : *Class D Audio Amplifier Basics*, IRF, Jun Honda & Jonathan Adams, www.irf.com/technical-info/appnotes/an-1071.pdf
- [7] *Class G/H amplifiers: how well do they deliver on their promise of high audio quality and low power consumption?*, AustrianMicrosystems, H. Theiler, H. Lenhard, H. Gether https://shop.austriamicrosystems.com/eng%20/content/download/163207/793905/file/TA_Audio_HeadphoneAmplifiers_AS3561_e.pdf
- [8] *Class E RF power amplifiers*, Nathan O. Sokal, QEX magazine, Jan/Feb 2001, www.eel.ufsc.br/~lci/siteramo/documentos/mestres/artigos/classe.pdf
- [9] *Doherty Amplifier Design*, Ken Mcknight, Microwave Monolithic Integrated Circuit (MMIC) Course, Johns Hopkins University, www.apl.jhu.edu/Notes/Penn/Cover09_files/MMIC_reportPA2.pdf
- [10] *Yamaha Power Amplifier*, White Paper, August 2008, http://download.yamaha.com/api/asset/file?language=en&site=countrysite-master.prod.exp.yamaha.com&asset_id=53052
- [11] Maxim, ampli audio MAX98090



bienvenue dans votre e-choppe

Elektor recommande



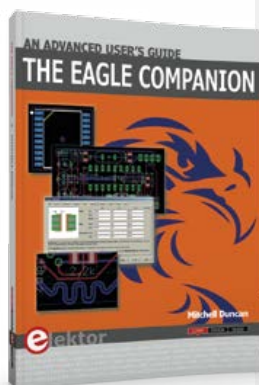
Le livre THE EAGLE COMPANION, nouveau chez Elektor, est un guide avancé (600 pages). L'auteur Mitchell Duncan permet aux utilisateurs d'EAGLE de pénétrer dans les entrailles de ce logiciel, du fonctionnement du programme jusqu'aux fonctions les plus sophistiquées. Je pensais bien connaître EAGLE, mais ce livre explore les modules, commandes et fonctions les plus ambitieux qui rendent ce logiciel incontournable dans le monde de la CAO.

Même si l'internet est un puits d'information sans fond, j'aime me documenter dans les

LIVRES, en particulier lorsque je suis confronté à un circuit ardu. Je garde précieusement dans ma bibliothèque mon exemplaire du livre de Mitchell. Le trésor caché d'EAGLE est son langage ULP (*User Language Program*), un outil pour l'adapter aux besoins et préférences de l'utilisateur. Mitchell, Cadsoft et moi avons obtenu que le manuel complet d'ULP soit inclus dans ce livre. Si EAGLE est votre fidèle compagnon, ce livre vous est indispensable. Attention : il sera bientôt dans votre boîte aux lettres !

Jan Buiting, Rédacteur en chef Elektor UK/USA

<http://www.elektor.fr/the-eagle-companion>



Vos favoris :

1. SmartScope Labnation

www.elektor.fr/labnation-smartscope



2. L'électronique pour les débutants

www.elektor.fr/l-electronique-pour-les-debutants

3. Raspberry Pi 2 (Mod. B)

www.elektor.fr/raspberry-pi-2-model-b

4. Domotique avec Zelio Logic

www.elektor.fr/domo-zelio

5. Collection de capteurs Arduino

www.elektor.fr/arduino-sensor-kit

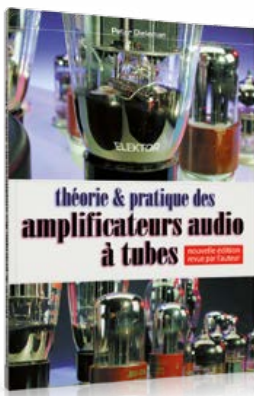
6. PicoScope 2204A

www.elektor.fr/picoscope-2204a

6. L'intégrale des articles d'Elektor (PDF)

www.elektor.fr/compilation-arduino-e-book

Théorie & pratique des amplificateurs audio à tubes



La mise au point d'amplificateurs à tubes nous met aux prises avec la rareté des tubes dont il ne reste qu'une fraction des nombreux types d'origine. Heureusement suffisamment de bons ! En fait, la grande pénurie est celle des connaissances : quels sont les avantages du SRPP ? Qu'en est-il des tubes à faible souffle ? Est-il souhaitable ou regrettable que la pentode produise des harmoniques de rang 3 ? Pourquoi le rendement des triodes plafonne-t-il à 25% ?



Prix (membres) : 39,15 €

www.elektor.fr/audio-a-tubes

L'intégrale des articles d'Elektor



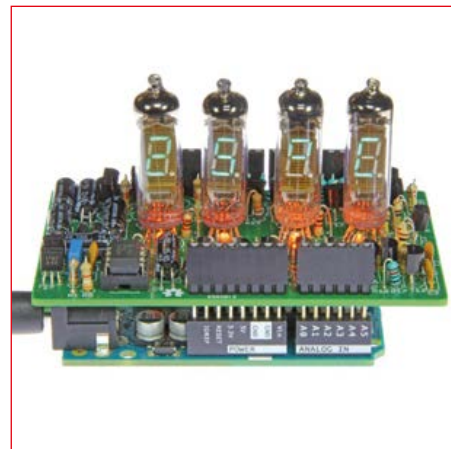
249 Pages bourrées d'idées, d'explications, d'astuces, de schémas, de programmes, de circuits imprimés. Des heures de lecture instructive et stimulante ! Cette compilation réunit l'intégrale des articles parus dans Elektor entre juillet 2012 et novembre 2014. Les liens dans la table des matières de ce document numérique (PDF) permettent de naviguer facilement vers les articles qui intéressent.



Prix (membres) : 9,50 €

www.elektor.fr/compilation-arduino-e-book

Shield VFD pour Arduino

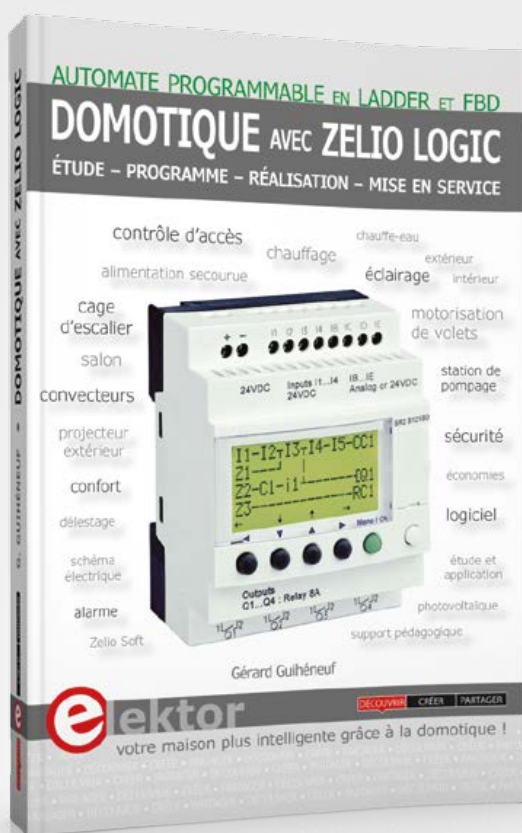


Ce *shield* Arduino contient quatre tubes VFD russes IV-3 à 7 segments avec les pilotes nécessaires. Quatre LED de 3 mm fournissent le rétroéclairage des tubes. Le logiciel disponible permet d'utiliser le *shield* avec une carte Arduino Uno pour créer une horloge, un compteur, un voltmètre... Le circuit ne repose que sur des composants traversants, aucun composant CMS. Le circuit imprimé peut facilement être garni par un novice dans le soudage.



Prix (membres) : 35,96 €

www.elektor.fr/vfd-shield-for-arduino



Domotique avec Zelio Logic

Le module programmable compact Zelio Logic avec 12 entrées/sorties est une solution technique bon marché et facile à mettre en oeuvre, dans un logement, pour automatiser l'éclairage, piloter le chauffage, optimiser la consommation d'électricité ou encore pour se protéger contre les intrusions.

Ce livre montre comment exploiter Zelio Logic avec son outil de programmation graphique en LADDER ou FBD (logiciel pour PC Zelio Soft 2). Le premier projet décrit (station de pompage domestique) permet au lecteur de s'initier à ces deux langages. Le dernier chapitre, conforme au thème « Domotique et confort » et complété par des activités pédagogiques, intéressera particulièrement les professeurs.

Domotique avec Zelio Logic

Une maison intelligente grâce à la domotique et un petit automate programmable

PicoScope

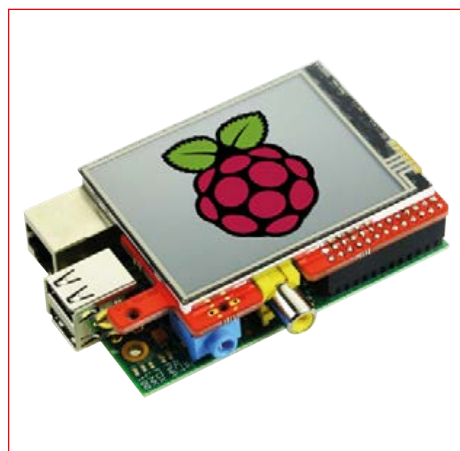
oscillo au format de poche



Prix (membres) : 41,85 €

www.ektor.fr/domo-zelio

Écran tactile pour Raspberry Pi



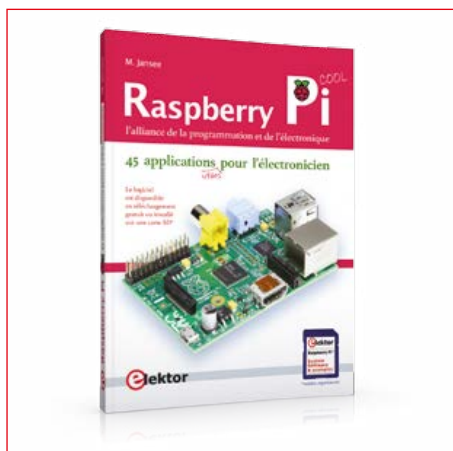
Notre boutique s'est enrichie de l'écran tactile pour Raspberry Pi de nos amis de Watterott Labor. Ajoutez à votre Raspberry Pi modèle B+ ou Raspberry Pi 2 un écran couleur de 2.8» (320x240 pixels) avec un panneau tactile. Un pilote de tampon mémoire (*framebuffer*) est disponible pour l'écran, ce qui permet une mise en service en un rien de temps. Une extension pour vos projets Raspberry Pi à un prix très raisonnable !



Prix (membres): 31,46 €

www.ektor.fr/touchdisplay-rpi

Raspberry Pi : 45 applications utiles pour l'électronicien



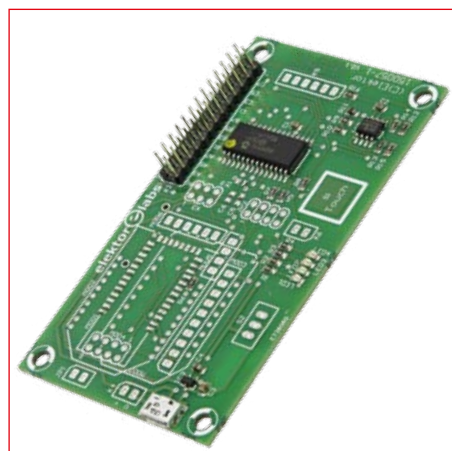
Après une courte présentation du R. Pi, vous installez le logiciel. Suit une introduction brève au système Linux et à la programmation en Bash, Python et Javascript. L'auteur expose seulement ce qui est nécessaire pour comprendre les projets et les adapter à vos besoins, puis il vous propose des projets plaisants, certains simples (clignotant alternatif, commande de moteurs), d'autres compliqués (régulation de vitesse de moteur, serveur ouïe avec CGI...)



Prix (membres) : 30 €

www.ektor.fr/rpi

Carte d'E/S Android



La carte d'E/S Android permet de piloter les ports d'un microcontrôleur PIC avec un appareil sous Android, par Wi-Fi, USB ou Bluetooth (versions 2.0 et 4.0). Les classes JAVA (bibliothèques) spécialement développées pour commander la carte d'E/S Android permettent de se concentrer sur l'application plutôt que sur la manipulation des entrées/sorties. La carte d'E/S Android basée sur un PIC16F1938-I/SO permet d'accéder aux 25 broches d'E/S.



Prix (membres): 22,46 €

www.ektor.de/android-i-o-board



par **Rei Vilo**

J'ai été séduit par l'idée de Microduino : tout un écosystème composé de deux cartes et de nombreux *shields* disponibles dès le départ ! En outre le prix est vraiment attractif. Enfin, au lieu de perdre du temps à chercher des capteurs et de passer d'interminables heures à souder et câbler, il suffit de prendre les cartes d'extension dont on a besoin pour son projet et de les empiler pour un résultat compact, en toute sécurité.

Pour les *shields* Arduino historiques, il y a des cartes d'extension spécifiques compatibles broche à broche avec une carte Arduino Uno. Microduino a enrichi ses pages wiki avec toute une section dédiée à la formation.

Microduino propose des cartes compactes et empilables avec nombre de *shields*, d'excellents tutoriaux, du matériel et du logiciel totalement compatibles avec Arduino. C'est idéal pour des applications où l'encombrement est le premier des soucis !

Rei Vilo publie des billets sur l'informatique embarquée sur son blog embeddedcomputing.weebly.com.

www.elektor.fr/microduino-basic-tutorial-kit



... vous aussi envoyez-nous votre avis sur votre produit favori d'Elektor ; si nous le retenons pour une publication, vous recevrez en échange un bon d'achat de 100 € à valoir dans l'e-shoppe d'Elektor...

J'essaye : www.elektor.fr/rotm



JOYEUX HALLOWEEN!

EFFETS SONORES ET LUMINEUX POUR VOTRE POTIRON AVEC LE KIT **SCARY HALLOWEEN** D'ELEKTOR

kit complet
32,50 €

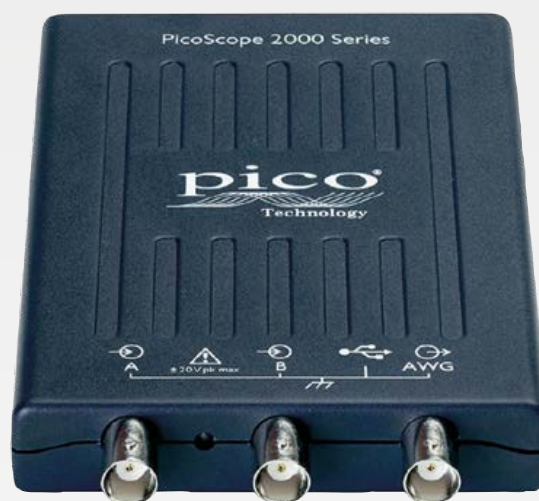
WWW.ELEKTOR.FR/HALLOWEEN



Oscilloscopes de la série **PicoScope 2200A** : petits, légers, modernes

Les oscilloscopes PicoScope 2200A sont une alternative moderne, légère et compacte aux volumineux oscilloscopes de table. Vous pouvez glisser ces oscilloscopes 200 MHz, 1 Gs/s dans la sacoche de votre ordinateur portable ! Ce sont des outils idéaux pour les ingénieurs en déplacement, adaptés à toute une variété d'applications : conception, essais, enseignement, entretien, suivi, recherche des défauts, réparation...

Ces oscilloscopes fonctionnent sur PC. Grâce au logiciel PicoScope 6, ils offrent des fonctions de pointe (décodage sériel, tests de limite de masque...). De nouvelles fonctions, optimisées grâce aux commentaires des clients, sont régulièrement disponibles (mises à niveau gratuites).



Domotique avec Zelio Logic

Une maison intelligente grâce à la domotique et un petit automate programmable

PicoScope

oscillo au format de poche



Prix (membres) : 143,10 €
www.elektor.fr/picoscope

Les microcontrôleurs PIC pour les débutants



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

Prix (membres) : 34,16 €

www.elektor.fr/debutpic

L'électronique pour les débutants

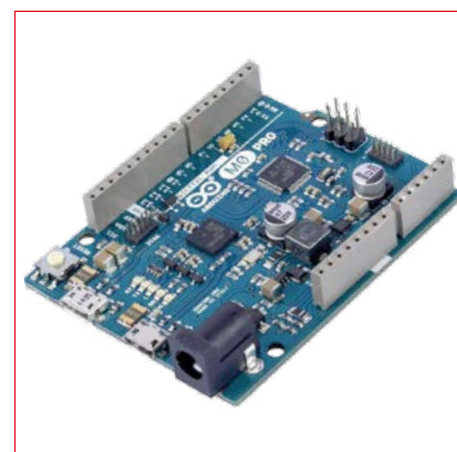


Vous rêvez de débiter en électronique, mais par où commencer ? Peut-on se fier aux schémas d'origine inconnue sur l'internet ? Faut-il acheter un livre avec des montages simples ou plutôt un livre sur les composants ? Fin pédagogue, l'auteur répond ici à toutes les questions du néophyte : quel fer acheter ? Un multimètre à 5 €, c'est bien ? Un oscilloscope, c'est indispensable ? Et bien d'autres interrogations que trop de livres laissent en suspens.

Prix (membres) : 38,25 €

www.elektor.fr/debut

Arduino M0 Pro



La carte Arduino M0 pro est une simple, mais puissante, extension à 32 bits de la plateforme Arduino Uno. La carte est animée par un μ C SAMD21 d'Atmel, avec un cœur ARM Cortex® M0 à 32 bits. La puissance du cœur Atmel donne à cette carte une plus grande souplesse et permet d'envisager des projets de grande envergure ; en outre, la carte Arduino M0 pro est un outil idéal pour apprendre à développer des applications à 32 bits.

Prix (membres) : 40,46 €

www.elektor.fr/arduino-m0-pro

le XXI^e siècle vient seulement de commencer
tout reste à inventer



j'explore l'électronique,
je lis **elektor**

www.elektor.fr/membres

bienvenue dans la section **PARTAGER****Thijs Beckers** thijs.beckers@eimworld.com**du changement dans la section PARTAGER**

Vous aviez sans doute pris l'habitude de savourer ici les morceaux choisis par notre collègue Jaime venu d'Hispanie. Toujours est-il que c'était le cas pour moi. Toutefois je dois parler d'ex-collègue parce que Jaime a quitté Elektor, certes à regret, mais pour rejoindre Berlin, cette immense métropole culturelle. Il avait trouvé sa place parmi tous les collaborateurs d'Elektor. À l'occasion, il continuera à travailler avec notre équipe, comme par le passé, mais de l'extérieur à présent.

D'humeur perpétuellement joyeuse, il nous a fait montre d'un art consommé pour dissimuler son agacement quand les délais de remise des articles à publier n'étaient pas respectés. Lors des expositions et des salons, il nous a donné de sérieux coups de main. Sur nos stands, il était disponible tout au long de la journée et ne se lassait pas de discuter avec les visiteurs du mieux qu'il pouvait en néerlandais, allemand, anglais et bien sûr en espagnol. Et jusqu'aux petites heures, il lui arrivait de conduire

une fourgonnette pour transporter notre matériel ou simplement faire le taxi çà et là. Pendant son temps libre, il s'adonne à la course extrême et autres parcours du combattant. Mais il apprécie aussi beaucoup la musique et nous avons eu de nombreuses conversations à ce sujet.

Il nous manquera et nous lui souhaitons « alles Gute », « all the best », « todo lo mejor » et plein de succès dans son nouveau travail.

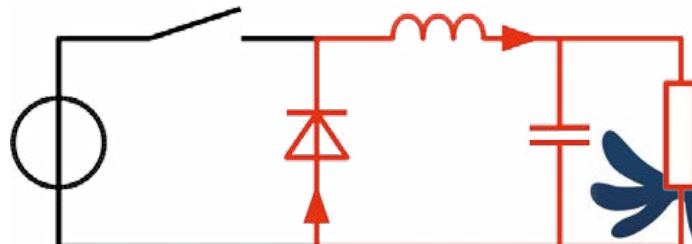
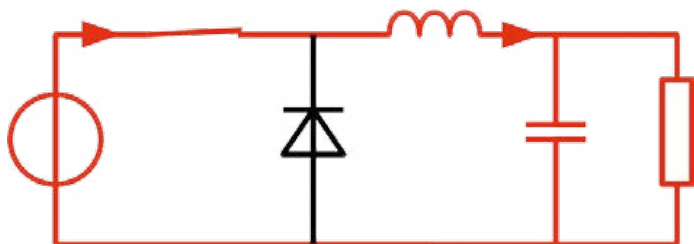
C'est moi qui le remplacerai pour l'éditorial de la section Partager, mais vous l'aviez sûrement deviné... ◀

(150481 – version française : Robert Grignard)



convertisseur *buck* têtu

au comportement erratique



Thijs Beckers (rédaction des Pays-Bas)

À l'aide ! Nous ne nous en sortons pas au labo d'Elektor. Ce convertisseur Buck n'en fait qu'à sa tête, il est incontrôlable. Nous ne trouvons pas ce qui cloche, peut-être que vous aurez une idée...

« Facile à dire après coup, mais je n'aurais pas dû choisir ce circuit intégré (LM2677T-ADJ) », voilà le commentaire de notre collègue chargé de réaliser une nouvelle alimentation.

« Même avec un diviseur de tension pour 30 V en sortie, une demande de courant de plusieurs ampères (impulsion) a des conséquences dramatiques ».

Reprenons l'histoire du début. L'idée consistait à concevoir une alimentation de laboratoire avec les spécifications suivantes :

- Tension de sortie variable de 0 à 30 V
- Au moins 3 A avec limitation
- Stable dans la plupart des applications usuelles d'un laboratoire

C'est une description assez floue, ce qui laisse au concepteur pas mal de choix entre diverses topologies. Mais ce n'est pas pour rien que nous nous appelons Elektor, et nous n'avons pas bâti notre réputation avec des idées banales, ni en recopiant des circuits ; bien entendu, nous ne choisissons pas la voie la plus aisée, ni la plus évidente. Des surprises vous attendent donc, ce circuit ne fait pas exception.

Après avoir fureté chez les fabricants à la recherche de circuits intéressants, nous sommes tombés sur le LM2677T-ADJ de Texas Instruments. Il est relativement simple à mettre en œuvre, offre un rendement élevé, une bonne stabilisation de la tension et un courant maximal de sortie de 5 A. C'est prometteur, maintenant au travail !

Ainsi est née la première version du schéma (v0.01). Une tension (raisonnablement) stabilisée de 36 V alimente le circuit (via K1). Un JFET de type P (T1) prévient les effets indésirables des marches/arrêts. La tension de sortie est réglée via l'entrée de contre-réaction (*feedback*). Grâce à la régulation active (avec les amplis op IC2B et IC2D), nous pouvons directement mettre

en œuvre la limitation de courant. La résistance de *shunt* R12 et IC2A permettent la mesure du courant. L'alimentation de IC2 (5 V) est fournie par un LM317 (IC3). C4, C5 et R10/C13 assurent la stabilité de la boucle en charge. Passons à la construction d'un prototype.

Sans charge externe, la tension de sortie minimale de notre prototype est de 0,3 V. Avec une charge de 40 Ω , on arrive à un minimum d'environ 0,017 V. Ce n'est pas mal sans l'aide d'une tension négative. Le rendement mesuré est lui aussi excellent, presque 94 % pour 30 V et 5 A en sortie.

Le test suivant, délivrer une impulsion de 5 A avec un rapport cyclique de 50 %, s'est moins bien déroulé. La tension de sortie s'est effondrée comme un château de cartes, et elle n'était plus du tout stable. Aucune combinaison d'inductance et de capacités du filtre de sortie (L1, C10-C12) n'arrangeait les choses ; 82 μH au lieu de 22 μH pour L1 ne changeait rien. Le seul « avantage » est que la tension s'effondre lors des impulsions de courant, au lieu de grimper au-delà de la tension fixée (à part un léger dépassement). Une plus grosse capacité de sortie a un effet positif sur la régulation des basses tensions, mais l'effondrement de la tension s'accroît encore pour des tensions supérieures à 20 V. Un rapport cyclique d'environ 50 % pour la régulation rend le régulateur quelque peu instable, ce qui a aussi un effet néfaste sur la régulation de la limitation de courant.

Réunir toutes les masses en un seul point améliore un peu les résultats. Un plan de masse sur le dessus du circuit imprimé sera sans doute indispensable (le circuit imprimé du prototype était simple face), mais il y a peu de chance que cela améliore sensiblement le comportement de la régulation. Lors de simulations avec Webench (webench.ti.com), les calculs donnaient des valeurs différentes des composants du filtre en fonction de la tension de sortie... Le LM2677 est peut-être très sensible aux ondulations à la sortie suite à la commutation de la charge ?



Une petite surprise nous attendait lors d'expérimentations sur une charge commutée, avec une alimentation « normale » et un filtre LC (82 μ H et 220 μ F) : à 1,32 kHz (presque la fréquence de résonance) et pour un rapport cyclique de 50 %, le courant continu circulant dans la bobine est égal à environ la moitié du courant de crête. Étonnamment, cela semble moins sur un écran d'oscilloscope, mais le multimètre indique bien la moitié. Il y a aussi une composante alternative, de fréquence égale à celle de commutation de la charge, et elle est pratiquement sinusoïdale. Notre test semble montrer que la pointe de courant (continu plus alternatif) est une fois et demie la valeur maximale du courant sans que la charge soit commutée. Donc, si l'alimentation délivre

3 A en continu, on s'attend à commuter 1,5 A en moyenne, avec un rapport cyclique de 50 %. Le courant de crête passe soudain à 4,5 A si on commute à la fréquence de résonance du filtre. Le régulateur de notre

alimentation devrait donc pouvoir délivrer ± 8 A pour répondre à un appel de 5 A sans problème...

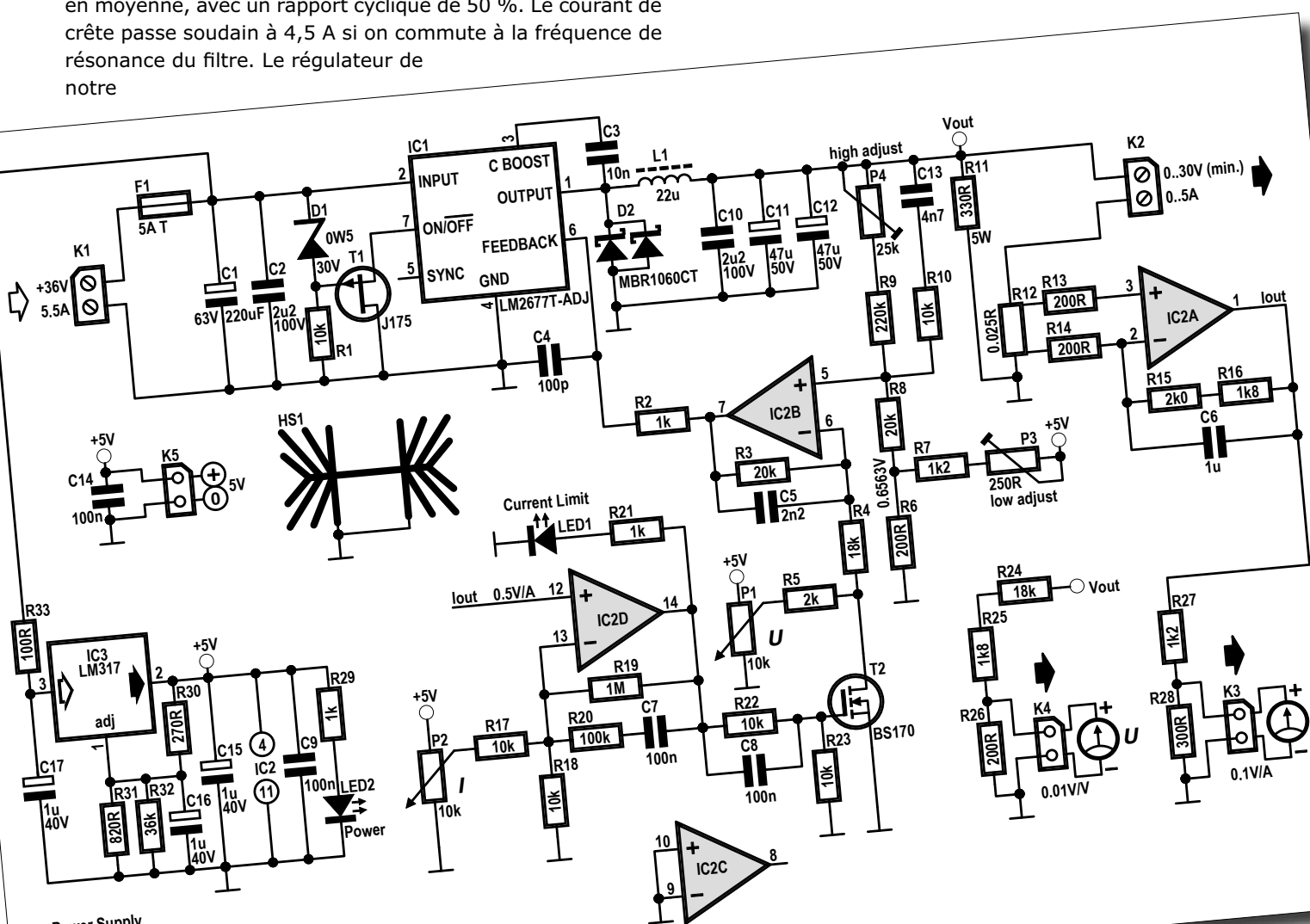
L'internet n'est pas d'un grand secours, nous n'avons en tout cas rien trouvé de concret. Nous ne sommes quand même pas les premiers à être confrontés à un tel comportement ? Et puis, une telle charge « impulsive » est-elle vraiment réaliste ? On pourrait bien sûr fixer une limite plus basse pour le courant, mais ce serait dommage : si la charge est relativement constante, 5 A peuvent être fournis.

Est-ce un comportement typique des convertisseurs de type *buck* avec un filtre en sortie ? Est-ce que le filtre entre en résonance (série) ? Car tout se détraque lorsque la fréquence de commutation de la charge approche de la fréquence de résonance du filtre. Peut-être y a-t-il un courant parasite qui circule ?

Que pensez-vous du raisonnement suivant : une charge commutée engendre des dépassements de tension ; mais puisque le régulateur tente de maintenir la tension de sortie constante, l'énergie stockée dans le circuit LC n'a d'autre choix que de s'évacuer sous forme de courant. Ce qui nous donne la pointe de courant.

N'hésitez pas à nous faire part de votre avis. Écrivez-nous à redaction@elektor.fr. À suivre... ◀

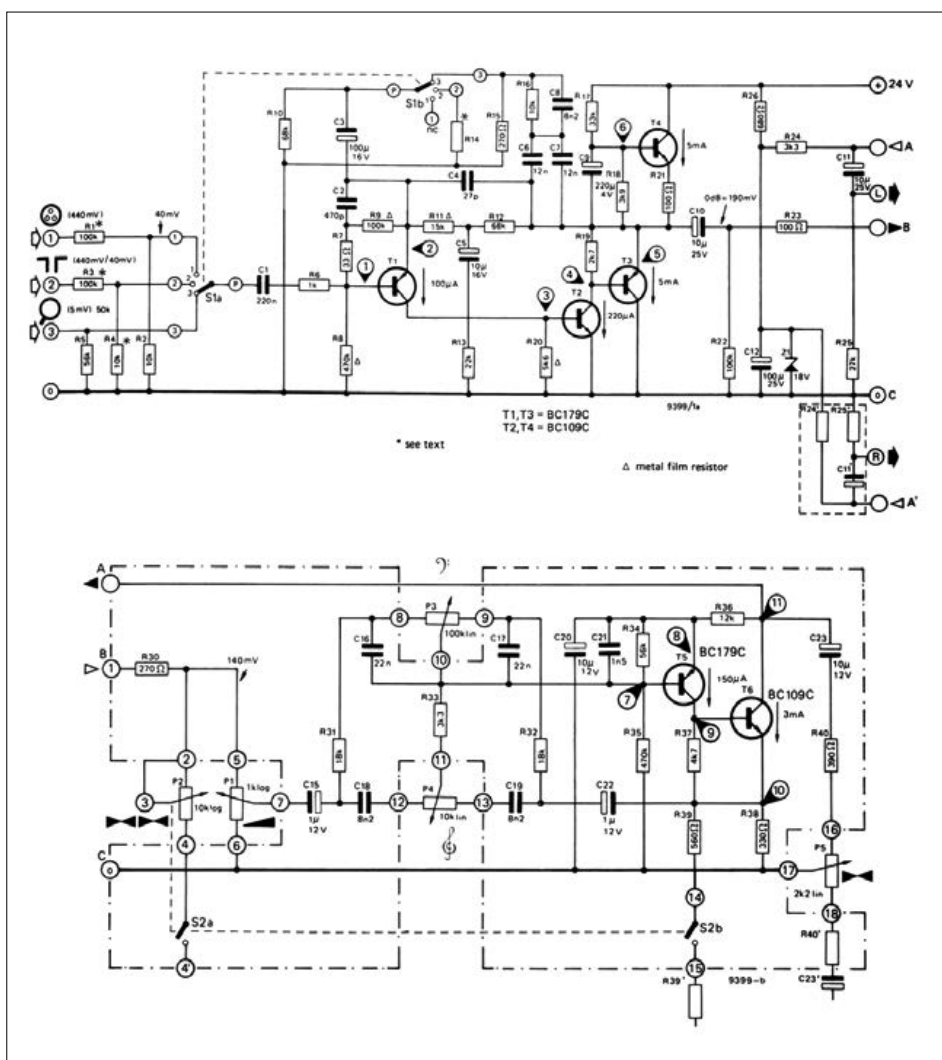
(150487 - version française : Jean-Louis Mehren)



Power Supply
Corrected 19-5-2015
150161-1
v0.01

préamplificateur Preco

39 ans en mode Hi-déFi (et comment j'ai réduit son bruit au silence)



La norme allemande DIN 45500 spécifiait les exigences minimales auxquelles un appareil audio devait répondre pour être qualifié de « Hi-Fi ». Si elle est depuis devenue obsolète, à l'époque déjà elle ne reflétait pas vraiment l'état de l'art. Les magazines spécialisés plaçaient la barre Hi-Fi un peu plus haut, ce qui chez Elektor donna l'amplificateur Equa.

La conception du projet Preco de 1976 avait ceci d'inhabituel qu'elle était composée de deux cartes : l'étage d'entrée, et un amplificateur variable (**fig. 1**). Outre les habituelles commandes de volume, de tonalité et d'équilibre des canaux, l'amplificateur variable permettait de régler la « largeur stéréo ». Les valeurs pouvaient aller de zéro (c.-à-d. mono) à environ deux fois la distance entre les haut-parleurs. Bien plus qu'un gadget, ce réglage permettait d'améliorer nettement l'image stéréo en fonction du placement des haut-parleurs et du matériel audio utilisé. L'ajout délibéré d'une diaphonie déphasée est une technique encore employée de nos jours, même si elle n'égale généralement pas la finesse du Preco : un téléviseur moderne peut ainsi proposer une « largeur stéréo » ou un équivalent dans un obscur sous-menu, mais souvent les seuls réglages possibles sont « on » et « off » et l'effet obtenu, à mes oreilles tout du moins, est généralement excessif et vite désagréable.

Le réglage de la largeur stéréo était

Figure 1. Schémas originaux du Preco.

Jürgen Friker (Allemagne)

Le préamplificateur Preco conçu par T. Meyrick a été décrit dans les numéros d'avril et mai 1976 d'Elektor. Sa qualité égalait bien sûr les standards Elektor, mais il était aussi beaucoup plus « Hi-Fi » que les exigences de la norme allemande alors en vigueur. Et il peut encore servir. D'ailleurs je l'ai amélioré !

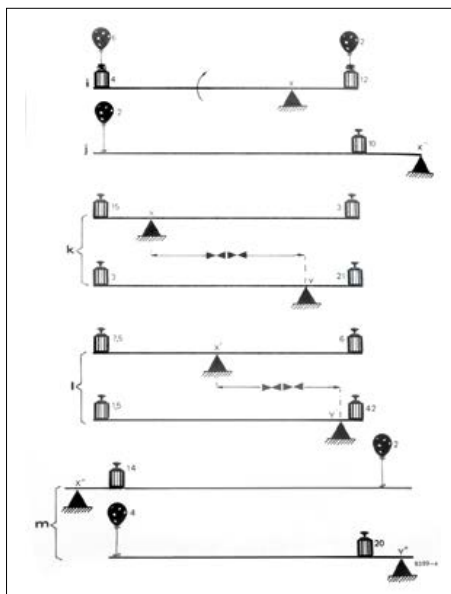


Figure 2. Une balance virtuelle, des poids et des ballons pour expliquer de façon imagée le contrôle de la largeur stéréo.



Figure 3. La télécommande du Preco.

cependant bel et bien une fonction de luxe en 1976, et Elektor s'évertua à en expliquer le principe à l'aide d'un dessin, reproduit ici à titre récréatif (**fig. 2**). Mode elektorienne typique des années 1970, le nom *Preco* avait été forgé par combinaison des mots *preamplifier* et *control unit*.

Une conception universelle

Un préamplificateur standard possède d'ordinaire un étage d'entrée pour une tête de lecture à aimant mobile, y compris un égaliseur implantant la courbe RIAA. Le préamplificateur/égaliseur peut être mis en œuvre au moyen par exemple d'un AOP à faible bruit par canal. Pour les applications haut de gamme, on ne recourt pas à une puce du marché, mais de préférence à un circuit construit à partir de composants discrets. L'étage d'entrée du Preco est arrangé de façon classique, si ce n'est que les lignes d'en-

trée suivent toutes le même chemin. Cela signifie en particulier que tous les signaux passent par l'étage préamplificateur. La sensibilité est réglable, ce qui permet de répondre à l'ensemble des niveaux d'entrée possibles sans réduire l'entrée du préamplificateur. On devrait même pouvoir l'utiliser comme amplificateur de microphone, cela dit je n'ai pas testé cette possibilité, car mon magnétophone possède déjà un ampli.

Disons pour résumer que le Preco était (et est toujours) un préamplificateur universel qui peut servir dans de nombreuses applications, en particulier lorsqu'on souhaite utiliser des sources à faibles niveaux de sortie, p. ex. un magnétophone à tête de lecture magnétique.

Télécommande

Une autre caractéristique du Preco est sa télécommande, l'amplificateur variable complet est placé dans un boîtier externe

(**fig. 3**). Je peux ainsi régler volume, équilibre des canaux et largeur stéréo depuis mon fauteuil. C'est bien, mais cette télécommande m'a surtout permis de monter l'ensemble, dont le préamplificateur, la radio et l'amplificateur de sortie (un « Elektornado »), dans un coffret rétro sans que j'aie à y percer des trous supplémentaires qui auraient gâté son charme désuet. La télécommande est optionnelle ; si elle ne sert pas, les deux cartes du Preco peuvent être insérées dans un boîtier, le Preco prend alors l'aspect de n'importe quel préamplificateur. Les figures 4, 5 et 6 montrent le Preco monté dans ma radio à lampe Lumophon. Je sais... la poussière... Mais après tout il s'agit de la rubrique Rétronique, non ? La séparation de l'étage d'entrée et de l'amplificateur réglable force le signal audio à faire de longs allers et retours jusqu'au boîtier de la télécommande, ce qui est plutôt étrange. On est en droit de



Figure 4. Le préampli Preco monté dans ma radio à lampe Lumophon.

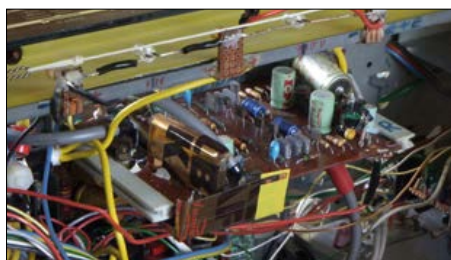


Figure 5. La partie pré-amplification du Preco vue de côté.



Figure 6. Le Preco incorporé parmi l'électronique et la mécanique de mon poste Lumophon.

se demander si un préamplificateur de qualité peut tolérer des chemins aussi longs pour ses signaux, en particulier au regard de la capacité du câble et de sa capacité à agir (malgré son blindage) comme une antenne captant les signaux parasites.

La capacité du câble, qui dépend bien sûr de sa construction et de sa longueur, forme un filtre passe-bas RC avec l'impédance de sortie de l'amplificateur. Selon l'article de 1976, cette impédance est si faible que même un câble long n'a qu'un effet négligeable sur la réponse en fréquence du Preco, et toute interférence externe a de grandes chances de rester inaudible, ce que confirme la pratique. La faible impédance de sortie signifie par

ailleurs qu'un casque audio d'impédance moyenne peut être piloté directement depuis le boîtier de la télécommande au moyen de condensateurs de blocage CC, car, comme nous allons le voir, la sortie de l'amplificateur variable achemine aussi l'alimentation.

CC : danger !

La connexion entre l'étage d'entrée et l'amplificateur variable ne nécessite que cinq fils (quatre plus des blindages) ainsi que des câbles adaptés, conçus pour les interconnexions stéréo permettant la lecture et l'enregistrement ; ils sont souvent équipés de prises DIN à cinq broches et faciles à trouver (les Allemands les appellent *Diodenkabel*, un

nom assez curieux). La conséquence de cette connexion est que l'alimentation de l'amplificateur variable (18 V) passe par les fils transportant les signaux. J'y reviens sous peu.

Quel est ce bruit ?

Pour certaines résistances, l'article original préconisait des films métalliques plutôt que des films de carbone, car les résistances à film métallique sont supposées être plus performantes vis-à-vis du bruit. Cette recommandation signifie sans doute qu'un utilisateur avait jugé le Preco plutôt bruyant pour un préamplificateur haut de gamme, tout du moins pour un préamplificateur se voulant meilleur que la norme DIN. Cela dit, si la qualité du Preco était telle qu'il fallait se préoccuper du bruit thermique de ses résistances, cela n'en aurait pas fait un projet *fais-le-toi-même* très réaliste ; et à l'évidence les sources audio (non numériques) auraient de toute façon été bruitées. J'ai tout de même suivi les conseils de 1976 et utilisé des résistances à film métallique ainsi que des transistors à faible bruit BC414/BC416. J'ai donc été d'autant plus contrarié de découvrir que la sortie restait notablement parasitée même en l'absence d'un quelconque signal d'entrée. Ayant éliminé les résistances et les transistors comme coupables potentiels, j'ai cherché à déterminer l'origine de ce bruit.

En écoutant plus attentivement et en me servant du réglage de la largeur stéréo, il m'a semblé que le bruit provenait d'un certain point du champ stéréophonique. C'était plutôt surprenant, car le bruit stéréo à la sortie d'un récepteur FM (p. ex.), qui est particulièrement marqué lorsque la réception est mauvaise, ne provient pas d'un point bien déterminé de l'espace : qu'on utilise les haut-parleurs ou des écouteurs, il est impossible de déterminer la direction du bruit. Dans le cas du Preco, le bruit des deux canaux n'est toutefois pas indépendant, mais synchrone. Ce qui compte est que le bruit est déphasé entre les deux canaux ; ainsi ce bruit disparaît-il lorsque le récepteur est mis en mode mono par réglage de la largeur stéréo et que les deux canaux sont additionnés. Cela signifie à l'inverse que si le bruit est perçu depuis un point bien défini de l'espace, alors un canal a vraisemblablement été connecté à l'envers et la qualité stéréophonique laissera à désirer tant que le problème n'aura pas été résolu.

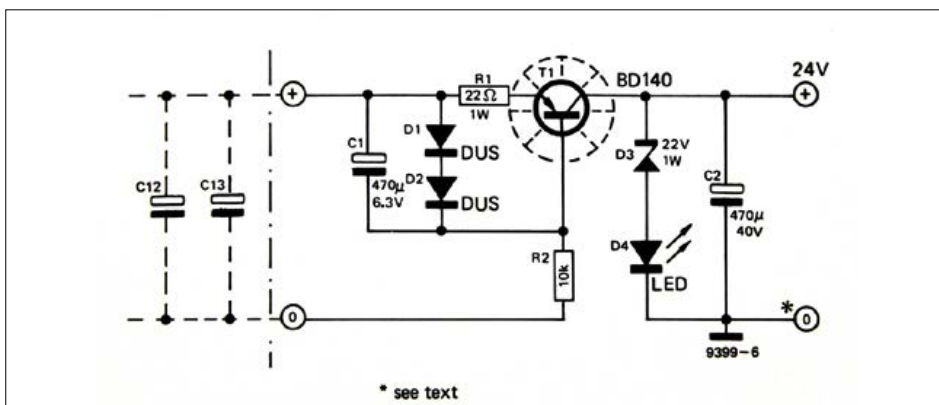


Figure 7. Schéma original de l'alimentation du Preco. Les DUS sont de retour !

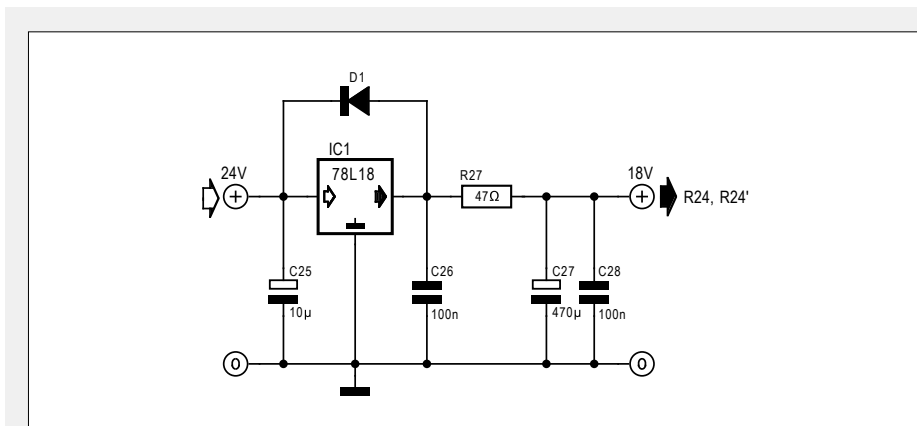


Figure 8. Suggestion de montage pour un régulateur abaisseur 24 à 18 V.

C25 = 10 μ F (tantale utilisé pour sa faible inductance, mais pour des raisons éthiques les condensateurs à feuilles d'aluminium sont désormais préférés ; il est certain qu'un condensateur moderne, à film, de 0,33 μ F conviendrait aussi)
C26 = 100 nF
C27 = 470 μ F, 25 V

C28 = 100 nF (non critique ; le prototype utilisait une plus grande valeur, mais on peut même se passer de ce composant)
R27 = 47 Ω
IC1 = μ A78L18
D1 = 1N4002 (ou n'importe quel composant similaire)

Le bruit interne du Preco n'est cependant pas affecté par un réglage de la largeur stéréo : tout ajustement effectué sur la télécommande n'a que peu d'effet. En tendant mieux l'oreille, je me suis aperçu que non seulement le bruit était fixe sur sa position stéréo, mais qu'il était au centre exact de l'image stéréo : le bruit était en mono !

Tout cela était bougrement surprenant : « mono » voulait dire que le même signal bruité se manifestait sur les deux canaux avec la même phase. Qu'une telle correspondance survînt par hasard entre les deux canaux pour le bruit (composé de fluctuations aléatoires de tension, par nature imprédictibles) était hautement improbable, pour ne pas dire impossible. Certes la théorie des probabilités nous dit qu'il est inévitable que de temps à autre deux signaux aléatoires indépendants aient la même forme, mais que cela se produise deux fois de suite (en laissant de côté ce qu'il faut entendre par « deux fois de suite ») est tout de même très improbable. À moins qu'une telle coïncidence persiste plusieurs microsecondes : là nous savons qu'il s'agit d'un signal de notre bonne étoile nous suggérant d'aller vérifier notre grille de Loto.

Bref j'étais face à un bruit monophonique provenant soit de quelque force mystérieuse affectant simultanément les deux canaux, ce qui était donc inconcevable, soit d'une unique source de bruit affectant les deux canaux. Les suspects se comptaient sur les doigts d'une main : la résistance R26, le condensateur C12, ou la zener Z1 ; tous trois font partie de l'alimentation (**fig. 7**), dont la sortie alimente le circuit de gain variable ainsi que le circuit de la télécommande pour les deux canaux.

Mon premier suspect aura été la zener, et j'ai finalement compris pourquoi jouer avec l'amplificateur réglable n'avait que peu d'effet sur le bruit : même sans ce circuit (et donc sans signal) la tension de bruit était acheminée directement vers la sortie du Preco via la résistance R24 et le condensateur C11 (et via R24' et C11'), et donc vers l'entrée de l'amplificateur de puissance.

J'aurais bien sûr pu me procurer une zener à faible bruit (il en existe), mais cela n'aurait pas résolu le problème des interférences induites entre l'alimentation et la sortie, et le bruit aurait encore été présent.

Quelques trucs pour ceux qui souhaiteront tester le Preco

- Si vous utilisez la télécommande, il est préférable de ne pas travailler sur la connexion reliant les deux circuits du Preco durant son fonctionnement. Le risque n'est pas de l'endommager, mais les niveaux de tension CC des connexions sont tels que par inadvertance vous pourriez envoyer une charge dans votre haut-parleur de graves, et même si le volume est sur zéro ça fait vraiment du bruit !
- Pour la même raison (et en conjonction avec la loi de Murphy), préférez pour la télécommande des connecteurs différents de ceux utilisés ailleurs dans le matériel.
- Pour une correspondance optimale entre l'entrée de la cellule magnétique et la cellule elle-même, on peut remplacer R5 et R5' de la carte par des 100 k Ω , et ajouter une prise audio supplémentaire en parallèle à chaque entrée (quel que soit le type de connecteur qu'elle utilise). On peut alors construire une fiche factice contenant une résistance accordée à une cellule individuelle, et l'insérer dans chaque nouvelle prise. Pour une meilleure correspondance (et un ajustement aisé), on peut aussi inclure un petit condensateur parallèle dans la fiche factice.

Le coupable découvert, j'ai étudié les remèdes possibles. Le circuit comportait déjà un condensateur de lissage censé réduire le bruit (C12), mais clairement il faisait mal son boulot. J'aurais pu augmenter sa valeur de façon sensible, mais cela n'aurait pas été une bonne idée : plus un condensateur placé en parallèle avec une source de bruit est gros et de meilleure qualité, plus les courants de lissage dus aux fluctuations aléatoires de tension qui le traversent sont intenses, ce qui aurait pu écourter la durée de vie de la zener.

Quelle autre solution avais-je pour obtenir 18 V à partir de 24 V ? La réponse est sur la **figure 8** : le régulateur 78L18. Le circuit d'origine s'en trouve grandement amélioré, même si le 78L18 fait consommer environ 2 mA de plus. (A priori ce n'est pas demain que les préamplificateurs auront des étiquettes-énergie comme les lave-linges, mais si ce jour arrive je me dis que le Preco pourrait en être exempté en tant qu'appareil de collection !) Le boîtier TO-92 peut en outre être placé sur le circuit imprimé existant avec son cortège de condensateurs chargés de supprimer les ondulations. Pris d'un élan perfectionniste, j'ai décidé de ne pas écouter immédiatement le résultat, mais d'encore améliorer l'ali-

mentation. Pour cela j'ai ajouté un filtre passe-bas composé de R27, C27 et C28. Comparé au circuit de la zener, ce filtre réduit plus avant le bruit de sortie du régulateur de tension, mais on peut bien sûr s'en passer. J'ai également ajouté la diode de protection D1, car une pointe intense peut facilement survenir dans la tension au secondaire lorsque la tension au primaire est appliquée ; dans certains cas cette pointe peut détruire l'autoprotection du régulateur de tension, une fonction par ailleurs très efficace dans toute autre situation. Il n'est pas difficile de comprendre qu'avec une résistance de 47 Ω le courant qui traverse R27 pourra aisément dépasser la capacité de sortie de 100 mA du 78L18, et donc le placer en mode de limitation de courant. Cela n'entraîne aucun risque pour le composant puisqu'il peut sans souci survivre à des contraintes répétées de ce type. Ce n'est pas un problème non plus pour ma réalisation, car avec le boîtier rétro que j'utilise l'ensemble de l'amplificateur est alimenté par une alimentation CA dont la tension de sortie n'augmente que lentement, ceci afin de reproduire la mise en température des circuits à tubes. Mais c'est une autre histoire... ◀

(150343 – version française : Hervé Moreau)

EST^D 2004

www.elektor.tv



Rétrotronique 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

drôles de drones

quelques modèles surprenants



Harry Baggen

Les quadricoptères et leurs congénères sont actuellement très populaires, pas seulement auprès des jeunes, mais aussi auprès des adultes. Ces petites merveilles volantes suscitent l'intérêt des électroniciens. Elles sont disponibles en plusieurs types et tailles, du simple jouet aux versions professionnelles. Dans cette rubrique, nous examinerons des modèles d'exception.

Autrefois, les jeunes (et leurs pères aussi) rêvaient de piloter un modèle réduit d'avion. Aujourd'hui, il semble plutôt que tout le monde veuille jouer avec un quadricoptère. On en trouve partout. Les magasins de jouets proposent des exemplaires très bon marché pour les enfants à côté de très nombreux modèles à des prix nettement supérieurs pour des acheteurs qui veulent prendre des photos - ou tourner des vidéos - depuis le ciel. Et nous ne parlons pas ici des appareils destinés à des utilisateurs professionnels ou aux militaires.

Il y a donc de nombreux modèles de drones dans toutes les catégories de prix, et dans toutes les tailles. Il existe des exemplaires dont la conception et les fonctions sont parfois très spéciales, nous allons en examiner plusieurs.

Drone pliable

Les drones - surtout les modèles les plus gros - ont un format peu pratique. Ils ne sont pas faciles à transporter à cause des supports de moteur (quatre ou plus). Même s'il est parfois possible d'enlever aisément les hélices, il reste les supports qui prennent encore beaucoup de place.



Certains fabricants ont résolu ce problème en créant un mécanisme pliable. On utilise souvent un mécanisme semblable à celui d'un parapluie, mais il en existe d'autres. La société Droidworx a développé un petit quadricoptère qui, une fois plié, ressemble à une boîte carrée noire [1]. Les supports de moteur se placent précisément dans les emplacements prévus dans la boîte, même les hélices peuvent être repliées sur elles-mêmes. À l'avant du boîtier se trouve un panneau avec plusieurs touches et une caméra HD. Le modèle *blu* peut être piloté via une commande à distance courante, un ordinateur ou un appareil sous Android ou iOS. Pour prendre des vues de vous-même, il y a une fonction autoportrait (*selfie*) qui envoie le quadricoptère voler derrière vous à une distance paramétrable. Le *blu* a l'air fragile à première vue du fait de sa forme, mais l'engin est réalisé avec des pièces en carbone et nylon, ce qui le rend très robuste. La puissance de calcul est fournie par un microcontrôleur à 32 bits STM32F7, en outre le drone contient de série un récepteur GPS rapide. Le *blu* coûte 500 \$ et il existe également un contrôleur Bluetooth sous la forme d'un bracelet.

En laisse

Pourquoi concevoir un quadricoptère relié à un fil ? C'est pourtant le cas du *Fotokite Phi*, un projet dont la production est financée sur Indiegogo [2]. Le *Fotokite Phi* plane attaché à un câble et vous suit sans que vous ne deviez le piloter via un ordi-
phone ou une commande à distance. La portée est de quelques

mètres et la poignée est une sorte de commande à distance qui permet de contrôler la hauteur et la rotation du quadricoptère. Le quadricoptère est facile à piloter et peut être manœuvré avec précision pour faire des photos à l'intérieur. C'est un peu bizarre de tenir un drone en laisse, mais il faut bien admettre que ce quadricoptère repose sur quelques idées très originales. La structure de cet engin est très pratique, les supports de moteur sont repliables, vous les ouvrez d'un geste et vous les verrouillez en tournant un bouton. Cet engin peut ainsi rentrer dans une boîte de la taille d'un thermos et peut facilement être transporté dans un sac. Une fixation est prévue sous l'appareil pour accueillir une caméra GoPro standard, sous différents angles.

Sur l'eau

L'*AquaDrone* est un quadricoptère conçu pour une application surprenante. Hélas ! La campagne de financement participatif sur Kickstarter n'a pas rapporté assez d'argent, mais l'inventeur veut quand même lancer la production de ce produit particulier. Il est intéressant de jeter un coup d'œil à cet engin. L'*AquaDrone* a été spécialement pensé pour les pêcheurs. Faites-le survoler l'étendue d'eau où vous voulez pêcher. L'engin est pourvu d'un boîtier amovible sur sa partie inférieure qui intègre un sonar. L'*AquaDrone* se pose sur l'eau et cherche des poissons à l'aide de son sonar. S'il en détecte, il envoie leur localisation à l'ordinateur du pêcheur pour qu'il sache où lancer sa canne. Le drone revient ensuite sagement près de son maître. Le compartiment sonar peut être enlevé pour y placer un petit crochet où l'on peut fixer une ligne. Il n'y a plus qu'à envoyer l'*AquaDrone* à l'endroit où l'on a repéré les poissons et y laisser tomber la ligne. N'est-ce pas le compagnon de rêve de tout pêcheur passionné de technologie ?

le drone en laisse vous suivra comme votre ombre



Intel Edison

L'*Eedu* (*Easy Educational Drone Unit*) est un quadricoptère spécialement conçu pour les projets pédagogiques [3]. Il se monte et se démonte facilement et l'électronique est étudiée pour vous permettre de faire immédiatement des expériences. La carte mère est équipée non seulement du rapide microcontrôleur ARM Cortex M7, mais aussi d'un module Intel Edison. Il y a de la place à bord pour des *shields* Arduino standards. Vous avez ainsi toute liberté pour concevoir vos propres combinaisons matérielles, logicielles et vos applications. La programmation est très simple du fait de l'utilisation de Forge, l'environnement de développement spécial robot (RDE). Les jeunes et les débutants pourront ainsi développer en un tournemain leurs propres programmes ou adapter des programmes existants à leurs besoins. Malheureusement, ce projet n'a pas atteint le montant espéré lors de la campagne de financement Kickstarter et on ne sait pas encore si la production de l'*Eedu* sera lancée.

Anti-drone

Du fait de cette multiplication des drones, de nombreuses personnes se sentent menacées dans leur intimité. C'est surtout aux États-

Unis que cette tendance conduit à créer des produits anti-drones destinés à les repousser ou à les dérégler. On trouve dans le commerce des appareils techniquement avancés visant à brouiller les connexions sans fil des drones. Certaines personnes optent pour une approche plus radicale. Une entreprise vend déjà des munitions [5] spéciales anti-drones pour les abattre... ◀

(150492 – version française : Eric Dusart)



Liens

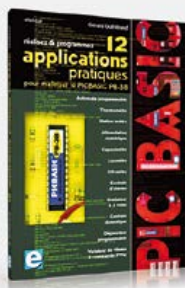
- [1] www.mydroidworx.org/blu.html
- [2] www.indiegogo.com/projects/fotokite-phi-tethered-uav-for-aerial-filming
- [3] <http://skyworksas.com/eedu/>
- [4] www.kickstarter.com/projects/1528442751/first-water-proof-drone-with-sonar-fish-finder
- [5] <http://snakerivershootingproducts.com/>

(re)découvrez la collection GUIHÉNEUF

Programmation PICBASIC

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

Réalisez et programmez 12 applications pratiques pour maîtriser le PICBASIC PB-3B



Les modules PICBASIC à base de microcontrôleurs PIC de Microchip sont simples à mettre en œuvre et à programmer en BASIC, ils permettent de développer rapidement des applications très diverses avec un minimum de connaissances et de moyens. Une source d'alimentation de 5 V et quelques composants suffisent pour leur donner vie. Grâce à leur simplicité, l'électronique numérique programmable est désormais à la portée de tous. Les douze applications pratiques du PICBASIC PB-3B de ce livre couvrent des domaines variés : la domotique, la protection des biens, la mesure, l'automatisation et l'électronique de puissance.

Les microcontrôleurs au service de l'écologie !

Eau de pluie : récupération & utilisation - Montages électroniques PICBASIC pour le recyclage des eaux de pluie

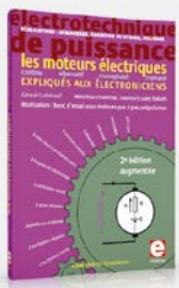


Ce livre aborde le thème de l'eau sous ses aspects quantitatif (cycle de l'eau, précipitations, prix) et qualitatif (traitement de l'eau potable, assainissement des eaux usées). Il détaille les différents éléments d'une installation de récupération des eaux de pluie : système aérien, cuve enterrée, pompe, filtres... Il contient six réalisations électroniques pour améliorer une installation domestique de gestion des eaux pluviales : mesure (pluviomètre à transmission haute fréquence, jauge numérique à ultrasons, compteur de consommation), contrôle de qualité (pH-mètre, conductivimètre) et automatisation (unité de pompe radiocommandée).

Moteurs

Réconcilier électroniciens et électrotechniciens autour des moteurs !

Electrotechnique de puissance : les moteurs électriques expliqués aux électroniciens - 2^e édition



L'électronique et l'électrotechnique fusionnent dans des applications industrielles ou domestiques qui mettent en œuvre des moteurs électriques. Dans ce livre, les électroniciens découvriront le monde des moteurs électriques tandis que les électrotechniciens admettront qu'électronique n'est pas obligatoirement synonyme de complexité. Les quatre chapitres s'appuient sur une information accessible et digeste : constitution, fonctionnement, caractéristiques, domaines d'utilisation des différents types de moteurs ainsi que des réalisations électroniques simples et concrètes : moteurs à courant continu ; moteurs à alimentation alternative monophasée ; moteur asynchrone triphasé ; moteurs sans balais.

Notre auteur Gérard Guihéneuf, titulaire d'un DUT Électrotechnique, a enseigné dans un lycée professionnel nantais (BEP Métiers de l'Électrotechnique et Bac Pro Électrotechnique Énergie et Équipements Communicants). La sortie de son nouveau livre *Domotique avec Zelio Logic* est l'occasion de (re)découvrir les autres ouvrages écrits par ce passionné d'électronique et d'informatique appliquées.

Énergies renouvelables

Énergies renouvelables : pour tout comprendre !

Comprendre et dimensionner les installations domestiques à énergies renouvelables

(disponible uniquement en version numérique)

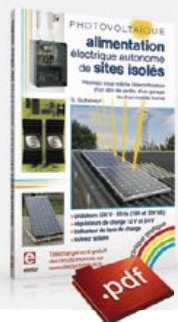


Les applications possibles des énergies renouvelables ne manquent pas. D'ailleurs nous sommes submergés de propositions techniques épatantes, mais pas forcément bien adaptées, même quand elles émanent de professionnels compétents. C'est pourquoi ce livre vous permettra d'acquérir assez de compétences (vocabulaire, principes physiques et méthodes de dimensionnement des matériels) pour évaluer la pertinence de ces propositions : production électrique photovoltaïque raccordée au réseau ; production électrique photovoltaïque en site isolé ; chauffe-eau solaire individuel (CESI) ; production électrique éolienne pour applications domestiques ; chauffage des locaux par géothermie ou par aérothermie ; chauffage des locaux par bois-énergie.

L'autonomie énergétique grâce à la production électrique photovoltaïque !

Photovoltaïque - alimentation électrique autonome de sites isolés - Réalisez vous-même l'électrification d'un abri de jardin, d'un garage ou d'un mobile home

(disponible uniquement en version numérique)



Ce livre constitue une réponse technique et pratique aux nombreuses questions qui se posent lors de l'électrification d'un site isolé ! Trois projets simples sont décrits dans un contexte domestique ; ils permettent d'électrifier soi-même, à partir de l'énergie photovoltaïque, trois sites ordinaires : un abri de jardin, un garage et un mobile-home. Des économies spectaculaires sont à la portée de l'électronicien qui montera et assemblera lui-même certains éléments essentiels, comme les régulateurs de charge, les onduleurs et autres suiveurs solaires mis en œuvre pour l'alimentation autonome d'un site isolé.

Automate programmable

Votre maison plus intelligente grâce à la domotique !

Domotique avec Zelio Logic - automate programmable en LADDER et FBD - étude, programme, réalisation, mise en service



Ce livre montre comment exploiter le module programmable compact Zelio Logic de SCHNEIDER, ses 12 entrées/sorties (4 entrées logiques, 4 entrées analogiques ou logiques et 4 sorties à contact). Le logiciel pour PC Zelio Soft 2 fourni avec ce module est un outil de programmation graphique en LADDER ou FBD. Les projets sont faciles à reproduire et à adapter à vos besoins : station de pompage domestique ; gestion multi-mode de l'éclairage d'une habitation ; gestion centralisée de l'éclairage d'une habitation ; gestionnaire de chauffage pour convecteurs ; centrale de chauffage biénergie à accumulation ; optimisation de la gestion du chauffage et de l'énergie ; alimentation secourue de circuits spécialisés ; centrale d'alarme filaire...



e-choppe

Retrouvez tous ces ouvrages
(sommaire, extrait...) ici :

www.elektor.fr/guiheneuf



est-ce e-éthique ?

Tessel Renzenbrink

« Levez la main s'il vous plaît : pensez-vous que notre profession devrait participer à l'élaboration des règles éthiques de l'électronique de demain ? »

La question a été posée lors de la conférence annuelle d'une organisation de professionnels de l'électronique. Parmi les trois cents représentants présents, à votre avis, combien ont levé la main ? Seulement cinq.

La question équivalait pourtant à demander à ces hommes d'affaires et décideurs : « Pensez-vous que nous devrions laisser les automobilistes programmer leur système de freinage automatique afin qu'ils décident eux-mêmes du comportement de leur véhicule face à une situation comme : *un chien traverse devant la voiture ET le freinage est dangereux pour les passagers* → freiner : OUI ou NON ? »

Pour Elektor, au moins 290 mains auraient dû se lever ! Car nous, concepteurs et fabricants, portons une réelle responsabilité en matière d'éthique dès lors qu'il s'agit d'électronique. Sans oublier le *business* proprement dit : toutes les discussions sur l'éthique mèneront d'une façon ou d'une autre à des innovations, à de nouvelles opportunités d'affaires et, enfin et surtout, il ne peut-être qu'enrichissant et amusant de discuter de la façon dont l'électronique peut changer le monde. C'est pourquoi nous créons une nouvelle rubrique à la fois en ligne et sur papier : « Elektor & éthique » ; les habituelles « nouvelles du monde d'Elektor » seront complétées ou parfois remplacées par ces informations éthico-électroniques. Contactez-moi (tessel.renzenbrink@eimworld.com) si vous avez des questions ou problèmes d'ordre éthique, je coordonne toutes les discussions relatives à ce sujet. ◀

(150491 – version française : Hervé Moreau)



hexadoku casse-tête pour elektorniciens

À votre retour du travail, une tâche vous attendait : découper et vider un potiron sans rien casser, c'est sportif. Maintenant vous avez bien mérité une petite pause. Installez-vous sur la table de la cuisine. Vous pourrez compléter votre grille tout en humant l'odeur de la soupe qui mijote. Régalez-vous !

Une grille hexadoku est composée de chiffres du système hexadécimal, de 0 à F. Remplissez le diagramme de 16 x 16 cases de telle façon que **tous** les chiffres hexadécimaux de 0 à F (0 à 9 et A à F) n'apparaissent **qu'une seule et unique fois** dans chaque rangée, colonne et carré de 4 x 4 cases (délimités par un filet gras).

Certains chiffres, déjà placés dans la grille, en définissent la situation de départ.

Pour participer, inutile de nous envoyer toute la grille, il suffit de nous donner **la série de chiffres** sur fond grisé.



Participez et gagnez ! Nous tirerons au sort **trois bonnes réponses** internationales reçues dans les délais ; leurs auteurs recevront chacun un chèque-cadeau d'une valeur de **50 €** à valoir dans l'e-shoppe d'Elektor. À vos crayons !

Envoyez votre réponse (les chiffres sur fond grisé) avant le **1^{er} décembre 2015** avec vos coordonnées par courrier électronique exclusivement à **hexadoku@elektor.fr**

Les gagnants

La solution de la grille du numéro de septembre est **6170C**

Les cinq bons Elektor d'une valeur de 50 € vont à : Marco **Retti** (Suisse), Martin **Heyer** (Allemagne), Marc **Moulin** (France), Gerald **Kreisler** (Allemagne), Frédéric **Hay** (France).

Bravo à tous les participants et félicitations aux gagnants !

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

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

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.



**novembre 2015
à décembre 2015**

Novembre 2015

♦ **Batimat & Interclima+Elec**
02 au 06/11 – Paris
www.interclimaelec.com & www.batimat.com

♦ **Journées Nationales de la Domotique**
16 au 29/11 – dans toute la France
www.promotelec.com



♦ **SATIS**
(industrie des technologies audiovisuelles)
17 au 19/11 – Paris
www.satis-expo.com

♦ **Salon Cartes Secure Connexion**
17 au 19/11 – Paris
fr.cartes.com

♦ **MIDEST** (sous-traitance industrielle)
17 au 20/11 – Paris
www.midest.com

♦ **Educathec/Educative**
18 au 20/11 – Paris
www.educathec-educative.com

♦ **Semaine Européenne de la Réduction des Déchets**
21 au 29/11 – dans toute la France
www.serd.ademe.fr



♦ **Energaïa** (forum des énergies)
26 au 26/11 – Montpellier
energaia.fr

Décembre 2015

♦ **Salon de la moto**
01 au 06/12 – Paris
lesalondelamoto.com

♦ **COP21/CMP11** - grande conférence sur le climat
04 au 10/12 – Paris
www.cop21.gouv.fr/fr

♦ **Assises de l'embarqué**
07/12 – Paris
www.assisesdelembarque.fr

♦ **ForumLED**
07 au 08/12 – Lyon
www.forumled.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.



OSCILLOSCOPES PC

Petit Prix



- Bande Passante 10 MHz à 200 MHz
- Échantillonnage de 100Mé à 1Gé/s
- Résolution 8 bits (améliorée à 12 bits)
- 8 to 48 kS buffer memory
- Mémoire tampon de 8 à 48 kS
- Dès 99 €

Signaux Mixtes



- 2 ou 4 voies analogiques + 16 logiques
- Bande passante de 50 à 200 MHz
- Résolution 8 bits (améliorée à 12 bits)
- Mémoire tampon de 64 à 512 Mé.
- Alimentation par USB ou bloc secteur
- Dès 659 €

8 voies



- Bande Passante 20 MHz
- Échantillonnage 80 Mé/s
- Résolution 12 bits (améliorée à 16 bits)
- Mémoire tampon 256 Mé.
- Alimentation par USB
- Pour 1845 € seulement

Résolution Flexible



- 8, 12, 14, 15, et 16 bits dans un même appareil
- Bande passante de 60 à 200 MHz
- Échantillonnage à 250 Mé/s to 1 Gé/s
- Mémoire tampon de 8 à 512 Mé
- Alimentation par USB ou bloc secteur
- Dès 929 €

Mémoire 2Giga-échantillons



- Bande passante de 250 MHz à 1GHz
- Échantillonnage à 5 Gé/s
- Résolution 8 bits (améliorée à 12 bits)
- Mémoire tampon de 256 Me à 2Gé
- Alimentation par bloc secteur
- Dès 2645 €

Sampling 20 GHz



- Bande passante du continu à 20 GHz
- Temps de montée 17.5 pS
- 16 bits, plage dynamique de 60 dB
- Alimentation par bloc secteur
- Générateur de signaux, reconstruction d'horloge, TDR/TDT diff.
- Dès 12,035 €

Logiciel complet inclus en standard avec décodage et analyse de bus série (CAN, LIN, RS232, I2C, I2S, SPI, FlexRay), mémoire segmentée, test de masque, analyse de spectre, kit de développement logiciel (SDK) tout en standard avec mises à jour gratuites.

Cinq ans de garantie pour les oscilloscopes temps-réel.

Deux ans de garantie pour les oscilloscopes à sampling.

www.picotech.com/PS456

Une qualité professionnelle à prix discount!

reichelt.fr

elektronik

- ✓ Plus de 45 ans d'expérience
- ✓ Envoi en 24 heures
- ✓ Plus de 50 000 produits

KONTAKT
CHEMIE

Les langues de notre boutique:  



Druckluft 67 2,983 € / 100 ml

Número de commande **14,20**
KONTAKT 334 400 ml

Kälte 75 2,983 € / 100 ml

Spray réfrigérant jusqu'à -52 °C
Número de commande **14,20**
KONTAKT 317 400 ml

Kontakt 60 2,121 € / 100 ml

Nettoyant de contact dissolvant les huiles
Número de commande **10,10**
KONTAKT 203 400 ml

Kontakt WL 2,189 € / 100 ml

Nettoyant universel
Número de commande **5,30**
KONTAKT 208 200 ml

Sprühöl 88 2,605 € / 100 ml

Huile mécanique fine
Número de commande **6,20**
KONTAKT 230 200 ml

Solvent 50 2,563 € / 100 ml

Dissolvant à étiquettes
Número de commande **6,10**
KONTAKT 244 200 ml

Screen TFT 2,478 € / 100 ml

Screen Cleaner
Número de commande **5,90**
KONTAKT 501 200 ml

S'ABONNER MAINTENANT !

Newsletter

Recevez chaque semaine les toutes dernières informations

- ✓ Nouveautés
- ✓ Les meilleures offres
- ✓ Réductions des prix



Prix du jour ! Prix à la date du : 22.09.2015

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

Un ensemble d'outils très intéressant !

en sacoche pliable de grande qualité, 22 pièces

Clés à cliquets et vis au format compact. Par la sélection minutieuse des outils les plus importants, vous êtes, grâce à cette combinaison, équipé au mieux pour la plupart des opérations à effectuer

- 1 cliquet Zyklop mini, 7 noix (8, 10, 12 et 13), 1 rallonge avec douille à rotation rapide
- 1 support manuel avec mandrin à serrage rapide pour embouts de 1/4", 4 embouts à fente en croix, 1 embout à fente, 5 embouts TORX, 4 embouts hexagonaux
- Sacoche pliable avec fixation à la ceinture

Número de commande WERA KK Z MINI

Edition limitée

49,00

Lame Lasertip Wera

Grâce à un traitement Laser-TIP, il y a à la pointe une surface de micro-précision avec des trempes superficielles allant jusqu'à 1000 Vickers qui permet à la forme de s'incruster parfaitement dans la tête de vis.

Fente
160 i VDE
1x 0,4x2,5x80,
1x 0,6x3,5x100,
1x 0,4x2,5x80,
1x 1,6x3,5x100,

Fente en croix
162 i PH VDE
1x PH 1x80,
1x PH 2x100
Contrôleur de tension
247 1x 0,5x3,0x75

Ensemble tournevis plats+cruciformes

Número de commande
WERA 6147

29,95

Ensemble 7 pièces + rack



Wera Tool-Check

Ensemble avec 1 support d'embout Rapidaptor, 1 clé à cliquet à embout Wera

- 28 embouts
- 1 adaptateur
- 7 noix



Número de commande
WERA 05056490001

79,95

Commander maintenant! www.reichelt.fr

Assistance téléphonique en anglais : +49 (0)4422 955-360

Modes de paiement internationaux:

SOFORT
ÜBERWEISUNG

VISA

MasterCard

PayPal