



nouvel LCR-mètre

de 50 Hz à 2 MHz

par Jean-Jacques Aubry

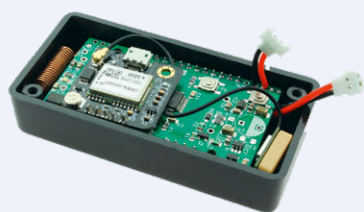
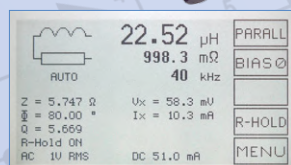
p.6

Soutenez
ce projet sur
Elektor LABS

dans ce numéro :

- > sonde de courant différentielle pour oscillo
- > entretien : apprentissage automatique : un bel avenir
- > bougie de Noël électronique
- > greffons et modules de KiCAD
- > comment bien gérer mes batteries ?
- > automates finis en assembleur et en C sur PIC à 8 bits
- > multitâche en pratique avec l'ESP32
- > applications mobiles pour Android et iOS
- > Fritzing : graphisme d'implantation sur plaques d'essais
- > conception de filtres analogiques actifs

et beaucoup d'autres !



balise GPS LoRa Matériel et logiciel libres et ouverts

p.98



banc d'essai par Cyrob
Philippe Demerliac teste le
générateur DDS Rigol DG4162

p.51

ISSN 0181-7450

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

L 19624 - 486 - F: 15,50 € - RD





NOTRE GAMME PAR DES TECHNICIENS POUR DES TECHNICIENS

The best part of your project: www.reichelt.com/assortiment

Uniquement le meilleur pour vous - provenant de plus de 900 marques.

Nos responsables produits sont employés par Reichelt depuis de nombreuses années et connaissent les exigences de nos clients. Ils rassemblent une large gamme de produits de qualité, à la fois parfaits pour les besoins dans les domaines de la recherche et du développement, la maintenance, l'infrastructure informatique et la production en petites séries et adaptés pour les fabricants.

Weller WE1010

La station à souder la plus puissante de sa catégorie

Station à souder ESD sécurisée avec stabilité de température élevée et navigation intuitive. La fonction de veille avec retrait automatique protège l'outil et l'environnement.

- Puissante : 70 W
- Conviviale : écran LCD clair
- Économique : faibles coûts d'exploitation



BESTSELLER

N° de commande : WELER WE1010

170,87
(142,39)



Weller®

Découvrez en ligne les
équipements pour la soudure
pour chaque application !



Découvrir maintenant ►
www.reichelt.com/soudure

DES SOLUTIONS À MOINDRE COÛT

**REPRENDRE L'ACTIVITÉ
DE MANIÈRE FLEXIBLE
ET SÛRE**

Découvrir maintenant ►



Types de paiement :



PRIX DU JOUR! Prix à la date du: 8. 10. 2020

- Excellent rapport qualité prix
- Plus de 110 000 produits sélectionnés

- Livraison fiable - depuis
l'Allemagne dans le monde entier

Assistance téléphonique: +33 97 518 03 04

www.reichelt.com

reichelt
elektronik – Tirer le meilleur parti de votre projet

Les réglementations légales en matière de résiliation sont applicables. Tous les prix sont indiqués en € TVA légale incluse, frais d'envoi pour l'ensemble du panier en sus. Seules nos CGV sont applicables (sur le site <https://rhc.it/CG-FR> ou sur demande). Semblables aux illustrations. Sous réserve de coquilles, d'erreurs et de modifications de prix. reichelt elektronik GmbH & Co. KG, Elektronikring 1, 26452 Sande (Allemagne), tél. +33 97 518 03 04

Elektor est édité par :

PUBLITRONIC SARL

c/o Regus Roissy CDG

1, rue de la Haye

BP 12910

FR - 95731 Roissy CDG Cedex

Pour toutes vos questions :

service@elektor.fr

www.elektor.fr | www.elektormagazine.fr

Banque ABN AMRO : Paris

IBAN : FR76 1873 9000 0100 2007 9702 603

BIC : ABNAFRPP

Publicité :

Margriet Debeij

Tél. : +49 (0)241 955 09 174

margriet.debeij@elektor.com

DROITS D'AUTEUR :

© 2020 Elektor International Media B.V.

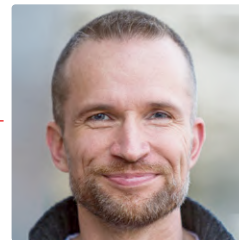
Toute reproduction ou représentation intégrale ou partielle, par quelque procédé que ce soit, des pages publiées dans la présente publication, faite sans l'autorisation de l'éditeur est illicite et constitue une contrefaçon. Seules sont autorisées, d'une part, les reproductions strictement réservées 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
Pijper Media - Groningen
Distribué en France par M.L.P.
et en Belgique par A.M.P.

Jens Nickel

rédacteur en chef d'Elektor Magazine



comprendre et construire

Le maître projet de ce numéro est un LCR-mètre. En 2013, un projet similaire, du même auteur, Jean-Jacques Aubry, publié dans Elektor en cinq ou six parties, avait connu un vif succès. Le kit de ce projet ambitieux avait fait beaucoup d'heureux. Entre-temps, de puissants appareils de mesure sont apparus, venus d'Asie, pour des budgets d'entrée de gamme comme on dit pudiquement. Voyez les nombreux oscillos proposés dans l'e-choppe d'Elektor ! Comme nous ignorons la proportion de lecteurs qui en 2021 souhaitent réaliser de grands projets comme celui-là, Elektor laissera ses lecteurs décider eux-mêmes si nous devons proposer un kit. Faites-nous part de votre intérêt (sans engagement formel) en vous faisant connaître sur la page www.elektor.com/lcr. Avant même la publication des articles sur le projet proprement dit (l'un dans ce numéro, le 2e dans le suivant), il y avait plus de 100 lecteurs intéressés. L'objectif de 150 engagements pourrait déjà être atteint avec cette édition. Ils seront récompensés chacun par une réduction de 100 € si la campagne est couronnée de succès.

Pour la rédaction, qui planche sur le contenu des magazines de l'an prochain, la question revient sans cesse de savoir comment encourager la pratique de l'électronique. En termes de coûts, la rentabilité est de moins en moins un motif convaincant. J'en veux pour preuve un autre grand projet de ce numéro : le tracker GPS qui transmet sa position sur des réseaux LoRa tels TheThingsNetwork. Ça s'achète désormais pour quelques euros ; certes, on n'apprend rien, mais que doit choisir l'électronicien passionné ? Je reste partisan de la publication de tels projets. Vous les réalisez, vous obtenez un système transparent, pour lequel nous vous offrons les fichiers de CAO, que vous modifierez et modulerez comme bon vous semble. L'article regorge d'informations précises et précieuses. Aussi pour ceux qui n'ont ni le temps ni l'envie de manier le fer à souder. Ils découvriront ce que signifient des termes tels que "clé de session" et "adresse de l'appareil".

C'est précisément ce mélange de théorie et de pratique qui distingue Elektor, et nous y resterons fidèles en 2021 !

Bonne lecture.

notre équipe



Rédacteur en chef :

Jens Nickel

Rédaction :

Eric Bogers, Jan Buiting, Rolf Gerstendorf,
Denis Meyer (traduction), Thomas Scherer, Clemens Valens

Service aux lecteurs :

Ralf Schmiedel

Correcteur technique :

Malte Fischer

Laboratoire :

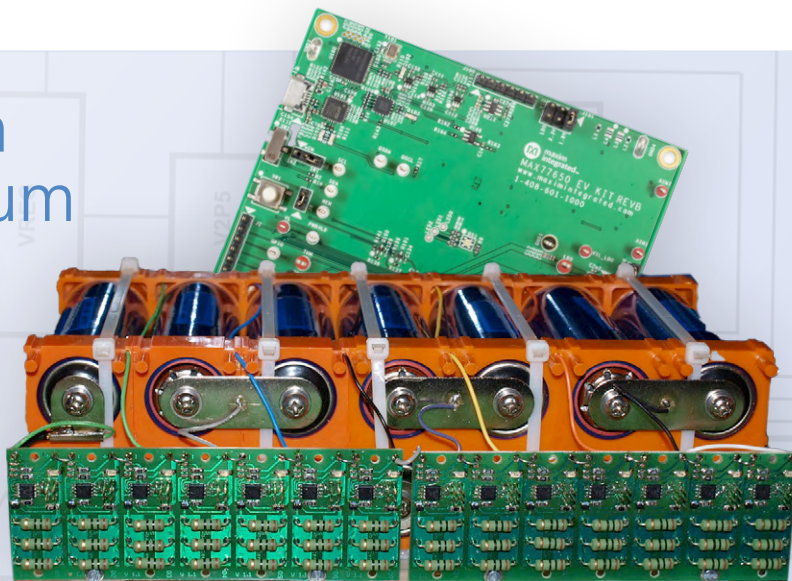
Mathias Claussen, Ton Giesberts, Luc Lemmens,
Clemens Valens (responsable), Jan Visser

Maquette :

Giel Dols, Harmen Heida

Qui veut aller loin ménage son lithium

25



rubriques

3 édit

30 démarrer en électronique

L'éternel recommencement

32 petits circuits

Les pépites d'Elektor

48 bureau d'études – Zone D

Trucs & Astuces, bonnes pratiques et info utile

74 rétronique : boucle d'interface Hewlett-Packard

Connecter le monde (en 1981) !

90 sur le vif

Professions en déficit de main-d'œuvre

92 5G : qui façonne qui, les infrastructures ou la société ?

La question n'est pas anodine, les conséquences non plus

114 hexadoku

casse-tête pour elektorniciens

contexte

25 comment bien gérer mes batteries

Qui veut aller loin ménage son Lithium

34 Fritzing : graphisme d'implantation sur plaques d'essais

Initiation en douceur

38 conception de filtres analogiques (2)

Filtres actifs

51 banc d'essai : oscillo à 4 voies Rigol DS1054Z

62 banc d'essai : générateur DDS Rigol DG4162

66 greffons et modules de KiCAD

78 apprentissage automatique : un bel avenir

Entretien avec Daniel Situnayake

82 applications mobiles pour Android et iOS

À partir d'un moule commun

94 multitâche en pratique avec l'ESP32 (5)

107 je programme des automates finis

En assembleur et en C sur des PIC à 8 bits

projets

6 nouveau LCR-mètre de 50 Hz à 2 MHz

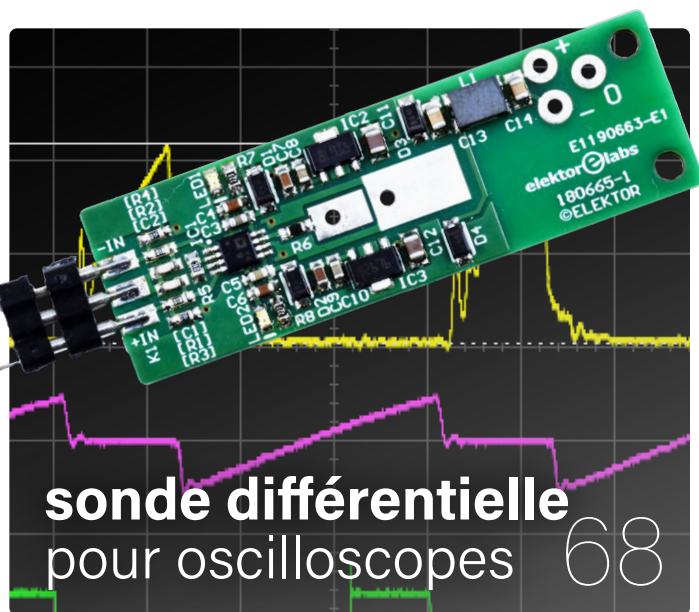
Pont de mesure d'impédance automatique pour mesurer la résistance, la capacité et l'inductance des composants de 10 mΩ à 100 MΩ

20 connectez votre sonnette à l'internet des objets

avec Home Assistant et ESPHome

46 compteur de tours de piste

Passionné d'électronique et grand-père



sonde différentielle
pour oscilloscopes 68



34



nouveau LCR-mètre
de 50 Hz à 2 MHz 6

bientôt dans ces pages

Elektor janvier-février 2021

- > générateur sinusoïdal I2S à 32 bits
- > caméra thermique à faire soi-même
- > analyseur de puissance
- > balises Bluetooth en pratique
- > capteur de particules
- > domotique avec Home Assistant (3)
- > FFT sur Maixduino
- > initiation à la programmation orientée objets
- ... et bien davantage!

Sous réserve de modifications

Le numéro de janvier-février d'Elektor paraîtra le 7 janvier 2021.

54 bougie de Noël électronique

Soufflez-la comme une vraie !

58 générateur sinusoïdal accordable à tubes

Passé mais pas dépassé

68 sonde de courant différentielle 2.0

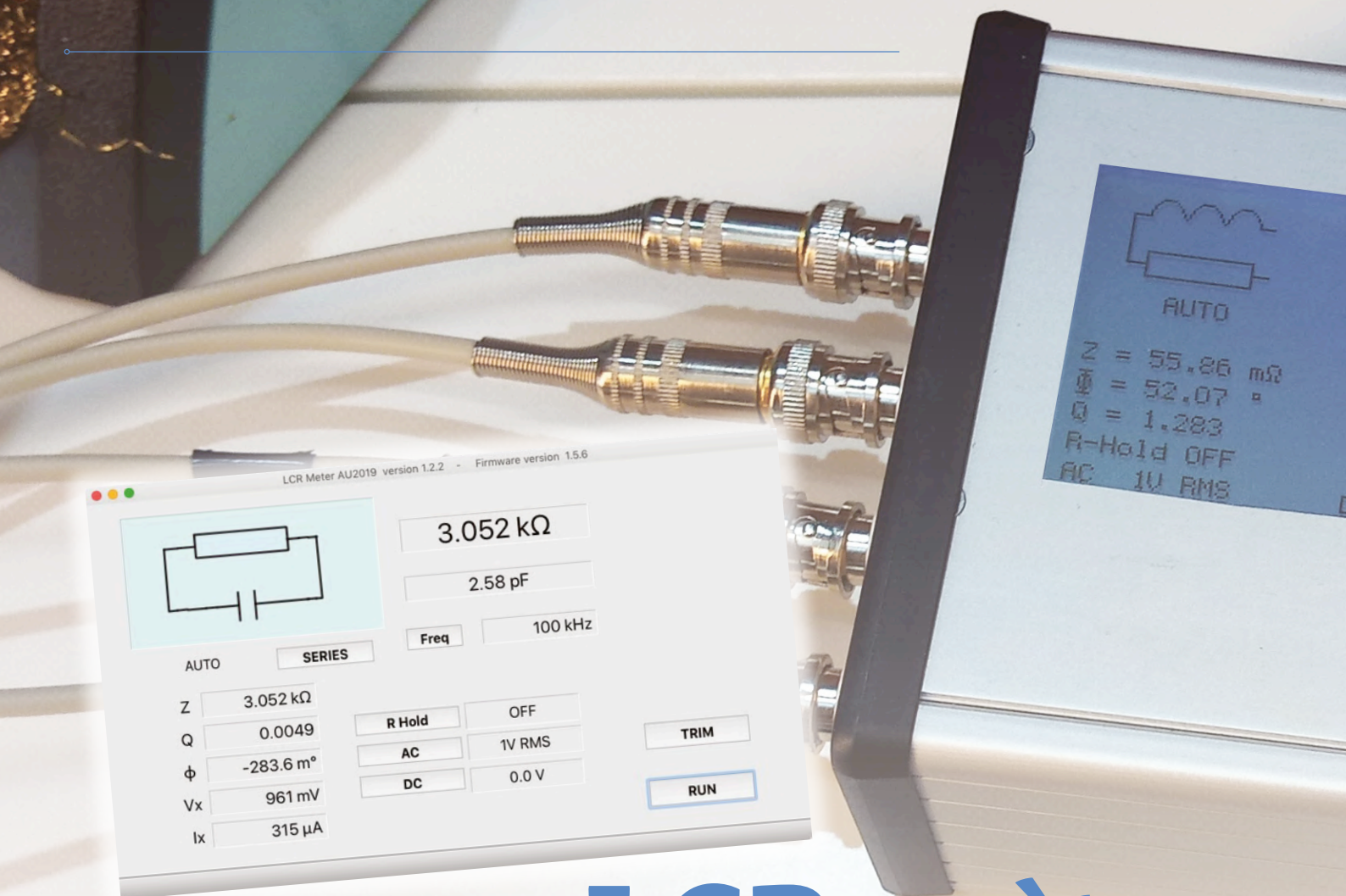
L'oscillo pour mesurer l'intensité

98 balise GPS LoRa

Matériel et logiciel libres et ouverts

Découvrez le site
Elektor Labs





nouveau LCR-mètre

50 Hz - 2 MHz

pont de mesure d'impédance automatique pour mesurer la résistance, la capacité et l'inductance des composants ayant une impédance de 10 mΩ à 100 MΩ

Jean-Jacques Aubry (France)

Il y a plus de 7 ans, Elektor a publié mon LCR-mètre de 500 ppm. Depuis lors, j'ai conçu un tout nouveau LCR-mètre, en sacrifiant l'extrême précision au profit de fonctions étendues avec : fréquence de test de 50 Hz à 2 MHz, 4 tensions de test possibles (100 mV, 200 mV, 500 mV et 1 Vrms), et une polarisation DC supplémentaire jusqu'à 5 V pour les condensateurs et 50 mA pour les inductances.



INFOS SUR LE PROJET

Mots-clés

mesure, kit

Niveau

débutant – connaisseur – expert

Temps

1 h

Outils

outillage standard de l'électronicien
tournevis HF

Coût

799€ (kickstarter: 699€)
voir www.elektor.com/lcr

Ceci n'est nullement une reprise de mon projet publié par Elektor en 2013. Il ne s'agit ni d'une mise à jour ni d'une évolution, l'instrument a été entièrement repensé, il est complètement différent. Une attention particulière est accordée à la facilité de mise en œuvre (étalonnage) et d'utilisation de ce nouvel appareil. Un codeur rotatif est utilisé pour naviguer dans le menu et pour changer la fréquence.

Informations générales

Ce LCR-mètre est un pont de mesure d'impédance automatique. Il mesure la résistance, la capacité et l'inductance de composants d'une impédance de 10 mΩ à 100 MΩ, et effectue des mesures à une fréquence choisie entre 50 Hz et 2 MHz.

Deux configurations sont possibles :

- Unité autonome (carte mère + extension d'affichage) qui fonctionne sans PC, avec une alimentation externe de 5 V (via un connecteur USB qui peut également être connecté à un ordinateur pour l'alimentation). C'est la configuration recommandée.
- Carte principale sans extension, qui ne fonctionne que connectée par USB à un ordinateur exécutant le programme utilisateur. Ce programme, développé à partir des bibliothèques Qt [3], a été testé sous Windows 7 et MacOSX.

Une extension Bluetooth pour la connexion à un téléphone tactile utilisé comme interface graphique est en cours d'étude.

Principe de mesure

L'impédance (Z) est un paramètre important pour caractériser les composants électroniques passifs (résistance, condensateur, inductance). C'est un nombre complexe, représentable par une partie réelle (R) et une partie imaginaire (X), tel que $Z = R + jX$, ou sous une forme polaire par le module de son impédance et le déphasage entre tension et courant : $|Z| \angle \theta$. Pour déterminer une impédance, il faut donc mesurer au moins deux valeurs (en grandeur et en phase), généralement la tension aux bornes du composant et le courant qui le traverse. Notre LCR-mètre utilise la méthode du pont auto équilibré avec l'utilisation d'un simple amplificateur opérationnel pour le convertisseur courant/tension (I-V) (fig. 1).

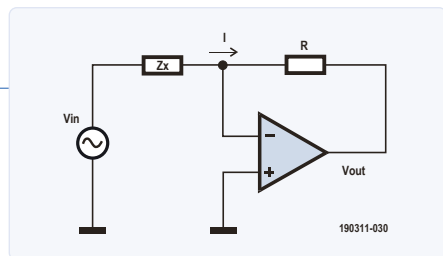


Figure 1. La méthode du pont autoéquilibré utilise un simple amplificateur opérationnel pour le convertisseur courant/tension.

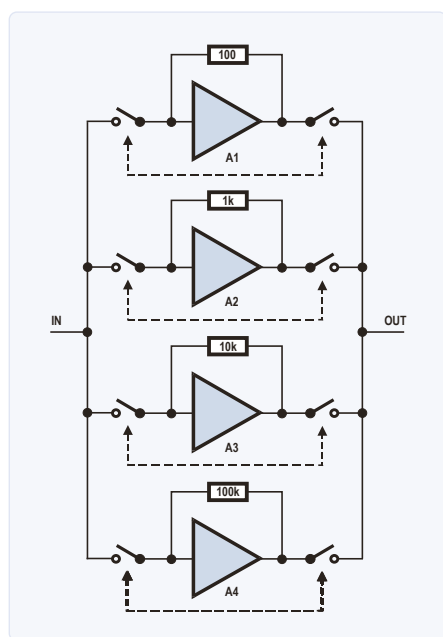


Figure 2. Quatre étages I-V, un pour chaque plage de mesure.

Cette méthode simple permet d'obtenir une bonne précision sur les mesures tout en maintenant un coût raisonnable. Son principal inconvénient est une gamme de fréquences limitée dans les fréquences

élevées par les performances de l'amplificateur opérationnel utilisé.

Pour obtenir une large gamme de mesure d'impédance (de quelques dizaines de mΩ à plus de 100 MΩ), il est nécessaire de commuter la résistance de précision (R) utilisée dans le convertisseur I-V. Malheureusement, les commutateurs analogiques intégrés courants (tels que le 74HC4052) introduisent des éléments parasites (principalement des capacités) qui limitent les performances à haute fréquence.

C'est pourquoi la plupart des instruments similaires ont leur gamme de fréquences limitée à 100 ou 200 kHz.

Il est cependant possible, par une conception originale et le choix de composants performants, de pousser **la limite de fréquence haute à 2 MHz** sans coûts excessifs, tout en conservant la simplicité de la réalisation.

La solution retenue (fig. 2) est de ne pas commuter les 4 résistances de mesure (donc capacités parasites réduites au minimum) mais d'avoir 4 couples **Amplificateur + Résistance**, sélectionnés chacun selon l'impédance à mesurer.

Les amplificateurs opérationnels utilisés (des AD8099 d'Analog Devices) présentent une fréquence de coupure d'environ 200 MHz pour une tension de sortie de 2 V c-c, et possèdent la commande d'inhibition nécessaire. Les commutateurs sont des PhotoMOS fabriqués par Panasonic qui présentent un très bas produit (Résistance ON x capacité de sortie). Un autre point important est le choix de la manière de produire la fréquence de test. Il est facile, et peu coûteux aujourd'hui, d'utiliser des composants de synthèse directe de fréquence (DDS), avec l'avantage d'obtenir n'importe quelle fréquence dans la gamme

50 Hz / 2 MHz retenue pour cet appareil. Il est, de plus, facile de produire pour le détecteur synchrone un signal de même fréquence, mais à phase relative variable, grâce à un second circuit DDS synchronisé sur le premier.

Schéma fonctionnel

Le schéma de la **figure 3** montre l'organisation du matériel du LCR-mètre et les liens entre les sections. Chaque section sera abordée dans cet article.

L'interface utilisateur occupe un circuit imprimé séparé, avec quelques composants réduits au strict minimum :

- > afficheur graphique LCD 240 x 128 points
- > 5 boutons poussoirs
- > codeur rotatif avec contact auxiliaire

Ce n'est pas recommandé, mais vous pouvez omettre cette carte d'extension et commander le LCR-mètre en le connectant à un ordinateur via l'interface USB, à l'aide d'un programme PC approprié. Pour l'instant, mon programme PC n'offre pas les mêmes fonctions que la version autonome. Je pense qu'un appareil de mesure autonome reste le meilleur choix. La manipulation de boutons physiques est plus facile, plus rapide ! À chaque mise à jour du système d'exploitation du PC, il n'est pas nécessaire de mettre à jour aussi le programme PC.

Pour alimenter le LCR-mètre à travers son connecteur mini-USB, vous avez le choix entre un chargeur de téléphone, une banque d'alimentation externe de 5 V (pour téléphone) ou connecter le LCR-mètre à un ordinateur (cette connexion facilite aussi la mise à jour du micrologiciel (*firmware*)).

Description du circuit de la carte principale

Entrées analogiques (fig.4)

La mesure est effectuée avec une configuration à 5 bornes [2] pour minimiser l'influence des câbles de test. Les connecteurs J4 (High Drive) et J7 (Low Drive) alimentent le DUT (*Device Under Test*) tandis que les connecteurs J5 (High Sense) et J6 (Low Sense) permettent de mesurer la tension au plus près du composant. Si une tension de polarisation est appliquée, les bornes positives sont à J4/J5. Les relais à semi-conducteurs *PhotoMOS* U54 à U57, qui relient les amplificateurs opérationnels U9 à U12, ayant une résistance ON d'environ 1 Ω, la tension doit être mesurée aux bornes du DUT et non par rapport à la masse. C'est la tâche de l'amplificateur différentiel construit autour de U7A, U7B, U7C. Son taux de réjection en mode commun

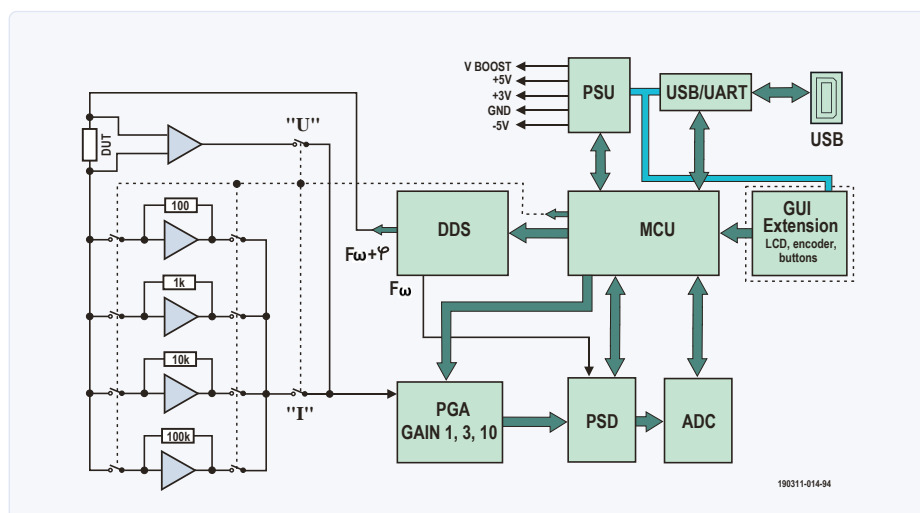


Figure 3. Les parties du LCR-mètre et la façon dont elles sont interconnectées

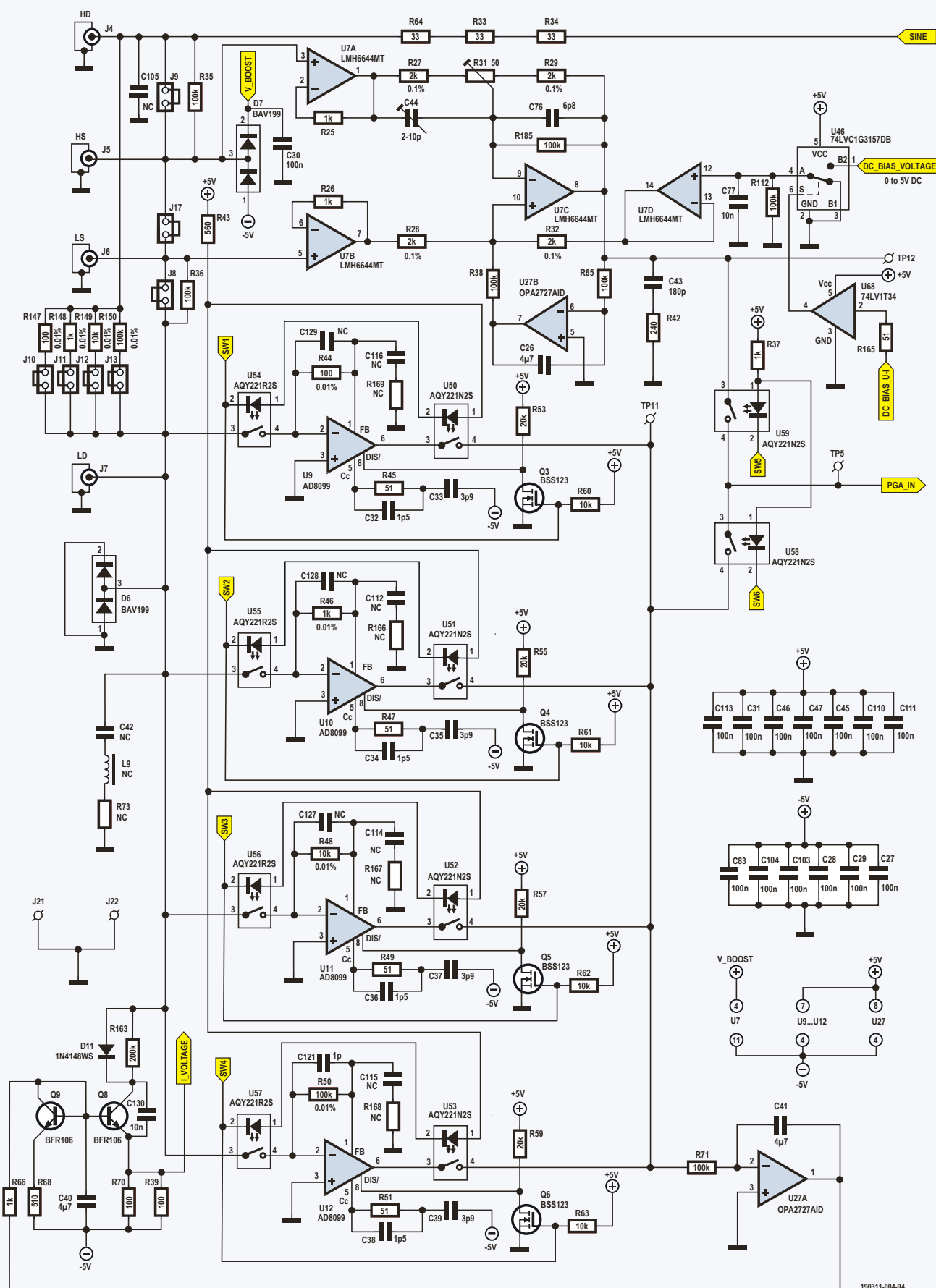


Figure 4. Les étages d'entrée analogiques. Quatre plages d'impédance peuvent être sélectionnées par le microcontrôleur.

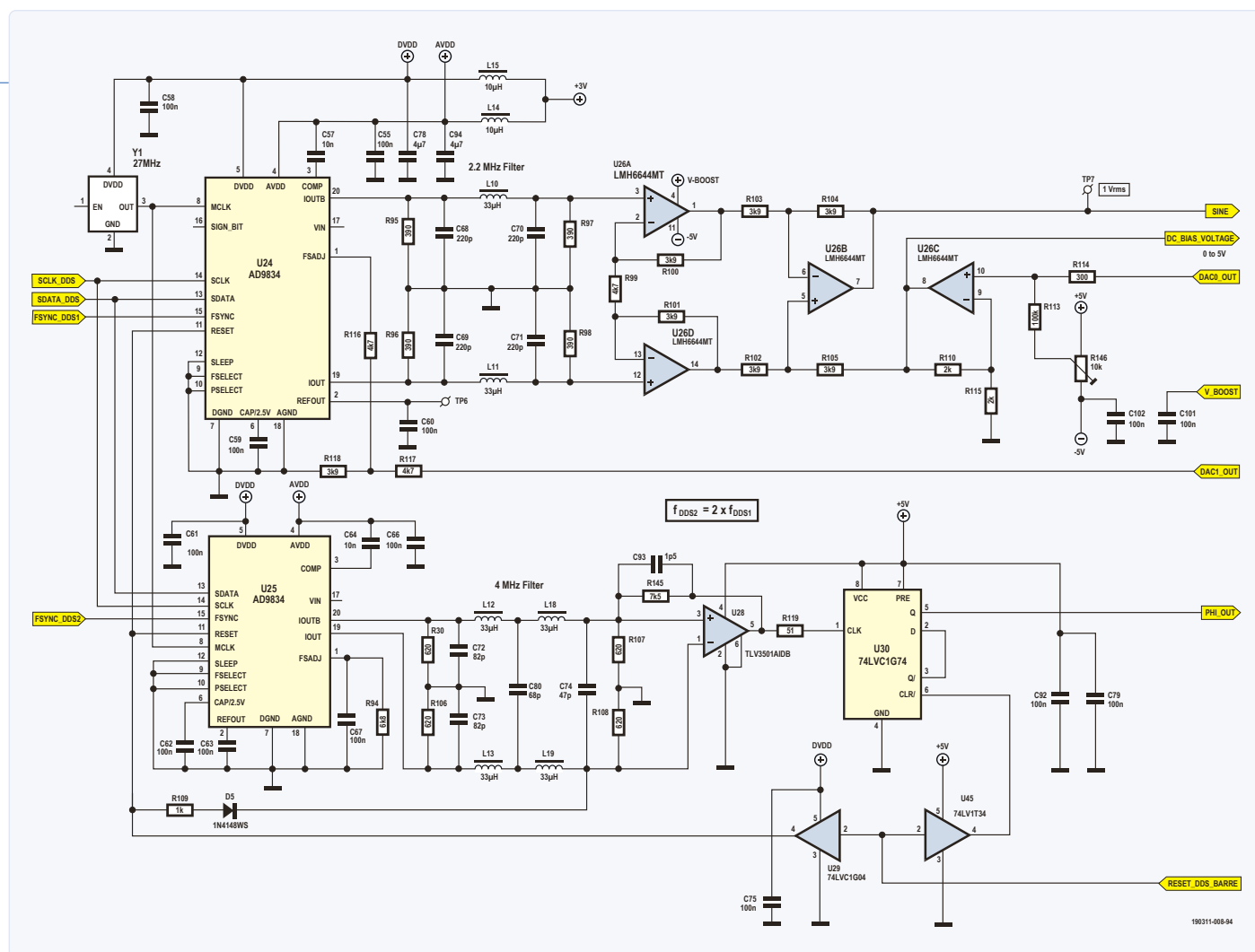


Figure 5. Deux circuits intégrés de synthèse numérique directe produisent les signaux de test.

(CMRR) doit être élevé sur toute la gamme de fréquences du LCR-mètre. Un réglage est donc nécessaire, d'abord par R31 pour les fréquences basses (10 kHz) puis par C44 pour les fréquences hautes (1 MHz).

L'intégrateur construit autour de U27B permet un réglage automatique de l'offset (décalage). Selon la plage de mesure, un des amplificateurs opérationnels U9 à U12 est sélectionné. Sa broche DIS/ doit être au niveau haut et les relais PhotoMOS associés (entrée par U54 à U57 et sortie par U50 à U53) doivent être activés par un des signaux SW1 à SW4 (niveau bas) de l'unité MCU.

Les signaux SW5 (commande U59) et SW6 (commande U58) envoient vers le circuit de mesure soit la tension de sortie U7C (tension DUT) soit la tension de sortie du convertisseur I-V (image du courant à travers le DUT). L'application de la polarisation continue sur le DUT est effectuée au niveau du générateur sinusoïdal par un décalage variant de 0 à 5 V. Comme une résistance de 100 Ω est mise en série avec la sortie de ce générateur, elle correspondra également à un courant de 0 à 50 mA si le DUT (inductif) a une faible résis-

tance série. Pour la tension aux bornes d'un condensateur, le suiveur U7D effectue une précompensation au niveau de l'amplificateur différentiel. Pour un courant à travers une inductance, c'est la source de courant continu formée autour de Q8 et Q9 qui l'absorbe. L'intégrateur construit avec U27A assure la présence en sortie du convertisseur I-V une tension continue proche de 0 V.

Pour l'étalonnage, des résistances de précision **identiques** à celles utilisées pour les convertisseurs I-V peuvent être connectées à l'aide d'un cavalier (J10 à J13).

Générateur d'ondes sinusoïdales (fig. 5)

Le générateur est construit autour du circuit intégré U24, un DDS AD9834 d'Analog Devices. Son horloge de 27 MHz vient de l'oscillateur Y1.

Sa sortie (deux signaux en opposition de phase) est filtrée puis amplifiée par U26A, U26B et U26D. L'amplificateur U26C permet d'appliquer une tension de décalage. Sans cette dernière, le potentiomètre R146 permet de régler le décalage du signal sinusoïdal (mesuré sur TP7).

Le détecteur de phase utilisé nécessite un signal parfaitement carré, à la même fréquence que le générateur, mais dont on puisse faire varier la phase relative. Ceci est obtenu par le circuit U25 (également un DDS AD9834) fonctionnant à une fréquence double et suivi par un comparateur rapide (U28) qui pilote un diviseur par 2 (bascule U30).

PGA (amplificateur à gain programmable) (fig. 6)

Un premier étage d'amplification, composé de U19 et U23, avec un gain sélectionnable de 1, 2, 5 ou 10, compense la réduction du signal utile lorsque l'amplitude du signal de test diminue. Comme son gain n'est pas modifié au cours d'un cycle de mesure, sa réponse en fréquence n'est pas très importante. L'étage suivant est soit U20 avec un gain de 1, soit U21 avec un gain de 3 (3,17 plus exactement), soit U22 avec un gain de 10 (10,1). U21 est compensé en fréquence par le réglage de C108, et U22 par celui de C51. La procédure d'étalonnage prendra en compte la réponse réelle de ce PGA, pour chacune des fréquences utilisées.

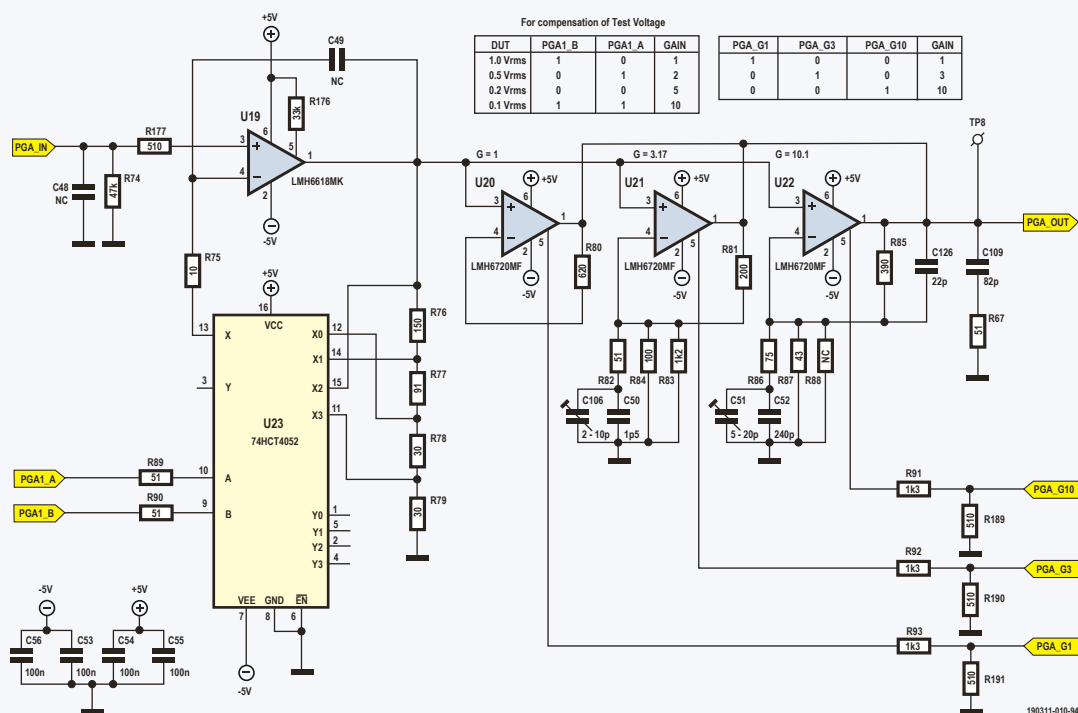
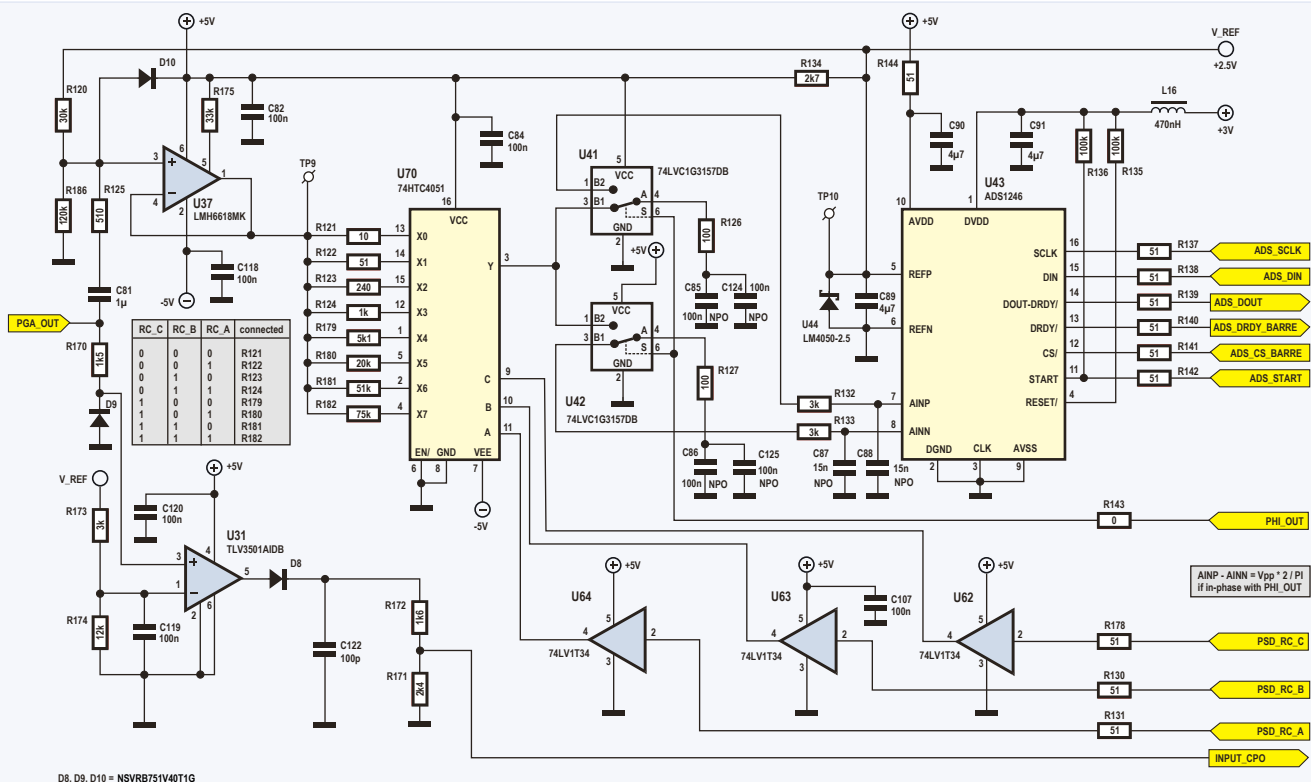


Figure 6. Amplificateur à gain programmable avec compensation de fréquence.



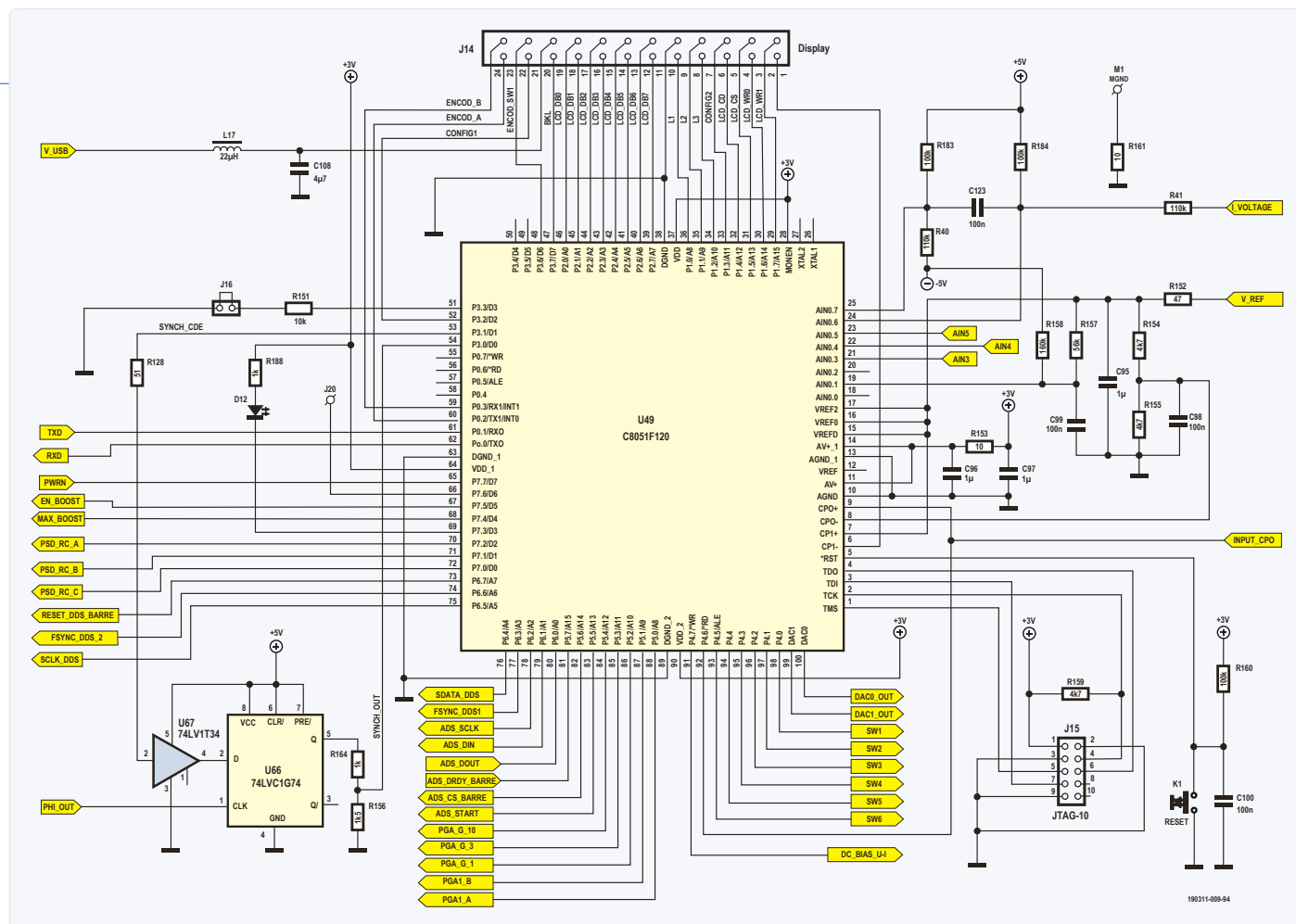


Figure 8. Un MCU de Silicon Labs est au cœur de ce LCR-mètre.

d'entrée de ce PSD.

La constante de temps pour le condensateur réservoir est ajustée en fonction de la fréquence de mesure : huit valeurs sont

possibles par la sélection, grâce à U70, d'une des résistances R121, R122, etc. Un décalage en continu de 2 V est appliqué sur l'entrée de U37 afin que le signal reste au maximum

dans la plage de résistance constante des interrupteurs analogiques (problème de linéarité de ce PSD).

Lors de la recherche des conditions de mesure optimales (choix du convertisseur I-V et du gain du PGA), le comparateur rapide U31 détecte tout dépassement de signal supérieur à 2 V crête, et envoie cette information à l'unité MCU (lancement d'une routine d'interruption).

MCU (microcontrôleur) (fig. 8)

Le cerveau de l'appareil est un MCU C8051F120 de Silicon Labs. Pourquoi utiliser ce type de circuit relativement ancien, alors qu'il existe des MCU plus modernes, par exemple avec une architecture ARM ?

- > il est suffisant quant aux périphériques intégrés et sa mémoire Flash programmable (64 K + 64 K).
- > le meilleur microcontrôleur est celui qu'on connaît bien...
- > je possède les outils de développement !

Son oscillateur interne (24,5 MHz) et une PLL (coefficient 3) donnent une fréquence d'horloge de 73,5 MHz.

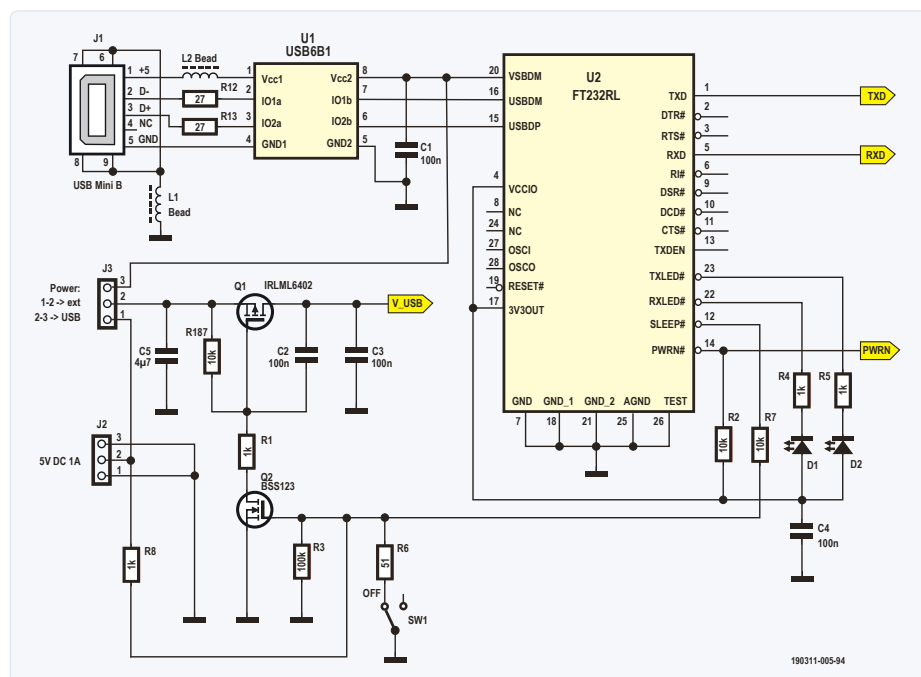


Figure 9. La passerelle USB/UART est utilisée pour la communication et les mises à jour de logiciels.

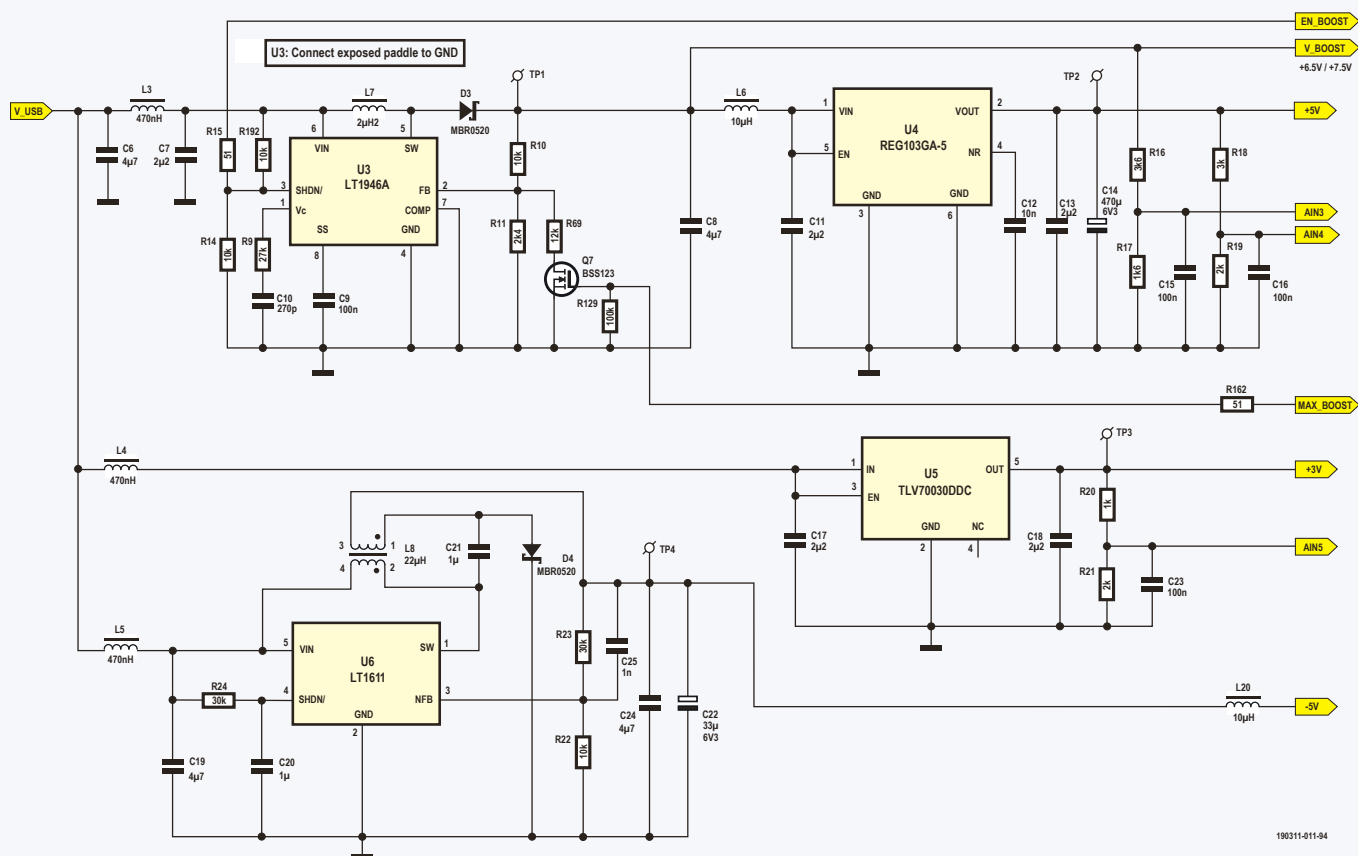


Figure 10. Quatre régulateurs de tension sont utilisés sur la carte principale.

L'embase J15 (JTAG) permet de raccorder un adaptateur de débogage USB de Silicon Labs (p. ex. RS réf. 757-0297) pour flasher le Bootloader dans l'EEPROM. Le connecteur J14 permet de connecter la carte d'extension de l'affichage.

La LED D12 est utilisée pour le débogage du microprogramme (par exemple, lorsque l'on appuie sur un bouton ou quand le codeur est actionné) et pendant le fonctionnement normal pour signaler certaines conditions d'erreur :

- Pas d'interface connectée sur J14 et pas de communication via USB : clignotement 0,5 s – 0,5 s.
- Erreur de test des alimentations : en plus du message sur le numéro d'erreur, clignotement permanent 0,5 s – 1 s.

Les circuits U66 et U67 permettent la synchronisation avec le signal sinusoïdal des mesures de tension et de courant du DUT.

Un cavalier sur le connecteur J16 informe le *Bootloader* d'une demande de mise à jour **inconditionnelle** du *firmware*.

Le poussoir K1 (RESET) est facultatif (il n'est utilisé que pendant la conception).

Contrôleur USB et tensions d'alimentation (fig. 9)

Un FT232RL (U2) de FTDI sert d'interface USB/UART avec le MCU. Normalement, l'alimentation se fait via le connecteur USB J1, mais il est également possible d'alimenter le circuit via le connecteur J2 et en plaçant un cavalier entre 1 et 2 de J3 au lieu de 2 et 3. Quatre tensions sont nécessaires pour alimenter tous les circuits (fig. 10) :

- tension V_BOOST de +6,5 V ou +7,5 V (selon le niveau de MAX_BOOST) fournie par le régulateur élévateur U3 avec L7 et D3.
- tension de +5 V fournie par le régulateur linéaire U4.
- tension de +3 V fournie par le régulateur linéaire U5.
- tension de -5 V fournie par le régulateur inverseur U6 combiné avec L8 et D4.

Toutes ces tensions sont vérifiées par le programme lors de la mise sous tension à l'aide de l'ADC à 12 bits et du multiplexeur intégré au MCU.

Afficheur (fig. 11)

Un câble plat en J1 relie cette carte à la carte mère.

Cinq poussoirs K1 à K5 et le commutateur rotatif SW1 sont disposés en matrice pour n'utiliser que 3 lignes de port du MCU. Le codeur rotatif utilise 2 lignes de port supplémentaires.

Les circuits RC opèrent un premier filtrage anti-rebond, simplifiant le filtrage par le logiciel.

Les diodes D12 et D13, en déclenchant une routine d'interruption du MCU, informent le programme interne qu'un bouton a été enfoncé. L'écran graphique U1 est alimenté en +3 V (régulateur linéaire U2) à partir de la tension d'alimentation USB filtrée. Le rétroéclairage est commandé par le transistor Q1, et le transistor Q2 permet, au moment d'éteindre l'appareil, une décharge rapide de la tension V_{LCD} interne, évitant ainsi des effets visuels indésirables.

L'afficheur est identifié, lors du démarrage, par la présence d'une résistance à la masse (R7 en série avec R9) sur la broche 19 (BKL) de J1.

Interface Bluetooth à faible consommation (fig. 12)

Une carte d'extension avec un module Bluetooth Low Energy (BLE) est en cours d'étude. Elle est identifiée par la broche 6 (CONFIG2) de J1 avec une résistance de 1 kΩ reliée à la masse.

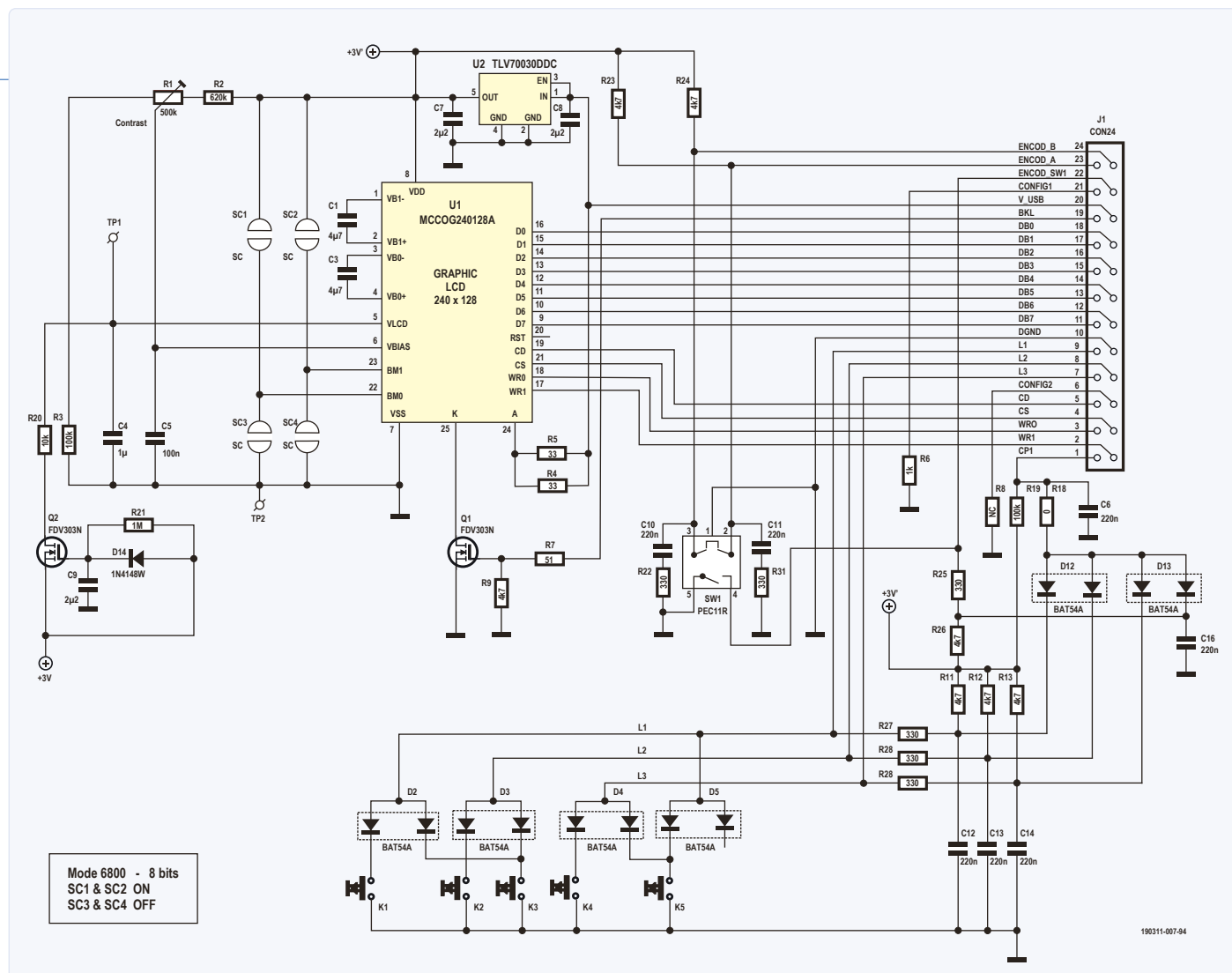


Figure 11. La carte d'extension derrière la façade, avec les boutons, l'écran LCD et le codeur rotatif.

Étalonnage dans le logiciel (fig. 13)

La technique du pont autoéquilibré utilisant un simple amplificateur opérationnel (A2) présente un gros inconvénient aux fréquences élevées : la masse virtuelle (fig. 13, tension V3) n'est pas maintenue à zéro en raison des limitations de cet amplificateur aux fréquences élevées.

La plupart des calculs simples de compensation des éléments parasites (OPEN/SHORT compensation) qui supposent que la tension V3 est une masse virtuelle ne sont donc plus applicables aux fréquences élevées. Il faut donc recourir à des méthodes plus sophistiquées de compensation, comme celle appelée *OPEN/SHORT/LOAD compensation*. Ceux qui s'intéressent à la théorie sur ce sujet liront le paragraphe 4.2.4 de l'article «Techniques de compensation des accessoires de test dans l'analyse d'impédance» [4]. L'étalonnage de l'appareil se fera en plusieurs étapes :

- Pour chaque fréquence prédéfinie et pour le PGA (A3), avec un gain de 3 puis 10.

- LOAD, pour chacune des 4 gammes avec $Z_x = R_{ref}$, pour chaque fréquence prédéfinie.
- OPEN TRIM (sans composant connecté), pour chaque fréquence prédéfinie.
- SHORT TRIM (entrée en court-circuit), pour chaque fréquence prédéfinie.

Les valeurs stockées seront utilisées pour calculer les corrections à apporter aux valeurs de mesure brutes.

Après ajustement des valeurs mesurées du DUT pour tenir compte du gain (valeur complexe) de A3, tant pour la mesure «COURANT» que pour la mesure «TENSION», la valeur finale sera calculée selon la formule :

$$Z_x = R_{ref} \frac{(Z_{xm} - Z_{short})(Z_{open} - Z_{stdm})}{(Z_{stdm} - Z_{short})(Z_{open} - Z_{xm})}$$

Toutes ces valeurs sont complexes :

Z_{xm} est la valeur mesurée

Z_{stdm} est la valeur mesurée de R_{ref} pendant

la phase de LOAD

Z_{short} est la valeur mesurée pendant la phase de SHORT TRIM

Z_{open} est la valeur mesurée pendant la phase OPEN TRIM

En cours d'utilisation, il est possible de répéter les deux opérations TRIM (bouton [TRIM]), à la fréquence actuelle, si par exemple la sonde de mesure a changé.

Lors de la sélection d'une fréquence utilisateur, les corrections seront effectuées par interpolation à partir des 2 fréquences prédéfinies proches.

Circuits imprimés (fig. 14a, b et c)

Les dessins des circuits imprimés peuvent être téléchargés sur le site web d'Elektor [3]. Étant donné la complexité de ces cartes - le PCB principal a quatre couches ! - il n'est pas question de les graver soi-même. Il est possible d'assembler soi-même à la main des cartes aussi complexes, mais cet exercice

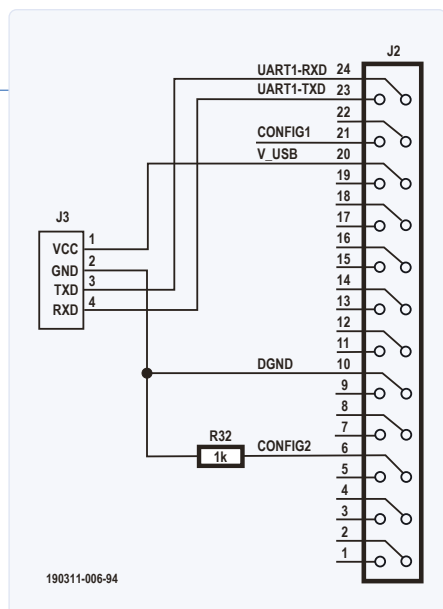


Figure 12. Une interface Bluetooth peut être utilisée pour commander le LCR-mètre.

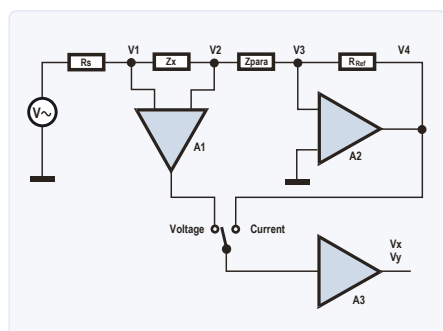


Figure 13. Schéma simplifié du circuit de mesure.

délicat est réservé aux professionnels de la soudure. Il faut de l'expérience beaucoup d'expérience, mais aussi les compétences et les outils adéquats. Avec de la patience et une main très sûre, une loupe et/ou un microscope (stéréo), le travail peut être effectué avec un fer à souder très fin, mais ce n'est pas idéal pour les composants avec *exposed pad*, dont la surface de soudure n'est pas directement accessible avec le fer à souder. Nous recommandons vivement d'utiliser de la pâte à souder avec un distributeur ou des pochoirs appropriés et un outil de soudure à air chaud ou un four de refusion. Si vous avez l'ombre d'un doute, **ne commencez même pas**.

Les fichiers Gerber pour les PCB (pour ceux qui veulent commander leurs PCB) sont disponibles en téléchargement, ainsi que les plans de perçage et de fraisage pour le boîtier Hammond que nous avons utilisé pour ce LCR-mètre [3]. En ce moment même, une campagne de soutien est en cours sur la page d'Elektor Labs <https://www.elektormagazine.com/labs/remake-lcr-meter>. Si nous atteignons les 150 promesses d'achat sur www.elektor.com/lcr

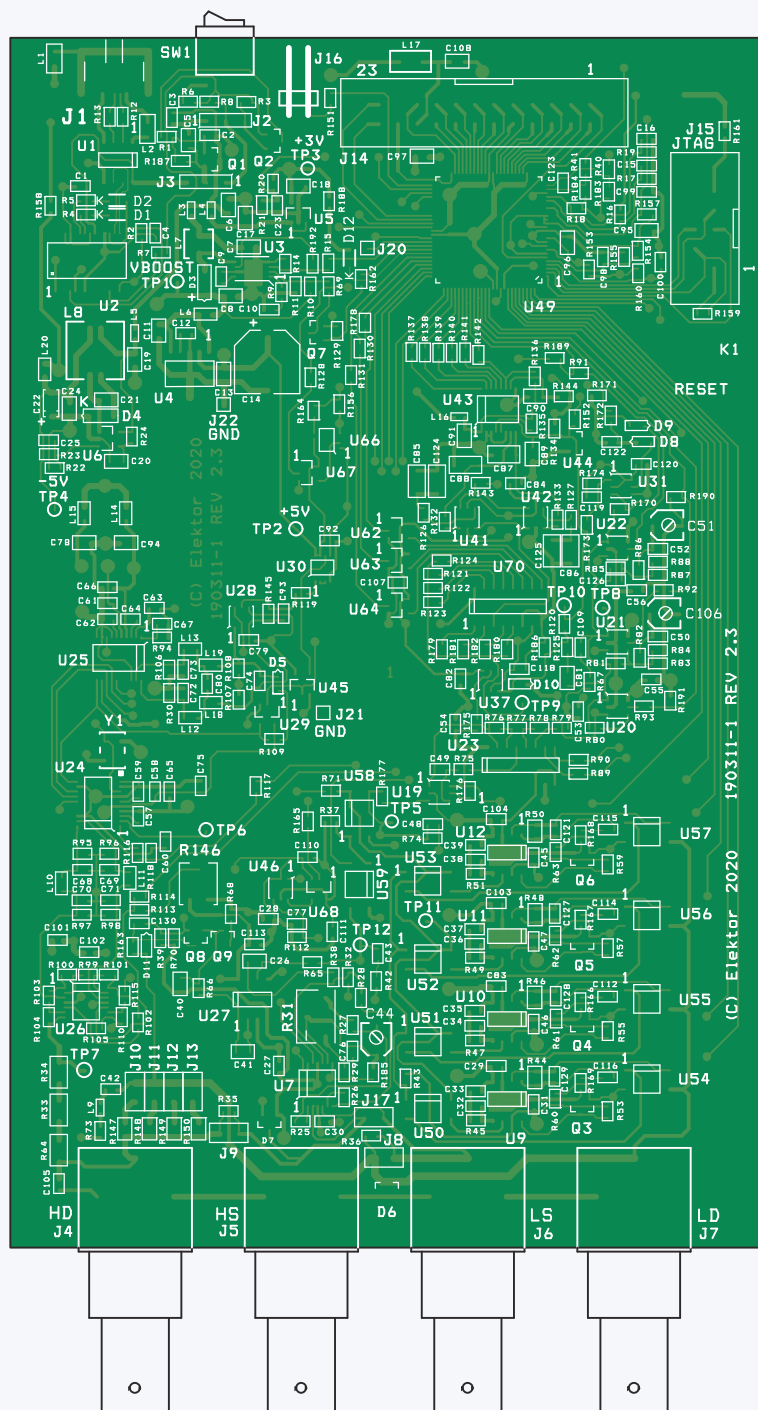


Figure 14a. Circuit imprimé de la carte principale, tous les composants sont montés sur la même face.

Elektor lancera la fabrication de **cartes** préassemblées (**fig. 14d**). Ne manquez pas votre chance de participer !

Accessoires

L'appareil est alimenté (5 V DC) par le connecteur Mini-USB de type B. Il vous faut donc un câble USB compatible avec le connecteur Mini-USB de type B (mâle) côté appareil et probablement avec le connecteur USB de type A (mâle) côté alimentation (celui d'un téléphone convient).

Comme la mesure est effectuée par une

configuration à 5 bornes [2] pour réduire au minimum l'influence des câbles de mesure, il y a 4 connecteurs BNC : de gauche à droite, devant : HD, HS, et LS, LD (H pour haut, L pour bas, D pour *drive* et S pour *sense*). Des sondes spéciales doivent être utilisées pour ce type de mesure.

Un clip Kelvin pour LCR-mètre avec câble à 4 fiches BNC (**fig. 15a**) est la solution la plus abordable (environ 10 € sur eBay). Il est recommandé de raccourcir les câbles autant que possible.

Le module TH26001A à 4 voies (**fig. 15b**)



LISTE DES COMPOSANTS DE LA CARTE PRINCIPALE - PCB 190311-1 V2.3

Résistances

(SMD 0603 1% 100 ppm sauf indication contraire)

R1,R4,R5,R8,R20,R25,R26,R37,R66,R109,R124,R164,R188 = 1 kΩ

R2,R7,R10,R14,R22,R60,R61,R62,R63,R151,R187,R192 = 10 kΩ

R3,R35,R36,R38,R65,R71,R112,R113,R129,R135,R136,R160,R183,R184,R185 = 100 kΩ

R6,R15,R45,R47,R49,R51,R67,R82,R89,R90,R119,R122,R128,R130,R131,R137,R138,R139,R140,R141,R142,R144,R162,R165,R178 = 51 Ω

R9 = 27 kΩ

R11,R171 = 2,4 kΩ

R12,R13 = 27 Ω

R16 = 3,6 kΩ

R17,R172 = 1,6 kΩ

R18,R132,R133,R173 = 3 kΩ

R19,R21,R110,R115 = 2 kΩ

R23,R24,R120 = 30 kΩ

R27,R28,R29,R32 = 2 kΩ 0,1% 25 ppm

R30,R80,R106,R107,R108 = 620 Ω

R31 = 50 Ω ajustable (Bourns 3266W)

R33,R34,R64 = 33Ω 100 ppm 1206

R39,R70,R84,R126,R127 = 100 Ω

R40,R41 = 110 kΩ

R42,R123 = 240 Ω

R43 = 560 Ω

R44,R147 = 100 Ω 0,01% 50 ppm 0805

R46,R148 = 1 kΩ 0,01% 10 ppm 0805

R48,R149 = 10 kΩ 0,01% 10 ppm 0805

R50,R150 = 100 kΩ 0,01% 10 ppm 0805

R53,R55,R57,R59,R180 = 20 kΩ

R68,R125,R177,R189,R190,R191 = 510 Ω

R69,R174 = 12 kΩ

R73,R88,R166,R167,R168,R169 = NC

R74 = 47 kΩ

R75,R121,R153,R161 = 10 Ω

R76 = 150 Ω

R77 = 91 Ω

R78,R79 = 30 Ω

R81 = 200 Ω

R143 = 0 Ω

R83 = 1,2 kΩ

R85,R95,R96,R97,R98 = 390 Ω

R86 = 75 Ω

R87 = 43 Ω

R91,R92,R93 = 1,3 kΩ

R94 = 6,8 kΩ

R99,R116,R117,R154,R155,R159 = 4,7 kΩ

R100,R101,R102,R103,R104,R105,R118 = 3,9 kΩ

R114 = 300 Ω

R134 = 2,7 kΩ

R145 = 7,5 kΩ

R146 = 10 kΩ ajustable (Vishay TS53YJ)

R152 = 47 Ω

R156,R170 = 1,5 kΩ

R157 = 56 kΩ

R158 = 160 kΩ

R163 = 200 kΩ

R175,R176 = 33 kΩ

R179 = 5,1 kΩ

R181 = 51 kΩ

R182 = 75 kΩ

R186 = 120 kΩ

Inductances

L1,L2 = perle Murata BLM21AG102SN1D

L3,L4,L5,L16 = 470 nH TDK MLZ1608DR47DT

L6,L14,L15,L20 = 10 µH TDK MLZ2012M100WT

L7 = 2,2 µH Coilcraft 1008PS-222KLB

L8 = 22 µH Cooper Bussmann DRQ73

L9 = NC

L10,L11,L12,L13,L18,L19 = 33 µH TDK MLZ2012M330WT

L17 = 22 µH Taiyo Yuden CBC3225T220MR

Condensateurs

(0603 sauf indication contraire)

C1,C2,C3,C4,C9,C15,C16,C23,C27,C28,C29,C30,C31,C45,C46,C47,C53,C54,C55,C56,C58,C59,C60,C61,C62,C63,C65,C66,C67,C75,C79,C82,C83,C84,C92,C98,C99,C100,C101,C102,C103,C104,C107,C110,C111,C113,C118,C119,C120,C123 = 100 nF 50 V 10% X7R

C5,C6,C8,C19,C24,C26,C40,C41,C78,C89,C90,C91,C94,C108 = 4,7 µF 10 V 10% X7R 0805

C7,C11,C13,C17,C18 = 2,2 µF 10 V 10% X7R 0805

C10 = 270 pF 50 V 5% NP0

C12,C57,C64,C77, C130 = 10 nF 50 V 10% X7R

C14 = 470 µF 6,3 V EEEFP0J471AP PANASONIC (aluminium électrolytique)

C20,C21,C81,C95,C96,C97 = 1 µF 25 V 10% X7R 0805

C22 = 33 µF 6,3 V CAS-A (tantale solide)

C25 = 1 nF 50 V 5% NP0 0603

C32,C34,C36,C38,C50,C93 = 1,5 pF 50 V 0,25 pF NP0

C33,C35,C37,C39 = 3,9 pF 50 V 0,25 pF NP0

C42,C48,C49,C105,C112,C114,C115,C116,C127,C128,C129 = NC

C43 = 180 pF 50 V 5% NP0

C44,C106 = 2-10 pF ajustable (Knowless Voltronics JZ100)

C51 = 4,5-20 pF ajustable (Knowless Voltronics JZ200)

C52 = 240 pF 50 V 5% NP0

C68,C69,C70,C71 = 220 pF 50 V 5% NP0

C72,C73,C109 = 82 pF 50 V 5% NP0

C74 = 47pF 50 V 5% NP0

C76 = 6,8 pF 50 V 0,25pF NP0

C80 = 68 pF 50V 5% NP0

C85,C86,C124,C125 = 100 nF 50 V 5% NP0 1206

C87,C88 = 15 nF 50 V 5% NP0 1206

C121 = 1 pF 50 V 0,25 pF NP0 0603

C122 = 100 pF 50 V 5% NP0 0603

C126 = 22 pF 50 V 5% NP0 0603

C112, C114, C115, C116, C127, C128, C19 = NC

Semi-conducteurs

D1,D2,D12 = LED rouge, faible puissance 0805

D3,D4 = MBR0520 diode Schottky

D5,D11 = 1N4148WS diode SOD323

D6,D7 = BAV199 LT1G double diode

D8,D9,D10 = NSVRB751V40T1G diode Schottky

Q1 = IRLML6402 MOSFET à canal P

Q2,Q3,Q4,Q5,Q6,Q7 = BSS123 MOSFET à canal N

Q8,Q9 = BFR106 RF NPN

U1 = USB6B1 protection ESD

U2 = FT232RL USB-UART

U3 = LT1946A régulateur à découpage DC/DC

U4 = REG103GA-5 régulateur linéaire 5 V 500 mA

U5 = TLV70030DDCR régulateur linéaire 3 V 200 mA

U6 = LM2611AMF convertisseur hacheur

U7,U26 = LMH6644MT quad OPAMP

U9,U10,U11,U12 = AD8099ARDZ OPAMP

U19,U37 = LMH6618MK OPAMP SOT23_6

U20,U21,U22 = LMH6720MF OPAMP SOT23_6

U23 = 74HCT4052D multiplexeur analogique

U24,U25 = AD9834BRUZ synthèse numérique

U27 = OPA2727AID amplificateur de précision

U28,U31 = TLV3501AIDB comparateur analogique

U29 = 74LVC1G04DBVT inverseur

U30,U66 = 74LVC1G74DCTR bascule

U41,U42,U46 = 74LVC1G3157DBVR commutateur analogique

U43 = ADS12461PW CAN à 24 bits

U44 = LM4050CIM3-2.5 référence de tension 2,5 V

U45,U62,U63,U64,U67,U68 = 74LV1T34DBVR tampon

U49 = C8051F120-GQ MCU à 8 bits

U50,U51,U52,U53,U58,U59 = AQY221N2S relais

U54,U55,U56,U57 = AQY221R2S relais

U70 = 74HCT4051D multiplexeur analogique

Divers

J1 = USB_B_Mini (TE Connectivity 1734035-2)

J2,J3 = SIL à 3 voies 2,54 mm

J4,J5,J6,J7 = BNC pour PCB (TE Connectivity 1-1337543-0)

J8,J9,J17,J10,J11,J12,J13 = SIL à 2 voies 2,54 mm

J16 = SIL 90° à 2 voies 2,54 mm

J14 = connecteur à 24 voies (Amphenol série T821)

J15 = connecteur à 10 voies (Amphenol série T821)

J20,J21,J22 = point de test

K1 = bouton-poussoir facultatif de 6 mm (Omron BF3-1020)

SW1 = interrupteur à bascule Nidec Copal AS1D-5M-10-Z

Y1 = oscillateur 27 MHz Epson X1G0044510005 SG5032CAN 27 MHZ TJGA

PCB 190311-1 VER 2.3

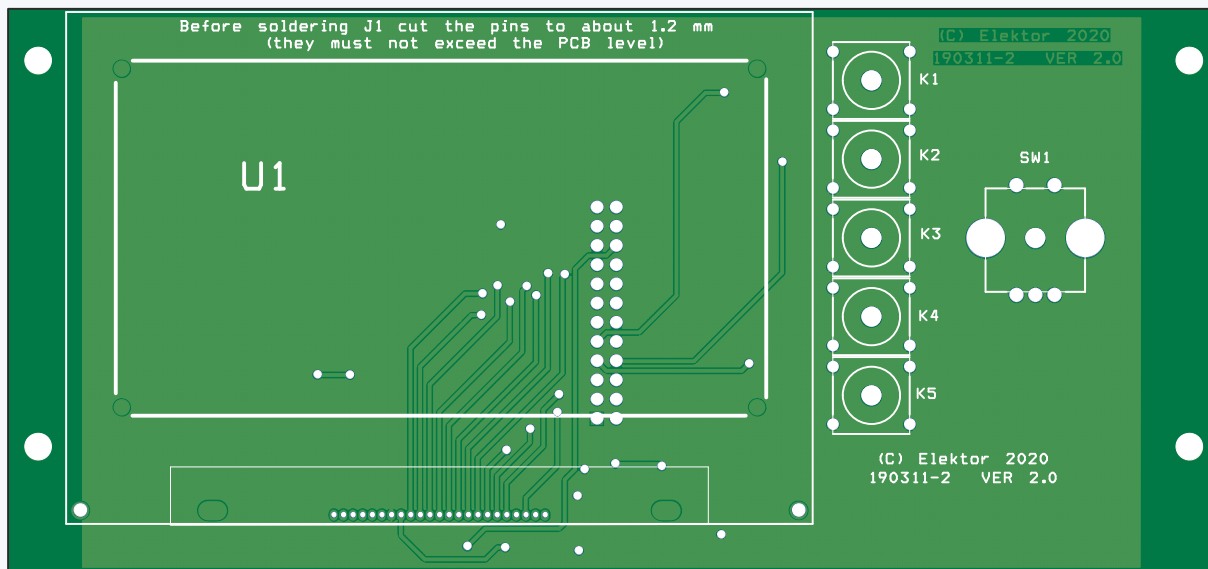
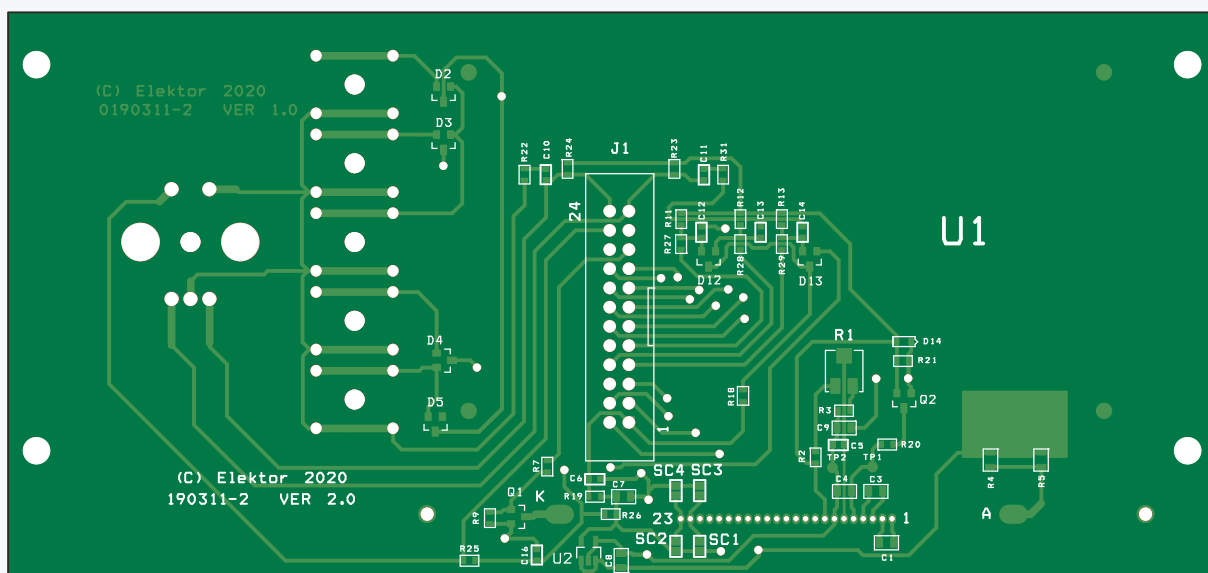


Figure 14b. Dessous du circuit imprimé de la carte d'extension. ▲

Figure 14c. Dessus du circuit imprimé de la carte d'extension. ▼



est plus cher (environ 55 € sur eBay), mais présente l'avantage certain d'une connexion courte et robuste entre le DUT et le LCR-mètre.

Remarquez l'inversion du marquage HAUT et BAS de ce module par rapport à celui du LCR-mètre !

Précision

La précision obtenue avec cet appareil dépend principalement

- du type de composant testé (L, C ou R) et sa valeur,
- de la fréquence de test qui doit être appropriée pour ce composant.
- du dispositif de test (pinces Kelvin, module compact...), là encore en fonction de la fréquence de test.
- du soin apporté à l'étalonnage (à refaire dès que la température a sensiblement

changé).

C'est l'impédance Z du composant qui conditionne la plage de mesure (choix du convertisseur I/U) et l'activation (ou non) du PGA.

La précision sera maximale (0,1%) pour $50 \Omega < Z < 330 \text{ k}\Omega$ et en fonction de la gamme pour les fréquences $< 100 \text{ kHz}$.

Pour les valeurs extrêmes, elle devrait être

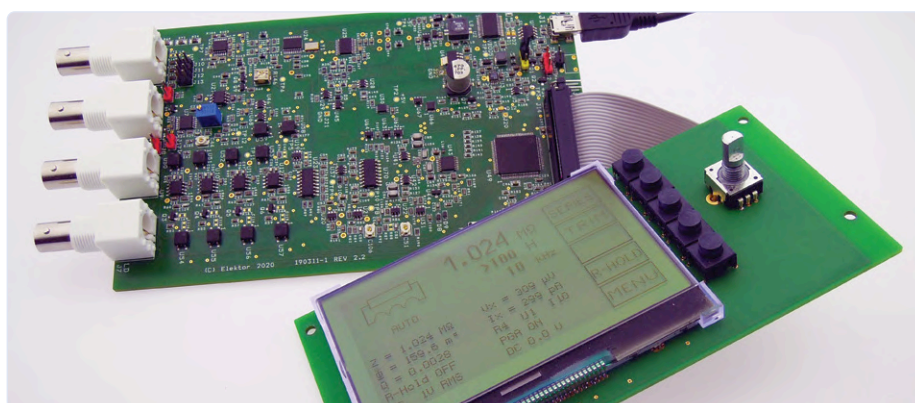


Figure 14d. Les deux cartes interconnectées pour les essais



LISTE DES COMPOSANTS DE LA CARTE D'EXTENSION - PCB 190311-2 V2.0 ET BOÎTIER

Résistances

(SMD 0603 sauf mention contraire)

R1 = 500 k Ω

R2 = 620 k Ω

R3, R19 = 100 k Ω

R4, R5 = 33 Ω 0805

R7 = 51 Ω

R6, R8 = NC

R9, R11, R12, R13, R23, R24, R26 = 4,7 k Ω

R18 = 0 Ω

R20 = 10 k Ω

R21 = 1 M Ω

R22, R25, R27, R28, R29, R31 = 330 Ω

Condensateurs

C1, C3 = 4,7 μ F 10 V 10% X7R 0805

C4 = 1 μ F 25 V 10% X7R 0805

C5 = 100 nF 50 V 10% X7R 0603

C6, C10, C11, C12, C13, C14, C16 = 220 nF 50 V 10% X7R 0603

C7, C8, C9 = 2,2 μ F 10 V 10% X7R 0805

Semi-conducteurs

D2, D3, D4, D5, D12, D13 = double diode BAT54A SOT23

D14 = diode 1N4148WS SOD323

Q1, Q2 = MOSFET numérique à canal N

FDV303N SOT23

U1 = afficheur MCCOG240128A6W-FPTLW (MIDAS)

U2 = TLV70030DDCR régulateur linéaire 3 V 200 mA SOT23-5

Divers

J1 = connecteur à 24 voies (Amphenol série T821)

SW1 = codeur rotatif PEC11R-4115F-S0018 (BOURNS)

PCB 190311-2 VER 2.0

coffret HAMMOND 1455N1601

4 x entretoise hexagonale femelle/femelle M3 7 mm

1 x entretoise hexagonale femelle/femelle M3 6 mm

1 x rondelle plate en acier 3 mm

5 x vis à tête fraisée M3 16 mm

5 x rondelle M3

5 x boulon M3

2 x prise IDC 24 contacts (AMPHENOL T81214A101CEU)

15 cm câble plat 24 voies

bouton (pour codeur rotatif) en aluminium (p. ex. MENTOR 507.61)

4 x cavalier 2,54 mm plaqué or



Figure 15a. Câbles Kelvin, disponibles à un prix raisonnable.



Figure 15b. TH26001A, module de test compact, plus coûteux.

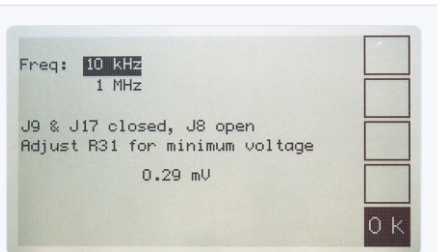
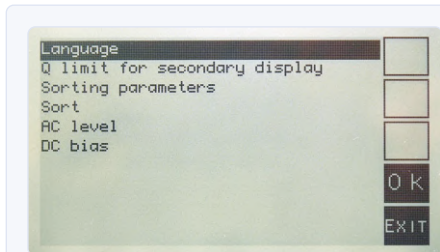


Figure 16. Deux captures d'écran du menu d'étalonnage, dont nous découvrirons les subtiles options dans le 2e volet de l'article. La simplicité de son étalonnage est l'un des atouts majeurs de cet appareil.

PROJET « KICKSTARTER » ELEKTOR

> Soutenez ce projet et bénéficiez d'une remise de 100 €

Kit comprenant une carte mère + une carte d'extension (affichage) + tous les composants

www.elektor.com/lcr



Test comparatif

Au sommaire du prochain numéro, on trouvera un article présentant les mesures comparatives effectuées avec le nouveau LCR-mètre AU2019 et des instruments de référence tels que HP 4263B-1 4263B-2, Hameg HM8118, Keysight (Agilent) U1732A). ◀

190311-02

Ont contribué à cet article :

Idée, conception, texte :

Jean-Jacques Aubry

Illustrations : Kurt Diedrich

Rédaction : Luc Lemmens, Denis Meyer

Maquette : Giel Dols

Mesures : Alfred Rosenkränzer

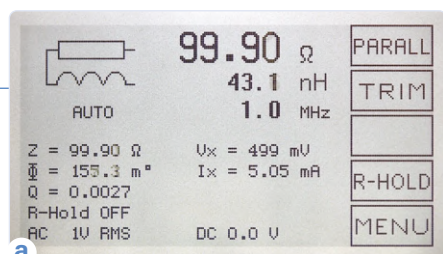


Figure 17a. Mesure d'une résistance de faible valeur (100 Ω), dont apparaît l'inductance parasite (43,1 nH) à fréquence élevée !

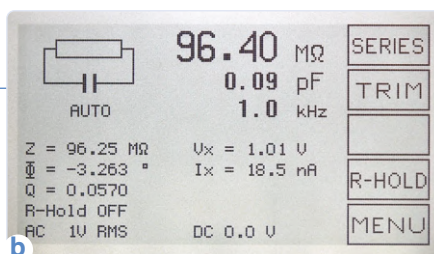


Figure 17b. Mesure d'une résistance de forte valeur (100 M Ω) dont apparaît la capacité parasite (0,09 pF) aux fréquences basses !

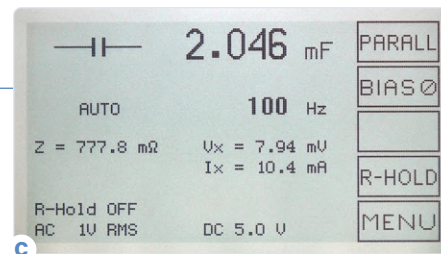


Figure 17c. Mesure d'un gros condensateur à basse fréquence avec polarisation de 5 V CC.

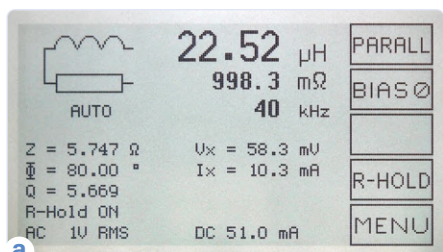


Figure 18a. Mesure d'une inductance avec courant de polarisation

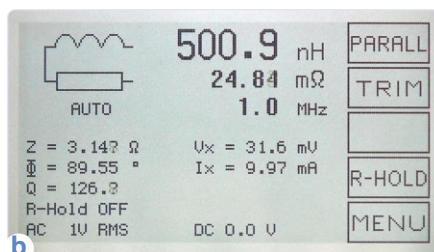


Figure 18b. Mesure d'une inductance à une fréquence plus élevée.

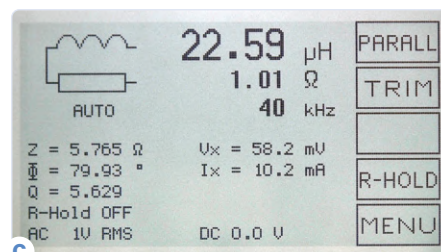


Figure 18c. Mesure d'une inductance sans courant de polarisation.

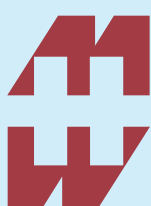
Votre avis, s'il vous plaît...

Adressez vos questions et vos commentaires à l'auteur jjacques.aubry@free.fr ou à la rédaction redaction@elektor.fr

LIENS

- [1] Qt : www.qt.io/
- [2] The Impedance Measurement Handbook (2003 – Agilent Technologies Co. Ltd) Fig. 3.4 : https://wiki.epfl.ch/carplat/documents/rcl_agilent.pdf
- [3] Téléchargement complet (Gerber, perçage, BOM, ...) : www.elektormagazine.fr/190311-02
- [4] Test Fixture Compensation Techniques in Impedance Analysis : www.eejournal.ktu.lt/index.php/elt/article/view/11206/5939
- [5] Manuel d'exploitation (version à jour) : www.elektormagazine.fr/190311-02
- [6] Manuel de l'utilisateur (version à jour) : www.elektormagazine.com/190311-02
- [7] Page du projet de LCR-mètre sur Elektor Labs : www.elektormagazine.fr/labs/remake-lcr-meter
- [8] LCR-mètre_2013-a-FR2013030441.pdf – mars 2013 p. 44 : www.elektormagazine.fr/magazine/elektor-201303/20354
- [9] LCR-mètre_2013-b-FR2013040561.pdf – avril 2013 p. 56 : www.elektormagazine.fr/magazine/elektor-201304/20056
- [10] LCR-mètre_2013-c-FR2013050461.pdf – mai 2013 p. 46 : www.elektormagazine.fr/magazine/elektor-201305/20563
- [11] LCR-mètre_2013-d-FR2013060121.pdf – juin 2013 p. 12 : www.elektormagazine.fr/magazine/elektor-201306/20705
- [12] LCR-mètre_2013-e-FR2013071141.pdf – juillet 2013 p. 114 : www.elektormagazine.fr/magazine/elektor-201307/20863
- [13] LCR-mètre_2013-f-FR2013100621.pdf – octobre 2013 p. 62 : www.elektormagazine.fr/magazine/elektor-201310/23467

Advertisement



**HAMMOND
MANUFACTURING®**

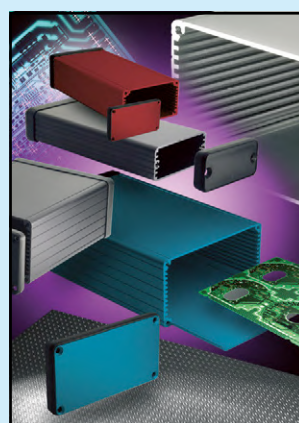
1455 boîtiers en aluminium extrudés

En savoir plus: hammfg.com/1455

Plus de 5000 modèles de boîtiers.

hammfg.com/small-case

eusales@hammfg.com • + 44 1256 812812



connectez votre sonnette aux objets sur l'internet

avec Home Assistant et ESPHome

Ont contribué à cet article :

Idée, circuit, texte et illustrations : **Clemens Valens**

Schéma : **Kurt Diedrich**

Rédaction : **CJ Abate**

Traduction : **Hervé Moreau**

Maquette : **Giel Dols**

ESPHome et Home Assistant sont deux projets domotiques open source facilitant grandement la programmation des appareils à puces ESP8266 et ESP32. Après avoir partagé avec vous ma découverte de ces deux outils dans le numéro précédent [1], je vous invite ici à découvrir comment je les ai mis à profit pour ajouter la sonnette de notre maison à son système domotique.



Figure 1. Comment transformer une sonnette rudimentaire en système connecté complexe.

Il y a quelques années, j'ai remplacé le carillon électromécanique de notre maison par un modèle sans fil offrant différentes mélodies. Deux récepteurs en allongeaient considérablement la portée, mais l'immanquable usure de leurs piles finissait par rendre la sonnette capricieuse. En outre les visiteurs n'entendaient pas la mélodie lorsqu'ils appuyaient sur le bouton. Comme le fameux facteur, ils sonnaient toujours deux fois. Je me suis dit que l'heure de la modernité avait sonné pour cette sonnette. Le résultat final est une sonnette connectée capable de produire un « ding-dong »

mécanique, jouer différentes mélodies numériques et envoyer des notifications dans le nuage (**fig. 1**). Tout ça en même temps. C'est en quelque sorte le croisement de trois générations de sonnettes. Et cette fois-ci les visiteurs l'entendent de l'extérieur.

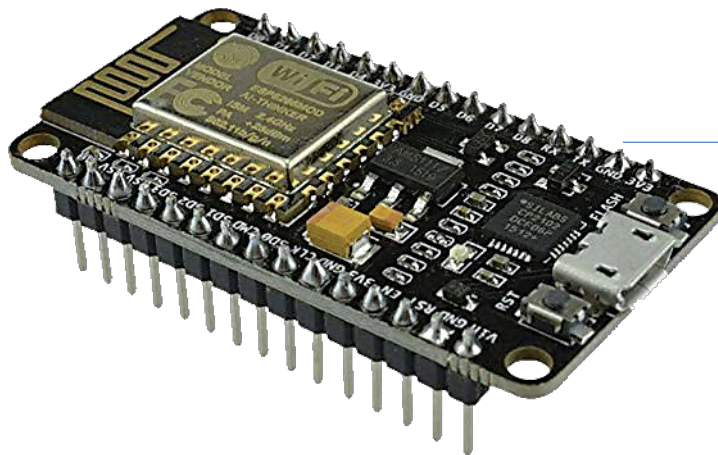
Circuit de l'interface de sonnette

J'ai choisi une carte NodeMCU à ESP8266 (**fig. 2**) pour construire l'interface Wi-Fi car elle offrait une entrée analogique dont j'allais avoir besoin. Comme elle se programme sans

câble USB-série, ce projet ne nécessite aucun outil spécial, fer à souder excepté.

Toujours en place avec ses fils et son alimentation, le boîtier du vieux carillon électromécanique offrait assez d'espace pour y loger l'interface Wi-Fi et l'alimenter. J'ai modifié le bouton de la sonnette sans fil et l'ai relié au carillon en profitant du câblage resté en place. Comme ce bouton est à l'extérieur, j'ai protégé la connexion avec des résistances en série, un condensateur de filtrage et des diodes de niveau. Avec ses contacts ne pouvant absorber qu'une cinquantaine de milliampères, ce

Figure 2. L'entrée analogique du module NodeMCU est flanquée d'un diviseur de tension qui divise par 3,2 la tension d'entrée.



petit bouton tactile aurait été bien incapable d'activer la bobine du carillon électromécanique puisqu'il lui aurait fallu pour cela délivrer 1 A. J'ai réglé le problème avec un petit relais de 5 V commandé par le μC via un transistor. La **figure 3** montre le schéma complet.

Pile sous surveillance

Le fil de commande du bouton de la sonnette est relié à V_{BAT} par sa résistance de rappel. Lorsque le bouton n'est pas pressé, il règne donc entre ses bornes la tension de la pile.

C'est pourquoi je l'ai relié à une broche GPIO du μC , mais aussi à son entrée analogique : cela permet à *Home Assistant* de surveiller le niveau de la pile et de relayer les messages de la sonnette. C'est pratique, car la portée de la sonnette sans fil est proportionnelle à la tension de la pile, et le ou les récepteurs ne peuvent plus être atteints lorsque cette tension est trop basse.

Notez qu'il serait bien sûr possible d'utiliser l'entrée analogique pour détecter les

pressions sur le bouton et surveiller le niveau de la pile, mais cela compliquerait l'écriture du micrologiciel. L'exploitation d'une broche GPIO libre évite cette complication.

Alimentation

Le carillon électromagnétique n'offrait qu'un transformateur de 12 V alors que le relais et la carte NodeMCU attendaient 5 V CC. Un petit aménagement s'imposait. Comme la bobine du carillon charge lourdement le transformateur, celui-ci voit sa tension de sortie

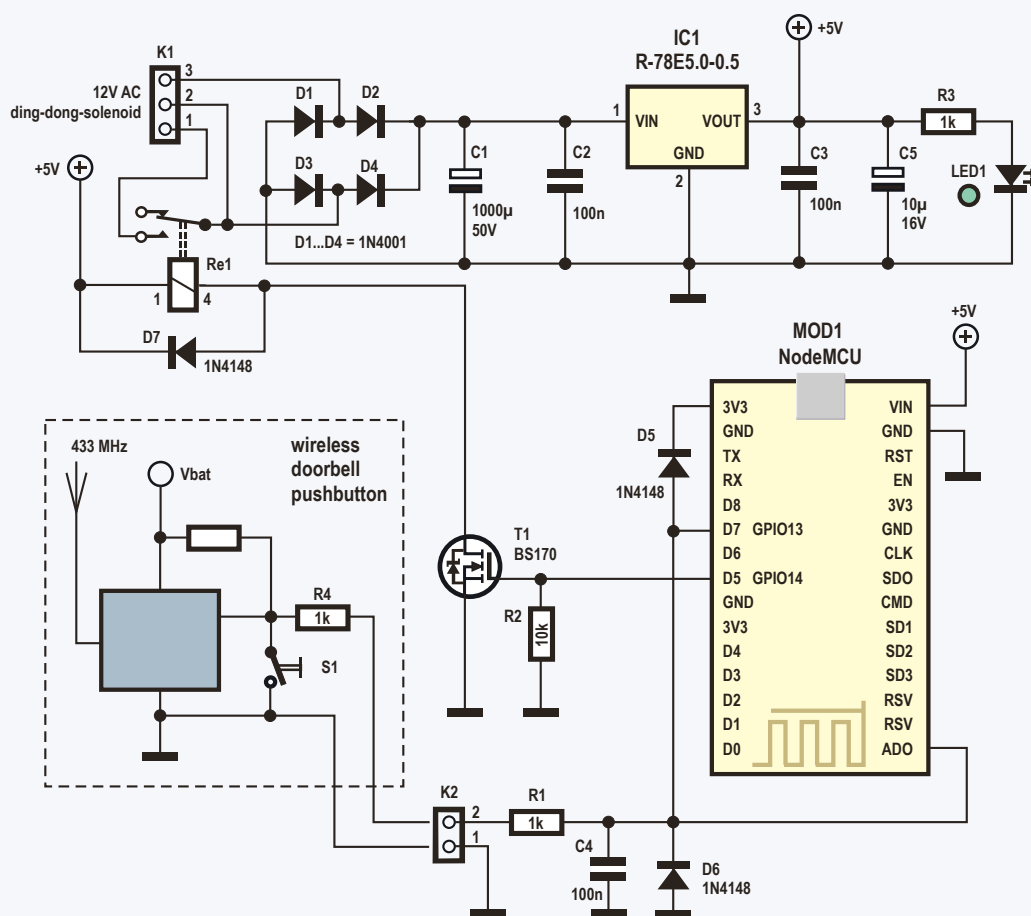


Figure 3. L'interface a besoin d'une alimentation régulée. Les broches reliées au bouton de la sonnette sont protégées des aléas naturels par une circuiterie particulière.

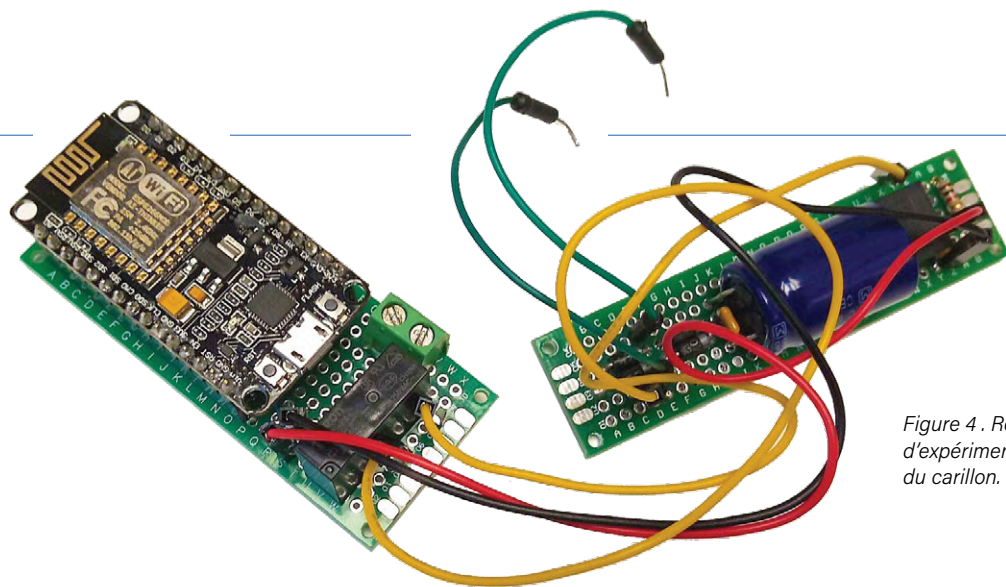


Figure 4 . Répartir l'interface sur deux platines d'expérimentation a permis de la faire tenir dans le boîtier du carillon.

BREF RAPPEL

ESPHome est un micrologiciel open source pour dispositifs à ESP8266 ou ESP32 : prises intelligentes, LED, capteurs, etc.

<https://esphome.io/>

Home Assistant est un contrôleur domotique open source qui relie des dispositifs connectés (capteurs, prises intelligentes, etc.) à des actionneurs ou des services, et permet de commander l'ensemble au moyen de règles complexes.

<https://www.home-assistant.io/>

Pour plus de détails, voir [1].

baisser chaque fois que la bobine est activée. Un condensateur de grande capacité filtre ces baisses indésirables. Sans lui, le module NodeMCU redémarrerait à chaque coup de sonnette.

Pour éviter les problèmes de dissipation de chaleur, j'ai choisi comme régulateur de 5 V un petit régulateur à découpage de type 7805. J'ai par ailleurs ajouté un témoin d'alimentation car la carte NodeMCU en est dépourvue et qu'une telle LED est toujours pratique.

J'ai réparti le circuit de l'interface sur deux plaques d'essai – une pour l'alimentation, l'autre pour le reste – et les ai nichées dans le boîtier de l'ancien carillon (fig. 4).

Configuration du micrologiciel ESPHome

Je me suis servi de l'outil **ESPHome** pour créer l'interface de ma sonnette. Avec

ESPHome, une demi-heure de programmation suffit pour mettre en service un périphérique intelligent doté de sa propre interface web et d'une fonction de programmation à distance (*Over-The-Air*, OTA). Et quand je dis « programmation », il s'agit en fait d'écrire un fichier de configuration avec le langage de balisage YAML. Le périphérique sera en outre pris en charge par l'extension **Home Assistant** et pourra être modifié à tout instant depuis un smartphone, une tablette ou un ordinateur.

ESPHome prend en entrée un fichier de configuration (ou script) YAML, le traduit en code C/C++ et le compile avec le reste de la bibliothèque. Le script spécifie les éléments d'**ESPHome** à inclure (paramètres Wi-Fi, interface web ou non, ce genre de choses), quels types de capteurs et d'actionneurs sont reliés aux broches du µC, et que faire avec. La plupart des choses communes sont prédéfinies, d'où

la facilité et la rapidité de la mise en route. YAML est un langage simple. Ses règles d'indentation sont strictes, mais on finit par s'y faire.

Association du relais au bouton

Le relais qui commande la bobine du carillon est défini en tant que **switch** interne relié à GPIO14. Mettre sa clé **internal** à **true** le rend invisible à **Home Assistant**, ce qui évite d'encombrer le système.

switch:

```
- platform: gpio
  pin: GPIO14
  id: ding_dong_relay
  internal: true
```

Le bouton est un **binary_sensor** sur la broche GPIO13. La valeur **true** de sa clé **inverted** indique à **ESPHome** qu'il est actif au niveau bas.

binary_sensor:

```
- platform: gpio
  name: "my doorbell pushbutton"
  pin:
    number: GPIO13
    inverted: true
  on_press:
    then:
```

```
- switch.turn_on:
  ding_dong_relay
- delay: 200ms
- switch.turn_off:
  ding_dong_relay
```

ESPHome crée un événement **on_press** lorsque le bouton est pressé, puis l'envoie à **Home Assistant**. Un tel événement peut servir de signal dans le fichier YAML pour déclencher une ou plusieurs actions. Appelé **automation** dans la documentation, ce mécanisme est l'une des forces d'**ESPHome**.

Votre avis, s'il vous plaît...

Pour toute question ou commentaire relatifs à cet article, contactez l'auteur (clemens.valens@elektor.com) ou la rédaction (redaction@elektor.fr).

Ci-dessus j'ai utilisé l'évènement `on_press` pour produire une impulsion de 200 ms sur la sortie du relais `ding_dong_relay`. Ainsi ma sonnette actionnera-t-elle systématiquement le carillon électromécanique, même en cas de coupure de courant. C'est ce qu'elle faisait il y a 40 ans, mais à l'ancienne, par électricité pour ainsi dire. Aujourd'hui elle est entourée d'une mégapole de transistors.

Surveillance de la pile

La pile est définie en tant que `sensor` sur A0. Son type (`platform`) est `adc` puisque sa tension doit être convertie en valeur numérique. Comme le module NodeMCU divise par 3,2 la valeur lue sur A0, il nous faut la multiplier par 3,2 pour retrouver la valeur originale. C'est ce que fait le filtre `multiply` ci-dessous. Un filtre est un opérateur modifiant la valeur de sortie d'un capteur.

`sensor:`

```
- platform: adc
  name: "doorbell battery level"
  pin: A0
  filters:
    - multiply: 3.2
```

PROGRAMMER ESPHOME SUR UN DISPOSITIF VIERGE

Tout nouveau dispositif à ESP8266 ou ESP32 doit d'abord être programmé par le port série avant de devenir compatible avec *ESPHome*. Pour cela il vous faudra peut-être installer un pilote USB-série sur l'ordinateur exécutant *Home Assistant*. La nécessité ou non de ce pilote dépend du dispositif. Les cartes NodeMCU à puce CP2102 USB de Silabs (*Silicon Laboratories*) que j'ai programmées depuis un Raspberry Pi n'en ont pas eu besoin.

Reliez votre dispositif à l'ordinateur exécutant *Home Assistant*, puis ouvrez le tableau de bord d'*ESPHome* depuis la barre latérale (si vous avez activé cette option). Vous pouvez aussi l'ouvrir depuis le panneau *Supervisor* en sélectionnant la carte de l'add-on *ESPHome* et en cliquant sur *Open Web UI*. Ouvrez la liste déroulante indiquant *OTA (Over-The-Air)* par défaut. Vérifiez que le port série est disponible. S'il ne l'est pas, redémarrez l'add-on *ESPHome* (en revenant au panneau *Supervisor*).

Cliquez ensuite sur le bouton rosâtre + du tableau de bord d'*ESPHome* et laissez-vous guider par l'assistant. C'est probablement stupide de ma part, mais je n'ai utilisé aucun mot de passe pour mes dispositifs...

Pour le chargement du micrologiciel, sélectionnez le port auquel le dispositif est connecté.

Exploitation de la LED embarquée

La carte NodeMCU est équipée d'un bouton *Flash* relié à GPIO0 et d'une LED reliée à GPIO16. Le premier peut p. ex. servir de bouton de test ; pour cela j'ai défini GPIO0 dans la section `binary_sensor` comme composant de type `binary`. Je me suis servi de la LED pour indiquer qu'un visiteur vient d'appuyer sur la sonnette et que *Home*

Assistant a détecté l'évènement.

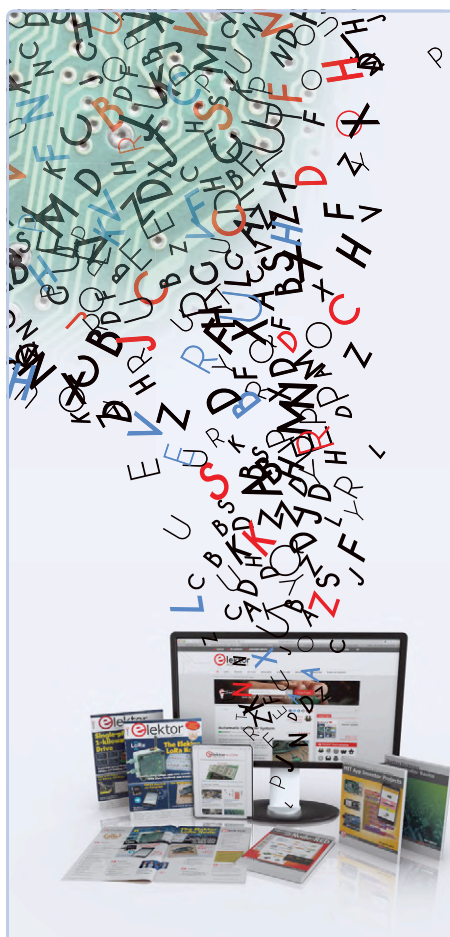
`light:`

```
- platform: binary
  name: "LED1"
  output: led1
```

`output:`

```
- platform: gpio
  id: led1
```

Publicité



Elektor cherche des auteurs

Le coronavirus bouleverse nos vies, avec parfois des conséquences positives.

Le temps libéré, vous pouvez l'utiliser pour **partager** vos connaissances en **électronique** avec d'autres. Selon vos talents, le plus simple consiste à donner des cours **vidéo** ou à écrire un **article** ou un **livre**. Vous avez une bonne idée ? Action !

Elektor vous assistera. Outre la satisfaction de cette expérience, il y aura des recettes pécuniaires.

Faites-nous part de votre idée, nous vous répondrons.

elektor.fr/cherche-des-auteurs

elektor
design > share > sell

**> Carte NodeMCU à microcontrôleur ESP8266**www.elektor.fr/nodemcu-microcontroller-board-with-esp8266-and-lua**> Livre IoT Home Hacks with ESP8266**www.elektor.fr/iot-home-hacks-with-esp8266

pin:

number: GPIO16

inverted: true

J'ai défini un composant `light` relié à une sortie appelée `led1`. Cette définition crée une entité « light » dans *Home Assistant* qui offre d'autres possibilités que celles associées à un composant « switch » (et qui a une icône différente). La LED reliée à GPIO16 est définie en tant que `output` avec une clé `id` appelée `led1`. Sa clé `inverted` vaut `true` car elle est active au niveau bas.

Avec ces définitions, tout message envoyé par *Home Assistant* au composant `LED1` pour l'allumer ou l'éteindre sera transféré à la sortie correspondante, soit `led1`.

Le fichier de configuration YAML complet est à télécharger depuis [2]. N'oubliez pas de compléter les clés `ssid` et `password` de votre réseau Wi-Fi, et éventuellement de changer le nom de la sonnette.

Extension ESPHome pour Home Assistant

L'extension *ESPHome* pour *Home Assistant* permet d'écrire le script YAML d'un dispositif *ESPHome* directement depuis le logiciel *Home Assistant*, de le compiler, de le charger dans le µC, et aussi de le mettre à jour à distance. Un système ESP8266 vierge doit toutefois être d'abord programmé par le port série car la programmation OTA n'a pas encore été activée (cf. encadré).

Automatisation depuis Home Assistant

Voyons maintenant comment associer une règle d'automatisation à la LED reliée à GPIO16 de façon à ce qu'elle s'allume lorsqu'un visiteur appuie sur la sonnette.

La règle pour cet automate est fort simple : déclencher l'action « mettre la LED sous tension » lorsque survient l'évènement « appui sur un bouton ».

L'éditeur d'automates de *Home Assistant* peut être ouvert de différentes façons. L'une d'elles est de cliquer sur *Configuration* dans le menu de gauche, puis de cliquer sur *Automations*. On crée un nouvel automate en cliquant sur le bouton rond orange « + » situé dans le coin inférieur droit.

Il est possible d'entrer une instruction écrite et de laisser *Home Assistant* tenter de la convertir en code, mais trouver la bonne phrase n'est pas facile. La règle créée risque d'être boiteuse, et la corriger prend du temps. Bref, mieux vaut ici cliquer sur *Skip*.

Entrez un nom pour l'automate et, en option, une description. Laissez le mode sur *Single (default)*.

Déclencheurs et actions

Là aussi il existe plusieurs façons de spécifier un évènement déclenchant l'automate. Selon moi la plus intuitive est de choisir un *Device* comme type de déclencheur (*Trigger type*). Dans la liste *Device*, cherchez le nom que vous avez attribué à votre dispositif dans le fichier YAML. Cherchez de même le nom de votre bouton dans la liste *Trigger*.

Il n'y a pas de conditions ici, donc passez cette partie.

Les actions sont semblables aux déclencheurs, et là encore il y a plus d'une façon d'obtenir le résultat souhaité. L'une d'elles est de choisir *Device* comme type d'action (*Action type*) et de sélectionner ce dispositif dans la liste *Device*. Dans la liste *Action*, sélectionnez *Turn on LED1* (si vous aviez choisi un autre nom pour LED1, utilisez-le).

Enregistrez votre automate en cliquant sur

l'icône orange située en bas à droite.

Vérifiez que l'automate fonctionne en appuyant sur la sonnette : la LED rouge s'allume (ou en tout cas elle le devrait) mais, petit problème, elle reste allumée...

Ajout d'une carte Light

Corrigeons cela en ajoutant une « carte » de type *Light* au tableau de bord de *Home Assistant*. Cette carte représentera la LED et permettra de l'éteindre (en cliquant dessus) et d'en connaître l'état à distance.

Dans le menu de gauche, cliquez sur *Overview*. Cliquez ensuite sur les trois points du coin supérieur droit (*Open Lovelace UI menu*) et sélectionnez *Configure UI*. Cliquez sur le bouton orange « + » du coin inférieur droit. Cherchez la carte *Light* et cliquez dessus. Cherchez maintenant la liste *Entity* pour la LED et sélectionnez-la (oui, je sais, on se balade beaucoup à dos de souris dans *Home Assistant*). Affectez-lui un nom, cela facilitera son identification dans l'interface. Car même si cette LED est pour l'instant la seule de son espèce, d'autres viendront s'ajouter à votre système domotique à mesure que celui-ci évoluera. Un nom bien choisi permet de repérer rapidement qui est qui et qui fait quoi.

Enregistrez votre carte, et quittez le mode de configuration en cliquant sur la croix du coin supérieur gauche (*Close*).

Cliquez sur la carte pour changer l'état de la LED.

Sonnez et entrez dans le monde de l'IdO

ESPHome et *Home Assistant* évoluant continuellement, il est possible qu'une nouvelle version parue entre-temps vous oblige à adapter l'une ou l'autre des étapes décrites ici. Quoi qu'il en soit vous devriez être en mesure d'assembler votre propre sonnette Wi-Fi connectée et d'automatiser ses réponses depuis *Home Assistant*. Vous connaissez les principes de base, à vous de franchir cette première marche pour créer une sonnette adaptée à vos besoins. ◀

200089-02

LIENS

[1] **La domotique, c'est facile avec...** : www.elektormagazine.fr/magazine/elektor-156/58991

[2] **How-To: Integrate Your Doorbell in Home Assistant Using ESPHome** : www.elektormagazine.com/labs/esphome-doorbell

comment bien gérer mes batteries ?

Qui veut aller loin ménage son lithium

Thomas Scherer

Avec les piles rechargeables modernes à base de lithium, un métal alcalin, le stockage de l'énergie électrique est devenu à la fois plus simple et... plus compliqué ! À volume égal, vous pouvez stocker plus d'énergie et n'avez pratiquement pas d'autodécharge, mais la gestion de la charge des batteries à plusieurs cellules nécessite un certain doigté.



Qu'y a-t-il de si particulier dans les piles au lithium ? Pourquoi leur faut-il, contrairement aux autres techniques de stockage courantes, une gestion électronique de la charge des batteries ? Cet article répond à ces questions.

Sans lithium

Plomb

Tout était si simple avec les batteries au plomb : vous injectez dans la batterie autant de courant que possible et vous arrêtez dès que la tension de fin de charge (que nous appellerons ici TFC, pour *seuil de tension de fin de charge*) est atteinte. Quant aux batteries composées de plusieurs cellules (en série), il suffit de multiplier leur TFC par le nombre de cellules et le tour est joué. Ça marche encore aujourd'hui, parce que les batteries au plomb supportent une certaine surcharge, en émettant tout au plus un peu de gaz. Si la capacité de l'une des cellules d'une batterie est inférieure au reste, rien ne vous empêche de charger toutes les cellules jusqu'au même seuil TFC. Pour transporter un ordinateur portable équipé de batteries au plomb, il faudrait un chariot. Il a donc fallu trouver autre chose. Dans l'automobile en revanche, leur robustesse expli-

que que presque toutes les autos restent équipées d'une anachronique batterie de 12 V au plomb, alors que d'autres types de batteries permettraient d'alléger le véhicule. Ces batteries sont si robustes qu'une batterie de (petite) voiture standard de 44 Ah résiste vaillamment pendant des années aux tortures infligées par un alternateur de 55 A et un régulateur rudimentaire. C'est assez impressionnant. La tension nominale d'une cellule est de 2 V. La TFC est d'environ 2,35 à 2,4 V/cellule, soit 14,1 à 14,4 V pour la batterie de voiture à six éléments. Une cellule de batterie au plomb ne doit pas être déchargée en dessous de 1,8 V. Une batterie de voiture sera donc endommagée si sa tension passe sous le seuil de 10,8 V, comme cela peut arriver avec une autodécharge d'environ 5 %/mois. Le motard en hibernation connaît bien ce problème saisonnier.

Nickel-Cadmium

Les piles rondes dites NiCd sont devenues populaires comme remplaçantes des piles zinc-carbone dans les appareils de grande consommation. Ce type de pile est interdit dans l'UE depuis plus de 10 ans en raison de la nocivité du cadmium. Les batteries NiCd ont maintenant disparu de la vie quotidienne.



Figure 1. Cellule NiMH typique au format AA avec autodécharge faible.



Figure 2. Batterie de l'ancienne Prius II de l'auteur. Vous pouvez voir 28 paquets de 6 cellules chacun soit 201,6 V de tension nominale.

Voici en bref leurs caractéristiques : les batteries NiCd tolèrent des courants d'assez forte intensité et de nombreux cycles de charge. Elles étaient souvent utilisées pour les modèles réduits de voitures, etc. Leur tension nominale est de 1,2 V par cellule. La charge est similaire à celle des batteries au plomb-acide jusqu'à des seuils TFC de 1,5 à 1,57 V/cellule (en fonction du courant). Une décharge profonde en dessous de 0,85 V/cellule endommage la batterie, même si celle-ci reste globalement beaucoup plus robuste qu'une batterie NiMH ou au lithium. L'autodécharge peut atteindre 20 % par mois. Pour la charge, la connexion en série de plusieurs batteries ne pose aucun problème.

Nickel-hydrure métallique

Les petites piles rondes NiMH, telles que le type AA (fig. 1) se sont substituées aux piles NiCd comme alternative non toxique. Elles sont disponibles en petites tailles, depuis des années dans les outils à piles p. ex. grâce leur prix bas et leur maniement facile. Il en existe aussi des modèles plus grands : le prototype de toutes les voitures hybrides, la Toyota Prius, était équipé sur les modèles II et III d'une batterie NiMH de près de cent livres, d'environ 200 V et d'environ 6,5 Ah. Sur ces voitures, cette batterie atteint des durées de vie et des nombres de cycles inouïs grâce à la gestion du comportement de charge et de décharge, et à l'utilisation d'une capacité partielle seulement. La figure 2 montre la batterie ouverte de mon ancienne Prius II. Une fois (suffisamment bien) résolus les problèmes liés à cette technique, Toyota et tous les autres fabricants sont passés au lithium pour leurs hybrides rechargeables et leurs modèles hybrides plus récents, en raison des économies de poids.

L'autodécharge des batteries NiMH est (en principe) similaire à celle des NiCd. En pratique, il existe cependant des variantes telles que la batterie de la figure 1, avec un taux d'autodécharge beaucoup plus faible, d'environ 1,5 %/mois, au détriment de la densité de puissance. Pour les petits appareils moins fréquemment utilisés, de telles piles sont toujours préférables, car les piles NiMH profondément déchargées deviennent inutilisables.

Lithium

Les piles à base de lithium prennent peu à peu toute la place, du fait de leurs meilleures performances (puissance, poids) et de leur faible autodécharge, de l'ordre de 5 %/an. Il n'y a pas de lumière sans ombre : les batteries dites li-ion (lithium-ion) sont très sensibles à la surcharge et à la décharge profonde. Non seulement elles perdent de leur capacité, mais certains types peuvent éclater ou prendre feu. À basse température, la capacité disponible diminue. Le phénomène est réversible, mais reste problématique sur les voitures électriques. Les batteries au li-ion ne supportent pas bien non plus les températures élevées.

En raison de la sensibilité à la surcharge, il convient de surveiller les tensions lors de la connexion en série. En tout état de cause,

il faut ici un „équilibreur“ supplémentaire (*balancer* en anglais), qui empêche la surcharge de certaines cellules dont la capacité serait par hasard un peu plus faible, tandis que d'autres cellules ne seraient pas encore complètement chargées. Et ce n'est pas tout. Les batteries li-ion nécessitent toujours un système de gestion de batterie (BMS pour *battery management system*) aux caractéristiques parfois complexes.

Ce qui complique encore le sujet, c'est la diversité des types de batteries au lithium. En fonction de la technologie utilisée et de la chimie interne, le nombre de cycles possibles, la TNC (pour *tension nominale de cellule*), la TFC, la TFD (pour *tension de fin de décharge*), le courant de décharge maximal et d'autres paramètres varient. Le BMS doit être adapté à ces paramètres et bien sûr aux courants et aux conditions de température à contrôler dans le cas spécifique. En dehors des voitures électriques, pour lesquelles des conditions spéciales s'appliquent et dans lesquelles d'énormes efforts sont faits pour protéger les batteries coûteuses, c'est heureusement un peu plus facile avec les systèmes de batteries plus petits, par exemple pour les appareils portables tels que les ordinateurs portables ou les HP Bluetooth, les outils tels que les tournevis sans fil ou les appareils ménagers tels que les aspirateurs sans fil, car le refroidissement et le chauffage ne sont pas nécessaires. D'ailleurs, le plus simple est de le faire avec des smartphones, car ils ne contiennent pas de piles connectées en série, ce qui explique pourquoi aucun équilibreur n'est nécessaire. L'électronique de charge intégrée ne doit prendre en compte que la TFC et la TFD ainsi que le courant de charge maximal.

Parmi les petites piles li-ion, deux catégories dominent : il existe des types dont les électrodes sont construites avec les matériaux suivants : cobalt, dioxyde de cobalt, manganèse et/ou nickel et aluminium. La technologie LiCoO₂, qui a fait ses preuves, se taille la part du lion dans ce domaine. Heureusement pour tous ces types, la TNC est d'environ 3,6 à 3,7 V, la TFC de 4,2 V et la TFD de 2,5 V (à l'exception du LiNi_xCo_yAl_zO₂, dont la TFD est de 3,0V). Même si les courants autorisés et la durée de vie sont différents, tous ces types peuvent en principe être gérés avec les mêmes procédures et donc les mêmes circuits intégrés.

La batterie est composée de phosphate de fer et de lithium (LiFePO₄), avec les caractéristiques suivantes : TNC = 3,2 V, TFC = 3,65 V et TFD = 2,5 V. Ce type de batterie offre un nombre de cycles élevé et une densité de puissance légèrement inférieure. Il faut donc des équilibreurs au moins pour ces tensions. L'industrie des semi-conducteurs propose bien sûr des solutions pour les deux catégories et pour divers nombres de cellules et courants connectés en série – certains circuits intégrés sont même programmables, commutables pour certains paramètres ou possèdent une interface pour les microcontrôleurs.

Les piles li-ion sont disponibles en plusieurs modèles : Les types li-po (lithium polymère) sont principalement utilisés dans les

appareils mobiles et les ordinateurs portables en raison de leurs formes possibles et de leur planéité. Le terme *polymère* rappelle simplement que les li-po sont scellées dans un film de plastique au lieu d'un boîtier rigide (ne pas couder !). Il y a ensuite diverses cellules rondes, dont le type particulièrement courant 18650 ($\varnothing = 18$ mm, longueur = 65 mm), qui, chose étonnante, est installé par milliers en paquets dans les voitures électriques Tesla – l'espace entre les cellules rondes sert à réguler la température. La **fig. 3** montre une cellule 18650 provenant de la production chinoise avec des spécifications fantaisistes : sur papier tout comme sur le plastique et en HTML, on peut dire n'importe quoi : 7 800 mAh ne correspondent pas, et de loin, au format de la cellule 18650. Les meilleures cellules sont environ deux fois plus petites en termes réels. Enfin, il y a les cellules dites prismatiques, dont la désignation indique simplement que leur logement n'est pas rond mais prismatique. De telles cellules sont par exemple utilisées dans la mini voiture *Hotzenblitz* de production allemande. La **fig. 4** montre que chaque cellule est équipée d'un système électronique supplémentaire de surveillance.

Équilibreur

La notion de *gestion de la batterie* est assez vague. Elle englobe tout ce qui concerne de près ou de loin la surveillance et le contrôle de la charge, de la décharge et de l'état de fonctionnement des batteries. Alors que dans le cas des batteries plomb-acide, NiCd et NiMH, cela ne concerne que la conformité avec les TFC, TFD et, occasionnellement, la limitation des courants, la question est plus complexe en raison des problèmes de connexion en série des cellules de batteries au lithium. Dans ce cas, le BMS a nécessairement besoin d'un équilibreur s'il y a plus d'une cellule !

Les piles au lithium n'aiment pas la surcharge. Et comme la capacité des cellules d'un même lot varie légèrement, même en cas de production de masse, et que des différences peuvent également apparaître au fil du vieillissement, il y aura toujours, dans une connexion en série, une cellule pleinement chargée avant les autres. À ce stade, cependant, la TFC pour l'ensemble de la batterie n'est pas encore atteinte, et si la charge devait se poursuivre, la cellule déjà chargée serait forcément endommagée – lors de la prochaine charge, elle serait „prête“ encore plus tôt ; après quelques cycles, ce cercle vicieux conduirait à une défaillance totale de cette cellule et donc de la batterie entière. Un équilibreur a pour vocation d'éviter cela.

Dans la version la plus simple pour une batterie à deux cellules (**fig. 5**), l'équilibreur surveille les tensions des deux cellules. Si l'une des deux cellules atteint le seuil TFC, une charge faible appropriée est ajoutée en parallèle avec cette cellule, de sorte que le courant de charge excédentaire passe ne passe pas dans cette cellule, mais à côté, et que l'autre cellule, dont la capacité est meilleure, puisse encore être chargée entièrement. Avec plus de cellules, l'équilibreur sera dimensionné en conséquence. Le petit tableau (**fig. 6**) montre un BMS pour quatre (petites) cellules proposées sur eBay pour différents nombres de cellules à des prix compris entre 1 € et 10 €. La **fig. 7** rend compte de mes efforts pour équilibrer la batterie LiFePo 48 V de mon clone *Segway* : chaque cellule est surveillée par son propre microcontrôleur.

Ça paraît lourd en théorie, mais pas en pratique. En termes purement formels, la charge connectée en parallèle en cas d'urgence devrait pouvoir faire passer en douce le courant de charge complet en dehors de la cellule à protéger, mais dans la pratique, ce courant



Figure 3. Pile au lithium typique au format 18650 avec une spécification de capacité outrancière. Cette cellule est dotée d'une électronique intégrée pour éviter les décharges profondes et les surcharges.

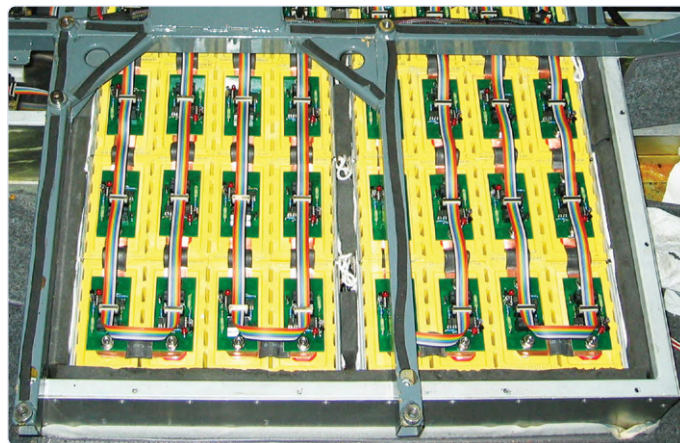


Figure 4. Partie d'une batterie LiFePo de la voiture miniature allemande *Hotzenblitz* avec électronique de gestion de la batterie (Source : Joes-Wiki, CCASA 3.0 FR [2]).

est bien inférieur au courant de charge maximum susceptible de circuler. La raison en est que les batteries li-ion sont initialement chargées avec le courant nominal jusqu'à un état de charge de 50 à 80%, mais à partir de là, seulement avec un faible courant, généralement en diminution constante. Avec les piles LiCoO₂, pendant la charge, le courant est assez faible au-delà de 3,95 V. Comme les différences entre cellules saines ne sont pas si grandes, il suffit

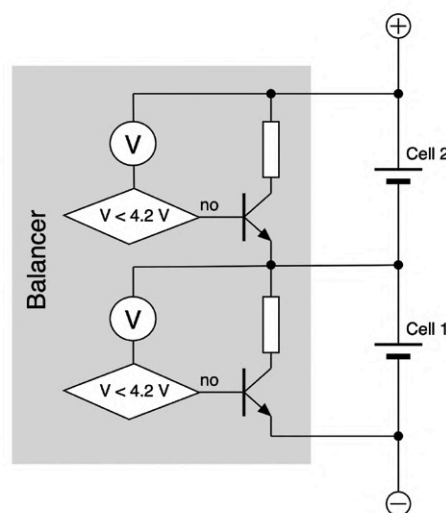


Figure 5. Schéma d'un simple équilibreur à deux cellules.

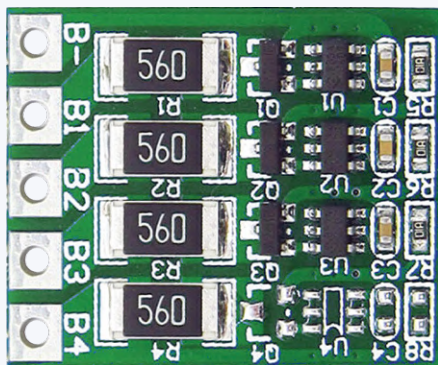


Figure 6. On trouve sur eBay des équilibreurs de ce type dans toutes les variantes et versions à un prix modique.



Figure 7. Batterie LiFePo avec équilibreur du clone Segway de l'auteur.

de décharger ce courant beaucoup plus faible. En pratique, des charges supplémentaires pour environ 1 à 5 % du courant maximal sont presque toujours suffisantes. Si cela ne suffit pas, la cellule est *défectueuse* et un ballast plus élevé ne serait pas utile à long terme. Selon les courants qui circulent, un équilibreur gaspille une énergie qu'il serait préférable d'utiliser. Il existe pour cela des équilibreurs spéciaux qui ne convertissent pas simplement en chaleur l'énergie excédentaire de la cellule à protéger, mais la réinjectent directement dans le système global par le biais de régulateurs à découpage (par cellule !). Il serait surprenant toutefois que vous rencontriez souvent un tel souci du minuscule.

Un autre critère important : les équilibreurs sont basés sur des composants électroniques et le choix de circuits intégrés bon marché pour hautes tensions est limité. Par conséquent, les possibilités de mise à l'échelle butent sur des limites pratiques. Les circuits intégrés modernes de BMS sont généralement disponibles pour 2 à 12 cellules. Avec 12 cellules, vous avez déjà une tension de fonctionnement d'environ 50 V. Si vous voulez plus de cellules dans une

batterie, il vous suffit de mettre à l'échelle l'électronique du BMS. Avec les systèmes sous 800 V des voitures électriques modernes, ça vous fera une vingtaine de modules avec des BMS-IC 12s en série (le „s“ après 12 correspond à „cellule“). Les différents sous-systèmes sont ensuite reliés entre eux par un bus (isolé électriquement) et généralement à un système de commande central.

Circuits intégrés

Il existe une grande variété de circuits intégrés spéciaux pour toutes les applications imaginables – trop nombreux pour espérer en donner un aperçu utile ici. Au lieu de cela, nous décrivons un circuit intégré précis en guise d'exemple. Avant cela, précisons qu'en plus de l'équilibrage, un BMS a également pour tâche, dans la plupart des applications avec des batteries au lithium, de protéger la batterie contre les sous-tensions et les surintensités. Parfois, le circuit de charge est également inclus dans le système. Pour ces tâches complexes, l'utilisation d'un microcontrôleur tombe sous le sens. Les BMS modernes ont donc parfois un petit SoC intégré ou une interface équivalente. Les circuits intégrés de BMS ne figurent donc pas seulement au catalogue des fournisseurs habituels de circuits analogiques ou de régulateurs à découpage, mais aussi à celui d'entreprises telles que *Microchip*, *Maxim* (**fig. 8**) ou *Renesas*. C'est d'ailleurs de *Renesas*, connu pour ses microcontrôleurs, que vient le circuit que nous examinerons.

Le schéma fonctionnel (**fig. 9**) du BMS intégré ISL94212 réunit sur la gauche les connecteurs pour la masse (en bas) et la borne positive et la tension de la batterie (en haut) ainsi que les nombreux connecteurs pour douze piles au lithium en tout. Les tensions proviennent du tampon d'entrée et adaptateur de niveau (*Input Buffer / Level Shifter*) via un commutateur (VC MUX) et un autre sélecteur (MUX) vers un convertisseur A/N. En outre, on remarque des tensions de référence, des unités thermométriques, etc., le circuit de surveillance de la TFD et bien sûr le plus important, le bloc numérique de commande et de communication (*Control Logic and Communications*).

Le circuit de base de l'application (**fig. 10**) montre le circuit intégré en détail. À gauche, la connexion des cellules, et à droite du circuit intégré ISL94212 de gauche, un microcontrôleur qui commande l'ensemble du BMS via une interface numérique. Si vous souhaitez gérer plus de 12 cellules, il faudra des modules supplémentaires, avec un ISL94212 pour chacun. Les modules sont couplés en série : il y a donc deux broches DHi et DLo deux fois par circuit intégré (numérotés 1 et 2). En raison des tensions élevées de ces chaînes

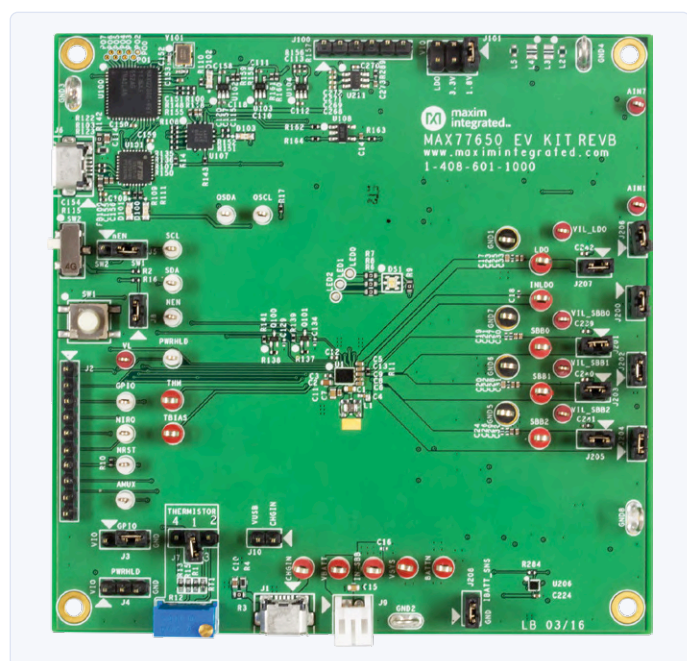


Figure 8. Kit d'évaluation pour les circuits intégrés BMS MAX77650/ MAX77651 (Source : Maxim Integrated).

de modules, les interfaces série ne sont pas couplées directement, mais par des condensateurs à faible capacité.

Le circuit intégré assure non seulement la surveillance de nombreuses cellules, mais aussi l'équilibrage nécessaire. Le principe de connexion d'une résistance de charge (à l'extrême gauche) à chaque cellule par un MOSFET de faible puissance (**fig. 11**) est commandé par la sortie CBX correspondante du circuit intégré. De cette manière, l'équilibrage peut également être assuré pour des courants d'assez forte intensité. Les détails des nombreuses fonctions du circuit intégré se trouvent dans sa fiche technique [1].

Pratique

Si vous avez affaire à une batterie au lithium, vous aurez le choix entre trois stratégies : pour les applications simples, il suffit d'acheter une petite carte chinoise, bon marché, assemblée (**fig. 5**). S'il s'agit d'une batterie plus grande et/ou plus chère, vous avez le choix parmi de nombreux BMS prêts à l'emploi de fabricants établis. Il est possible enfin de trouver une solution sur mesure pour les batteries que vous souhaitez gérer, ce ne sont pas les circuits intégrés appropriés qui manquent. Dans certains cas, il faudra écrire votre propre code pour surveiller et commander la charge des batteries avec précision. Dans tous ces cas, Google finira par vous mettre sur la bonne piste. ❏

180350-02

Ont contribué à cet article :

Texte & illustrations : **Thomas Scherer**

Rédaction : **Jens Nickel**

Traduction : **Dominique Escarlata**

Maquette : **Giel Dols**

Votre avis, s'il vous plaît...

Vous pouvez adresser vos questions et vos commentaires à redaction@elektor.fr



DANS L'E-CHOPPE D'ELEKTOR :

- **Intelligent Charge Controller (12/24 V, 10 A)**
www.elektor.fr/19126
- **Multimètre OWON XDM2041 (55000 points)**
www.elektor.fr/19312
- **Enregistreur de données RS232 (140126-41)**
www.elektor.fr/170181

LIENS

- [1] **Fiche technique ISL94212 de Renesas** : www.renesas.com/en-us/www/doc/datasheet/isl94212.pdf
 [2] **CCSA 3.0 DE** : <https://creativecommons.org/licenses/by-sa/3.0/de/deed.en>

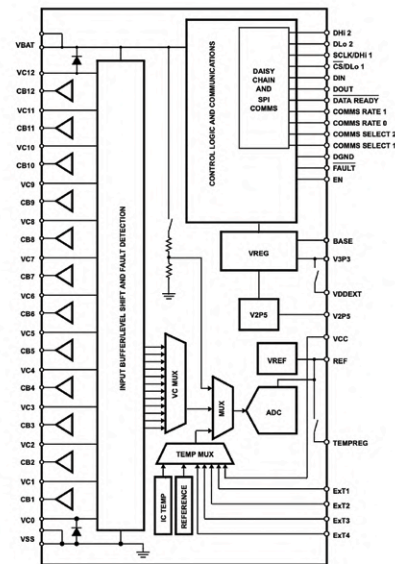


Figure 9. Synoptique du circuit intégré BMS ISL94212 (Source : Renesas).

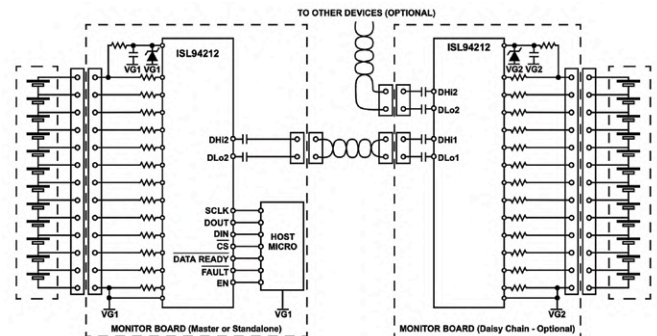


Figure 10. Configuration de principe lors de la montée en charge de la BMS à plus de 12 cellules avec l'ISL94212 (Source : Renesas).

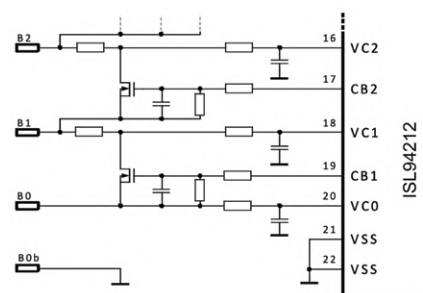
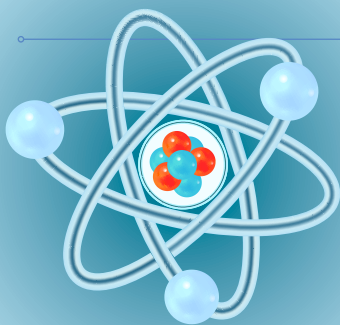


Figure 11. L'ISL94212 utilise des MOSFET externes pour l'équilibrage, afin de connecter une résistance de charge en parallèle sur chaque cellule.



démarrer en électronique (5)

...est moins difficile qu'on ne l'imagine !

Dans l'épisode précédent de cette série d'articles, nous avons examiné le circuit en H et remarqué qu'il est préférable de ne pas ou pas toujours négliger la résistance des conducteurs dans le monde réel. Il est temps de se pencher sur le courant alternatif.

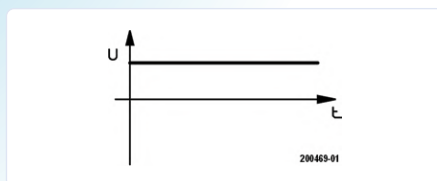


Figure 1. Tension continue.

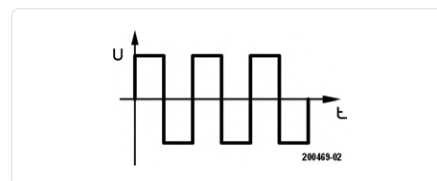


Figure 2. Tension alternative (ici en forme de rectangle).

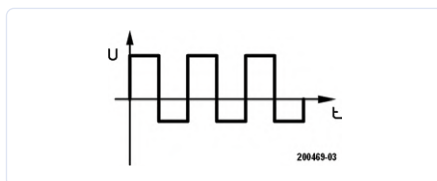


Figure 3. Une tension alternative superposée à une tension continue.

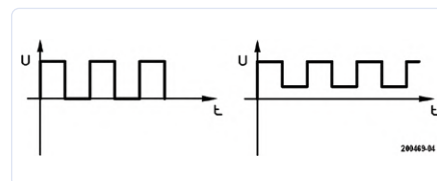


Figure 4. Tension continue pulsée.

Ont contribué à cet article :

Idée et illustrations : **Michel Ebner**

Rédaction : **Eric Bogers**

Traduction : **Alice Coper**

Maquette : **Giel Dols**

Tous les exemples et calculs des épisodes précédents de cette série d'articles étaient basés sur une *tension continue*. Les choses se corsent avec une *tension alternative*. Un amplificateur audio, par exemple, n'amplifie que des tensions alternatives.

Tout le monde sait qu'une pile délivre une tension dont la polarité ne change pas, c'est une tension continue. L'*amplitude* de cette tension peut changer, elle diminue jusqu'à ce que la batterie soit entièrement déchargée, mais la *polarité* ne change pas. La **figure 1** montre un exemple de tension continue.

Avec une tension alternative, en revanche, la polarité change périodiquement. L'amplitude de la tension peut changer aussi (voir ci-dessous), mais pas forcément, comme le montre la **figure 2**. C'est une onde carrée, une tension alternative, mais ce n'est pas son niveau qui change, seulement sa polarité. La valeur *moyenne* de cette tension alternative, considérée sur une période plus longue, est égale à zéro. Il s'agit alors d'une tension purement alternative.

Il est concevable que la valeur *moyenne* d'une tension alternative ne soit pas nulle, bien que la polarité change sans cesse. Dans ce cas, nous avons une tension alternative *superposée* à une tension continue, en d'autres termes, la tension en question est la *somme* d'une tension alternative pure et d'une composante continue (**fig. 3**).

La **figure 4** montre une situation similaire, à un gros détail près. Le niveau change périodiquement, mais la polarité reste la même. C'est une tension continue dite *pulsée*.

Durée et fréquence de la période

La durée d'une période *complète d'une* tension alternative est appelée la durée de la période *T* de cette tension alternative et est donnée en secondes. La réciproque de cette durée de période est la fréquence *f*, exprimée en hertz (Hz) :

$$f = \frac{1}{T}$$

En Europe, la fréquence du réseau est de 50 Hz (60 Hz outre-atlantique). La durée de la période de la fréquence du réseau de 50 Hz est

$$T = \frac{1}{f} = \frac{1}{50\text{Hz}} = 0,02\text{s} = 20\text{ms}$$

Tension alternative sinusoïdale

La tension alternative qui nous intéresse le plus est la tension alternative sinusoïdale (fig. 5). L'évolution dans le temps de cette tension est déterminée par la formule :

$$U(t) = U_{top} \cdot \sin\left(\frac{2 \cdot \pi \cdot t}{T}\right) = U_{top} \cdot \sin(2 \cdot \pi \cdot f \cdot t) = U_{top} \cdot \sin(\omega \cdot t)$$

La valeur la plus élevée atteinte par une tension alternative est appelée **valeur de crête**. Vous ne serez pas non plus surpris d'apprendre qu'une tension alternative qui n'atteint sa valeur de crête que **périodiquement** est moins **efficace** (c'est-à-dire qu'elle ne pourra pas développer autant de puissance) qu'une tension **continue** ayant la même valeur de crête. C'est pour pouvoir comparer le courant continu au courant alternatif que l'on a forgé la notion de **valeur efficace** :

La valeur efficace d'une tension alternative correspond à la valeur efficace de la tension continue, qui libère dans une résistance ohmique la même quantité de chaleur que la tension alternative correspondante.

Ainsi, une ampoule brille-t-elle autant avec une tension continue de 12 V qu'avec une tension alternative de 12 V_{eff}. Remarque : lorsque la valeur d'une tension alternative est donnée, il s'agit toujours de la valeur efficace, sauf indication contraire explicite.

À ce stade, nous pourrions solliciter votre amour des mathématiques pour administrer la preuve de cette correspondance. Dans le cadre de cet article, nous nous contenterons d'affirmer qu'il existe la relation suivante entre la valeur de crête et la valeur efficace d'une tension alternative sinusoïdale, que vous devrez croire sur parole :

$$U_{top} = U_{eff} \cdot \sqrt{2}$$

Nous voici armés pour calculer une tension alternative bien connue de tous depuis longtemps : la tension alternative de 230 V du réseau électrique. Nous connaissons sa valeur efficace, mais quelle est donc sa valeur de crête (*crest factor CF*) ?

$$U_{top} = U_{eff} \cdot \sqrt{2} = 230V \cdot 1,41 = 324,3V$$

Facteur de crête

Une dernière explication avant de laisser la théorie pour nous concentrer sur la pratique de l'électronique. Le rapport entre valeur de crête et valeur efficace d'une tension alternative est appelé **facteur de crête** :

$$CF = \frac{U_{top}}{U_{eff}}$$

Pour une tension carrée (fig. 2), le facteur de crête est de 1. Pour une tension d'onde sinusoïdale, il est de $\sqrt{2}$. Pour un signal triangulaire, de $\sqrt{3}$. Il convient de mentionner que le facteur de crête pour un signal de bruit est infini, puisque plus le bruit est « aléatoire », plus le facteur de crête est élevé. Il est peu probable que vous ayez un jour besoin de cela, mais ne vaut-il pas mieux le savoir que l'ignorer ?

Entre théorie et pratique...

Entre théorie et pratique, il y a les schémas et les liste de composants ! On ne construit pas un circuit électronique en vidant sur la table une boîte de composants pour les assembler ensuite au hasard. Au contraire, vous réfléchissez soigneusement à ce que vous voulez, puis vous faites sur papier un schéma de ce qui pourrait devenir le circuit final. Vous l'essayez d'abord sur une carte d'expérimentation, et après avoir apporté les améliorations nécessaires, vous finissez par arriver au schéma final.

À partir de ce schéma, vous dessinez les pistes d'un circuit imprimé, avec son plan d'implantation des composants. Il existe différentes variantes de modes de câblage des composants pour arriver à un circuit final qui marche.

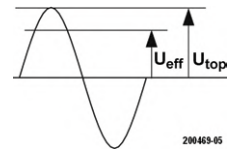


Figure 5. Une période complète d'une tension alternative sinusoïdale.

La liste des composants est indispensable, notamment pour réunir tous les composants requis, soit qu'ils proviennent de vos réserves, soit que vous les commandiez.

Si vous parcourez cette édition (ou toute autre) du magazine Elektor, vous trouverez de nombreux exemples de schémas, de dessins de circuits imprimés, de listes de composants, etc. Si vous souhaitez vous faire une idée complète de ce qu'implique la réalisation d'un produit électronique destiné à la vente, je vous renvoie à l'excellente série d'articles *bureau d'études - zone D — De l'idée au produit* de Clemens Valens, dont le premier épisode avait été publié dans le numéro de septembre/octobre 2019 d'Elektor. La série se termine dans cette édition.

Dans notre prochain article, nous reviendrons sur les éléments tangibles dont vous aurez à vous occuper en pratique. ◀

200469-04

La série d'articles *démarrer en électronique* est basée sur le livre « *Basic Course Electronics* » de Michael Ebner, publié par Elektor.

Votre avis, s'il vous plaît ...

Vous pouvez adresser vos questions ou vos commentaires à l'auteur de cet article par courriel : redaction@elektor.fr



DANS L'E-CHOPPE D'ELEKTOR

- **Basic Electronics for Beginners (PDF)**
www.elektor.fr/basic-electronics-for-beginners-e-book
- **L'électronique pour les débutants**
www.elektor.fr/l-electronique-pour-les-debutants



elles sont petites, mais font de grandes choses

Les pépites d'Elektor

Voici quelques petits circuits pour mettre à jour vos compétences en matière de soudure et d'expérimentation si vous n'avez pas envie de vous lancer dans quelque chose de très compliqué.



idée : Michael A. Shustov (Russie)

Pseudo-thyristor

Un thyristor conventionnel fonctionne comme un interrupteur. C'est assez simple : il s'allume dès l'application (brève) d'une tension. Pour le désactiver, il faut interrompre son alimentation. Notre pseudo-thyristor (**fig. 1**) fonctionne autrement. Il est activé en connectant brièvement l'entrée à la tension d'alimentation positive, puis désactivé en connectant brièvement la même entrée à la masse. Son fonctionnement n'est pas difficile à comprendre. Lorsque la tension d'alimentation est activée, le transistor Q4 se bloque, tandis que Q1 et Q3 conduisent et que le commutateur Q2 se bloque : la charge (ici la lampe EL1) est éteinte. La LED D1 indique que le circuit est armé.

Si on appuie sur le bouton SB1, l'entrée du circuit passe brièvement à un niveau haut, Q4 devient conducteur. Ainsi, la base de Q3 est reliée à la masse et ce transistor se bloque. En conséquence, Q2 conduit et active la charge. Cette situation perdure même une fois SB1 relâché.

Si on appuie sur SB2, le circuit revient à son état initial et la charge est à nouveau coupée.

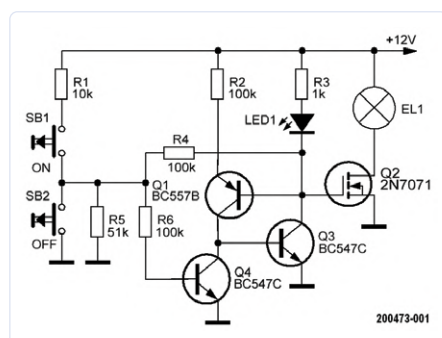


Figure 1. Avec quelques transistors, on peut construire un pseudo thyristor, dont les propriétés sont meilleures que celles d'un «vrai» thyristor.

Si l'on appuie simultanément sur SB1 et SB2, SB2 est prioritaire : dans ce cas aussi, la charge est désactivée.



idée : Roel Arits (Pays-Bas)

Étincelles mélodiques

Noël est une période passionnante pour les bricoleurs amateurs de loupottes en tous genres. Avec ce circuit, nous faisons monter la tension. Jusqu'à 30 kV pour être précis. Et pour quoi faire ?

Pour faire des étincelles. Des étincelles musicales, qui s'accompagnent de petites mélodies comme *Jingle Bells* ou *Mission Impossible* si vous préférez. Le circuit est étonnamment simple (**fig. 2**).

Le maître de cérémonie est un Arduino Pro Mini, qui délivre un signal PWM (en fait, c'est un signal carré ordinaire) dont le rapport cyclique fixe est de 10 %. Ce signal est porté à 12 V puis tamponné. Un transformateur spécial le convertit ensuite en une tension d'environ 30 kV, assez pour produire de beaux éclairs. Pour obtenir la mélodie promise, le logiciel de l'Arduino [6] fait varier la fréquence du signal carré pour reproduire les notes de l'une des deux mélodies fixes programmées. Le résultat peut être vu et entendu en vidéo [1].

La construction du circuit proprement dit n'est pas critique, seul le transformateur demande un peu de doigté. L'enroulement primaire compte 20 tours de fil de cuivre émaillé de 0,7 mm au bout de deux bâtons de ferrite adjacents d'un diamètre de 10 mm et d'une longueur de 80 mm. Pressés l'un contre l'autre, ces deux bâtons s'insèrent exactement dans l'ouverture d'une bobine de 16 kV finie, qui sert d'enroulement secondaire (**fig. 2 et 3**).

Note : La bobine utilisée par l'auteur n'est plus disponible. Il existe une alternative [3], déjà équipée d'un primaire, de sorte que vous

n'avez pas à le faire vous-même. Il existe une autre possibilité [4], mais dans ce cas c'est vous qui bobinez vous-même l'enroulement primaire. Et enfin une dernière possibilité [5] avec laquelle la tension de l'enroulement primaire devra peut-être être abaissée. Dans tous les cas, il faudra peut-être tâtonner un peu pour déterminer le rapport cyclique optimal pour obtenir un maximum d'étincelles fortes sans amener le noyau à saturation.

Ce générateur d'éclairs, certes rudimentaire, peut être illégal car émetteur de parasites. Si vous en faites un usage modéré, vous ne devriez pas avoir de problèmes. Un petit récepteur AM vous permettra d'évaluer la portée des interférences susceptibles de perturber votre voisinage. Faites preuve de sens civique.

Le logiciel pour ce projet et une version plus grande de la figure 2 peuvent être téléchargés gratuitement sur la page de notre site associée à l'article [6].



idée : Elektor Labs

Détecteur de mensonges

Les vrais polygraphes utilisent divers capteurs placés sur le corps de la personne interrogée et mesurent des paramètres tels que la pression sanguine, le rythme cardiaque, la respiration, la résistance de la peau et la température. Le détecteur transcrit ces valeurs sur une bande de papier et, à partir de ces données, un opérateur essaye de déterminer si la personne interrogée a menti ou non. L'utilisation des détecteurs de mensonges est controversée.

Voici un circuit simple qui mesure les changements de résistance de la peau et peut être utilisé comme détecteur de mensonges ludique. Un ohmmètre ordinaire

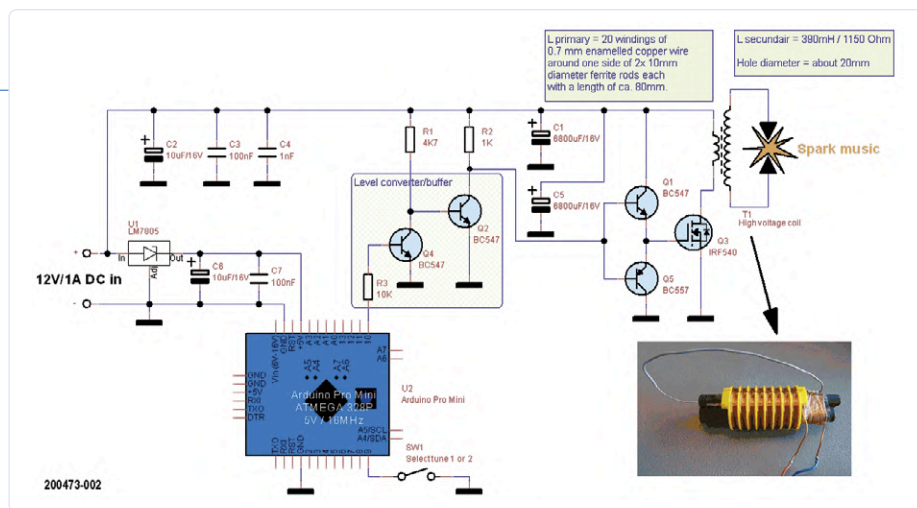


Figure 2. Le cœur du générateur d'étincelles musicales est un Arduino Mini Pro.

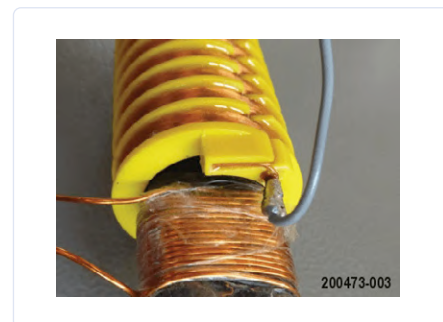


Figure 3. L'enroulement primaire est fermement arrimé à deux bâtons de ferrite, qui à leur tour s'insèrent dans la bobine haute tension prête à l'emploi.

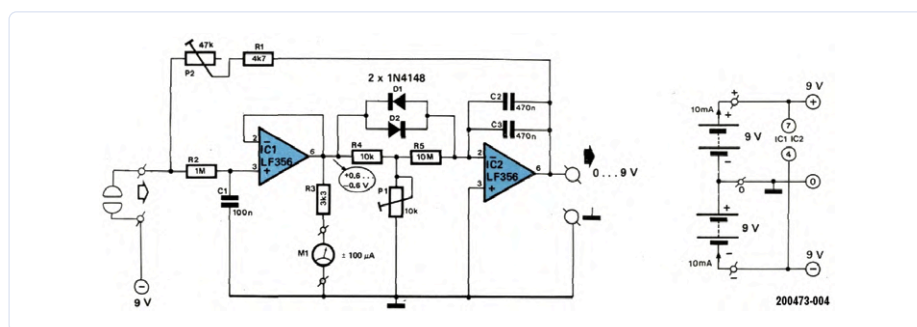


Figure 4. Un simple détecteur de mensonges qui montre les changements dans la résistance de la peau.

ne convient pas, car il ne s'agit pas de mesurer la valeur absolue de la résistance de la peau, mais son évolution.

Sur le côté gauche du schéma (fig. 4), nous voyons (par exemple) deux électrodes posées sur le dos de la main et l'avant-bras du sujet. Une tension négative de 9 V est appliquée à l'électrode inférieure. À travers la résistance de la peau et R2, cette tension règne aussi sur l'entrée non-inverseuse et à très haute impédance de l'ampli op IC1. Presque aucun courant ne circule par cette entrée, de sorte que le courant qui passe par la peau doit passer par P2 et R1 vers IC2. Cet ampli op ajuste sa tension de sortie de manière à ce que l'électrode supérieure soit toujours à un potentiel de 0 V.

La valeur élevée de R5 et la connexion en parallèle de C2 et C3 introduisent ensemble un certain retard. Les grands sauts de tension sont rapidement éliminés par les deux diodes. Pour les faibles différences de tension, le retour à 0 V de la tension sur l'électrode supérieure peut durer jusqu'à 30 s. De cette façon, une modification de la résistance de la peau peut être facilement lue sur l'instrument à bobine mobile M1 dont la position de repos est au milieu de l'échelle. Le filtre passe-bas formé par R1 et C2/C3 élimine ronflements et autres parasites.

Le réglage n'est pas difficile : apposez les deux électrodes (à vous de trouver un

matériau adéquat) sur la peau du sujet testé puis connectez un multimètre à la sortie du circuit (calibre : 10 V CC). Tournez ensuite le curseur de P2 jusqu'à ce que le multimètre affiche une tension entre 2 et 6 V (attendez la stabilisation de la valeur mesurée). Le détecteur est alors calibré pour le type de peau de ce sujet. Appuyez maintenant légèrement sur les électrodes, l'aiguille de M1 devrait bouger. Le détecteur est prêt à détecter.

Attention : ce circuit ne doit être alimenté que par deux piles de 9 V. N'utilisez en aucun cas une alimentation secteur !

200473-04

LIENS

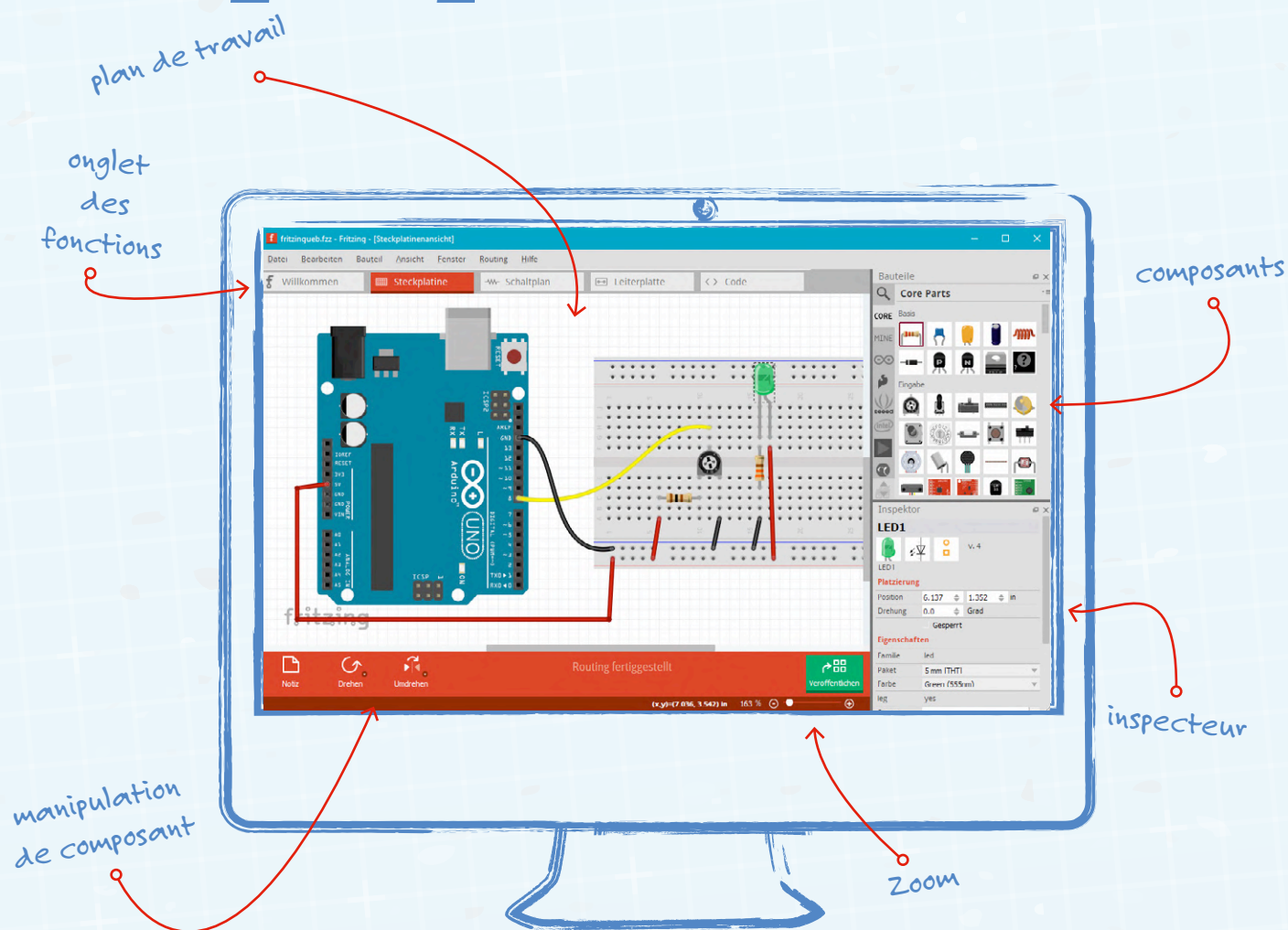
- [1] Vidéo Jingle Bells : www.youtube.com/watch?v=NGSjcOffbul&feature=youtu.be
- [2] Bobine HT originale : <https://bit.ly/3IRr8R7>
- [3] Bobine HT alternative n°1 : <https://bit.ly/35crVGo>
- [4] Bobine HT alternative n°2 :: <https://bit.ly/2FfsCU4>
- [5] Bobine HT alternative n°3 : <https://amzn.to/3jPcAiX>
- [6] Page de cet article : www.elektormagazine.fr/200473-04



DANS L'E-CHOPPE
D'ELEKTOR

> Livre (en anglais)
"Electronic circuits for all"
www.elektor.fr/electronic-circuits-for-all

Fritzing : graphisme d'implantation sur plaques d'essais



Florian Schäffer

Le chemin qui mène d'un schéma à un prototype assemblé sur plaque d'essais n'est pas toujours semé de roses : les lignes abstraites du schéma doivent être transformées en composants concrets et en connexions réelles. Le logiciel *Fritzing* permet de créer des plans d'implantation réalistes, qui simplifient ensuite l'assemblage du circuit et constituent une bonne documentation du projet.



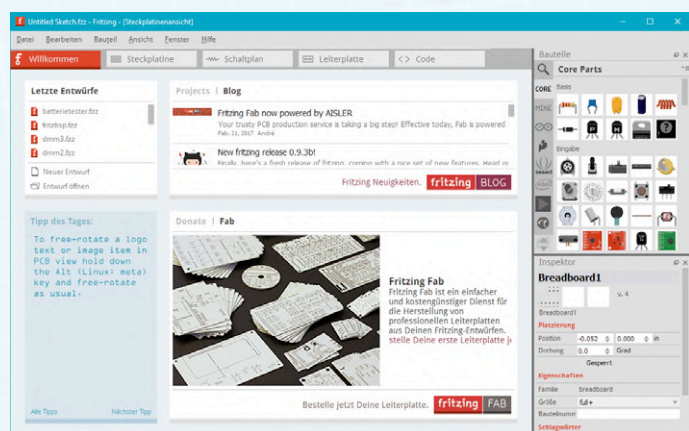
En principe, pour la documentation d'un projet, une photo du montage devrait aussi faire l'affaire, mais il faut un certain savoir-faire pour obtenir une bonne photo, avec le bon éclairage, le bon angle, la netteté, la profondeur de champ, etc.. Souvent le résultat est décevant. Avec le logiciel *Fritzing*, vous créez facilement des images de plaques d'essais à la fois lisibles et attrayantes, à mi-chemin entre abstraction et réalisme. Les composants sont réduits à l'essentiel, leurs connexions sont bien lisibles. Il est toujours préférable, avant d'assembler les composants, de les disposer sur une planche à dessin virtuelle pour en déterminer l'agencement optimal.

Logiciels alternatifs

Depuis son apparition, Fritzing est très populaire : gratuit, simple, il donne des résultats agréables. Le programme offre des fonctions supplémentaires, mais moins utiles, notamment le dessin de schémas, de circuits imprimés et de code source. Dans le prochain article, nous nous familiariserons avec *Tinkercad Circuits*, un simulateur de circuits et logiciel d'agencement fournissant de belles images des montages assemblés. Dans *Fritzing*, tout utilisateur peut créer de nouveaux composants à son catalogue d'objets graphiques intégrés, mais ce n'est pas permis dans *Tinkercad Circuits*.

Le programme est disponible gratuitement depuis fin 2019, son téléchargement est caché derrière une demande de paiement. Pour contourner ce problème (en toute légalité), vous pouvez rechercher une ancienne version ; dès la première exécution, vous serez invité à la mettre à jour. Vous pouvez aussi télécharger directement la version à jour (0.9.4), avec le lien <https://fritzing.org/download/api/1.0/update/344>.

Après l'avoir installé, lancez *Fritzing* : il recherchera les mises à jour du programme et des bibliothèques qui le composent. La langue de l'interface utilisateur est sélectionnée automatiquement en fonction de la configuration du système. Pour la plupart des lecteurs, ce sera l'anglais par défaut. En haut de l'écran de bienvenue, vous verrez une rangée d'onglets correspondant aux fonctions disponibles. Commençons par la fonction *Breadboard* en cliquant sur ce bouton.



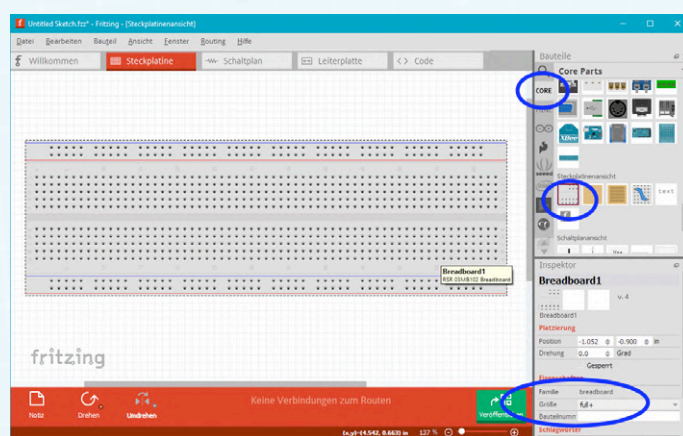
L'écran de bienvenue de Fritzing. Cliquez sur Breadboard.

Formats d'exportation

Fritzing connaît divers formats de fichiers. La fonction *File/Save* enregistre les fichiers dans leur format natif, avec une extension *.fzz* ; ces

fichiers peuvent être rouverts pour les retravailler. La fonction *File/Export* donne le choix d'autres formats divers. Parmi les plus utiles, citons les formats d'image : le format PNG donne une image binaire, idéale pour les sites ; contrairement au format JPG, les images au format PNG ne subissent aucune compression. Le format vectoriel SVG est le plus approprié lorsque l'illustration doit encore être mise à l'échelle ou modifiée à l'aide d'un programme de traitement graphique. Chaque éditeur graphique a tendance à interpréter les SVG à sa manière, ce qui peut entraîner des différences selon le programme utilisé. On a vu disparaître des composants entiers à cause d'erreurs dans la bibliothèque de composants de *Fritzing*. Pour cette raison, je recommande d'exporter au format PDF que de nombreux éditeurs d'images vectorielles sont capables de traiter.

Choisissez une plaque d'essais

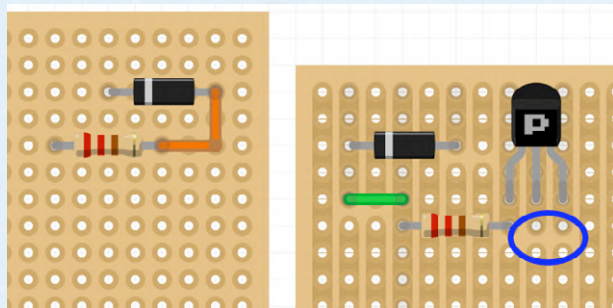


Am Anfang wird stets eine große Steckplatine bereitgestellt, deren Größe Sie aber ändern können. Mit dem Mauseisen können Sie die Ansicht zoomen und mit der mittleren Maustaste verschieben.

Fritzing commence toujours par une plaque d'essais (*breadboard*) dont la taille peut être ajustée. Vous pouvez aussi la supprimer. La molette de la souris permet d'agrandir ou de réduire la vue, et le bouton central de la souris permet de faire un déplacement panoramique. Cliquez sur un composant pour le sélectionner et afficher ses attributs dans l'inspecteur à droite. Si vous avez réglé la grande plaque par défaut, vous pouvez utiliser l'attribut *Size* pour changer de taille de plaque comme cela vous convient. Si vous souhaitez ajouter une plaque à votre dessin, vous le trouverez dans la rubrique *Core* de la liste de composants *Parts list*. Les nouveaux composants peuvent être déposés directement sur le plan de travail.

Vue du circuit imprimé

Une fonction moins connue et pourtant très utile de *Fritzing* est l'agencement de plaques à pastilles ou à bandes de cuivre. Elle vous permet d'étudier l'agencement de vos composants sur une plaque d'expérimentation **avant de les souder**. Le programme vous indique où couper les bandes et quelles liaisons établir. À côté du symbole de carte d'expérimentation dans la liste des composants, il y a deux symboles marron pour deux types de cartes : le premier représente une carte à pastilles de cuivre à souder ; le second représente une telle carte d'expérimentation, mais celle-ci est à bandes. Avec l'Inspecteur, vous vérifierez votre choix et modifierez au besoin le type de carte ou sa taille.

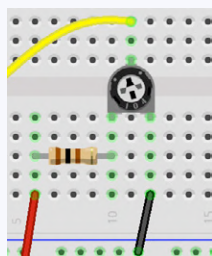


Cartes d'expérimentation cuivrées, l'une à pastilles, l'autre à bandes. Dans le deuxième cas, les pistes peuvent être coupées pour ne garder que les connexions (équipotentiels) souhaitables

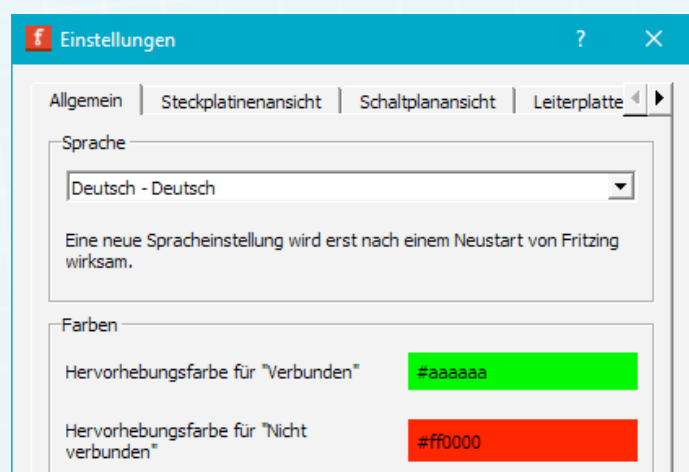
Les cartes d'expérimentation à bande facilitent les manipulations, car il suffit de couper les pistes de cuivre aux endroits appropriés (ou d'établir avec du fil isolé la connexion entre tronçons de piste équipotentiels). Pour couper une piste (ou rétablir le contact), cliquez entre deux trous : agrandissez la vue pour être sûr de cliquer au bon endroit. En pratique, l'interruption des pistes est faite sur la plaque à l'aide d'une lame tranchante, d'une pince coupante de précision ou d'une fraise (bien affûtée) d'un diamètre correspondant à la taille de la piste à couper.

Paramètres pratiques

Pour faciliter la vérification et la rectification des erreurs, les connexions établies par les contacts à ressort de la plaque d'essais sont surlignées de vert. C'est bien, mais dans la documentation cette couleur peut gêner.

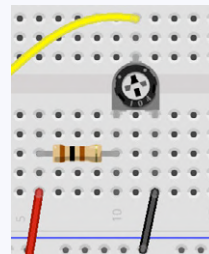


Les connexions établies par les contacts à ressort dans la plaque d'essais sont indiquées en vert.



Paramètres de personnalisation de Fritzing.

La rubrique *Edit/Settings* de *Fritzing* permet de modifier certains paramètres du logiciel. Dans l'onglet *General*, vous pouvez ajuster les couleurs : cliquez sur la zone verte représentant la couleur de surbrillance des connexions (*Connected highlight colour*) et choisissez la couleur souhaitée, un gris moyen moins aveuglant.



Le gris comme couleur de mise en évidence des connexions est beaucoup plus discret que le vert.

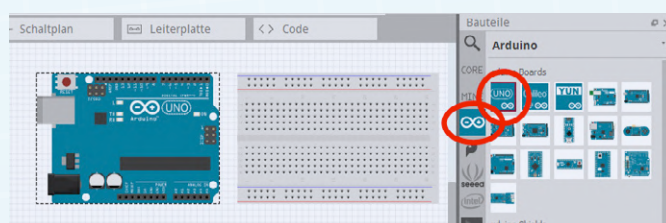
La représentation normale des connexions câblées est rectiligne. Pour de nombreuses connexions, des courbes sont préférables et plus conformes à la réalité. Dans l'onglet *Breadboard view*, vous pouvez activer l'utilisation de fils et de connexions courbes.

Exemple

Exerçons-nous à reconstruire l'image présentée au début de cet article (peu importe la fonction exacte du circuit pour l'instant).

1. Sélectionnez la plaque, puis ajustez son type dans l'inspecteur pour qu'il soit le même que celui de l'illustration que nous allons copier.

2. Sous *Parts* (composants), sélectionnez la collection portant le logo Arduino. Lorsque tous les composants sont chargés, faites glisser le premier symbole (*Uno*) sur le plan de travail de sorte : une carte Arduino Uno se trouve maintenant à gauche de la plaque d'essais.

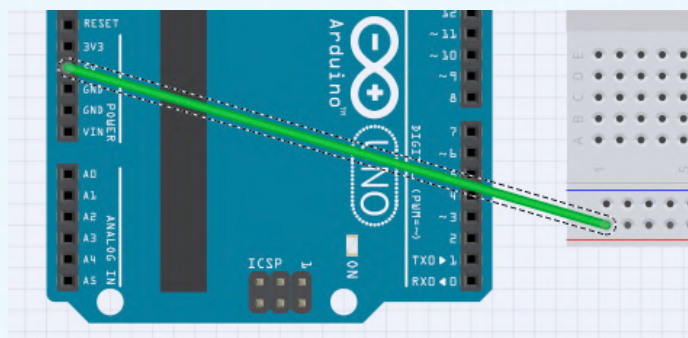


3. Sélectionnez la carte Arduino pour la faire glisser puis, à l'aide de l'outil au bas de la fenêtre, la faire pivoter, jusqu'à ce que la prise USB soit en haut.





4. Vous pouvez ajouter un fil de connexion en cliquant sur un point de contact. Il peut s'agir de l'extrémité de la broche d'un composant, d'un point de contact sur la platine, ou d'une broche sur une carte comme l'Arduino Uno. Établissez une liaison entre la borne 5 V de l'Arduino et le bus d'alimentation rouge sur la plaque d'essais.

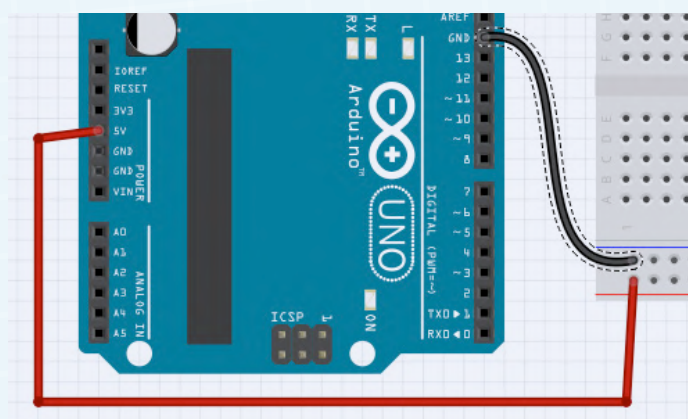


5. Vous pouvez choisir la couleur de la ligne, habituellement c'est le bleu. Cliquez sur la ligne pour en changer la couleur dans l'inspecteur.

6. Si vous souhaitez couder la ligne, double-cliquez sur la ligne à l'endroit où vous voulez qu'elle change de direction. Le coude pourra être déplacé. Vous pouvez ajouter autant de coudes que vous voulez. Un clic droit sur un coude fera apparaître un menu contextuel pour en modifier les propriétés.

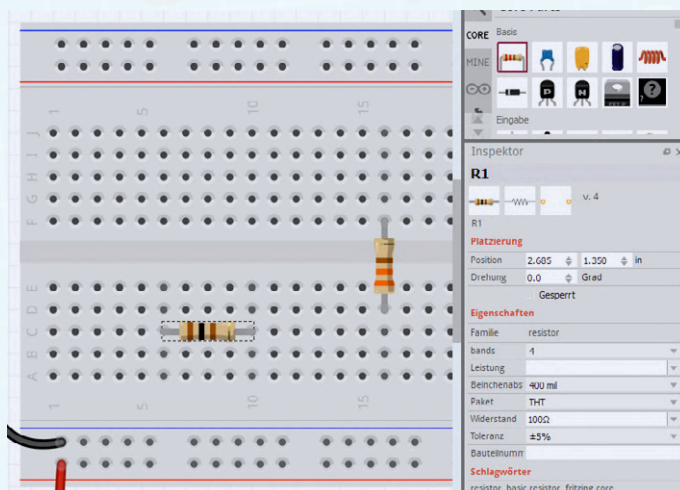
7. Établissez une connexion rectiligne entre la borne GND de droite de l'Arduino et le bus d'alimentation bleu sur la plaque d'essais, et remplacez sa couleur par du noir.

8. Si vous souhaitez transformer une connexion rectiligne en connexion courbe, saisissez la ligne à un point quelconque et faites-la glisser dans une direction quelconque. Vous obtiendrez une courbe de Bézier. Ceci vous permet d'appliquer jusqu'à deux points de courbure pour modifier le trajet de la connexion.



9. La plupart des différents types de composants peuvent être placés de la même manière. Pour chaque type de composant, il existe en général différents réglages dans l'inspecteur, qui affectent par exemple sa couleur, sa valeur ou sa taille. Les composants les plus fréquemment utilisés se trouvent dans la rubrique *Core* ; dans les autres, il y a des modules proposés par les fabricants d'accessoires Arduino. Si vous ne trouvez pas le composant dont vous avez besoin, cliquez sur la loupe pour le chercher en utilisant des mots-clés en anglais.

Ajoutez une résistance sur le plan de travail et faites-la glisser jusqu'à la position souhaitée. Dans l'inspecteur, vous pouvez choisir couche de carbone à trois bandes et couche métallique à quatre bandes. Vous pouvez également choisir une valeur (ou saisir une valeur personnalisée qui n'apparaît pas dans la liste) et une tolérance. Les couleurs des anneaux de la résistance changeront en fonction de ces paramètres.



10. Ajoutez ensuite le potentiomètre et la LED. Lorsque vous placez la LED, la broche la plus longue (qui est l'anode, borne positive, coudée dans la représentation qu'en donne *Fritzing*) doit être à droite. Pour mettre un composant en miroir, utilisez le bouton de rotation au bas de la fenêtre. La longueur d'onde choisie pour la LED a pour effet d'en changer la couleur dans le dessin.

11. Il ne reste plus qu'à ajouter les dernières connexions sur la plaque d'essais et sur l'Arduino.

Cet article est paru dans une édition spéciale d'Elektor (en allemand) offrant une introduction à l'électronique à base d'Arduino. Cette publication sera bientôt disponible en anglais. ◀

200225-B-03

conception de filtres analogiques (2)

Filtres actifs

Alfred Rosenkränzer

Cette deuxième partie de la série d'articles sur la conception de filtres analogiques traite des variantes actives. Les bobines utilisées dans les filtres passifs sont remplacées ici par des composants actifs, généralement des amplificateurs opérationnels. Cela présente des avantages comme on le verra.

Chez beaucoup d'électroniciens, les bobines ne sont pas des composants très appréciés. Pas étonnant, les propriétés des bobines sont loin d'être idéales : constituées d'un fil de grande longueur, elles présentent une résistance ohmique. Celle-ci n'est pas négligeable et dépend de la section et de la longueur du fil utilisé. Sans oublier les inconvénients des capacités de couplage internes qui peuvent conduire à des résonances indésirables. En outre, les basses fréquences nécessitent des inductances fortes et donc des bobines volumineuses, coûteuses et sensibles aux champs magnétiques externes. Dans certaines circonstances, elles produisent elles-mêmes des champs parasites indésirables. Les grosses bobines sont courantes,

en particulier dans les filtres des enceintes passives (fig. 1).

Actifs

Dans les filtres actifs, les bobines sont rares, et le cas échéant elles sont petites. Ce sont évidemment des composants actifs qui rendent un filtre actif : semi-conducteurs sous forme de transistors (rares) ou d'amplificateurs opérationnels (fréquents). Du fait de l'absence de bobines et grâce au recours à des filtres à plusieurs étages, qui ne s'influencent pas mutuellement par la charge, le comportement des filtres actifs se rapproche de l'idéal visé. Dans le cas le plus simple, un transistor (monté en circuit collecteur) suffit comme tampon en tant qu'élément actif. Avec les amplis op cepen-

dant, non seulement la conception est plus simple, mais des amplifications > 1 peuvent facilement être obtenues à certaines fins.

Type de filtre

Comme pour les variantes passives, les filtres actifs peuvent être passe-bas, passe-haut, passe-tout et passe-bande ou encore à réjection de bande. Ces types ont été décrits [1], nous n'y revenons pas ici.

Approximations

Comme les filtres passifs, les filtres actifs peuvent être conçus avec les caractéristiques de Bessel, Butterworth, Tchebychev (et autres). La réalisation de filtres de Cauer et de filtres inverses de Tchebychev avec leurs encoches dans la zone de filtrage

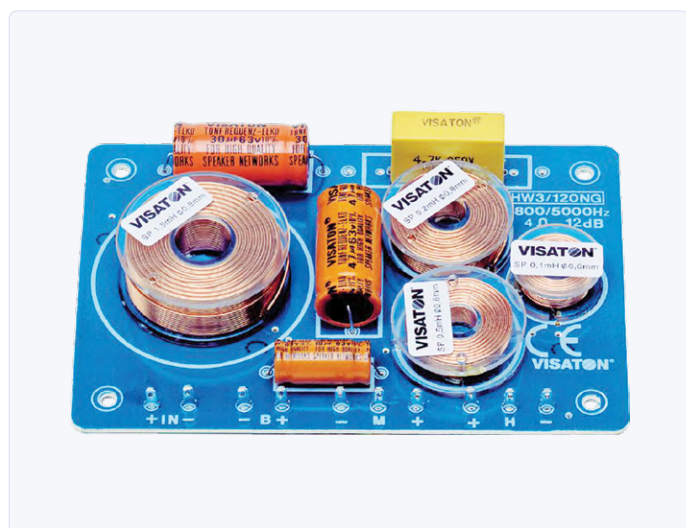


Figure 1. Les filtres de haut-parleurs sont des filtres passifs à bobines volumineuses. Voici le modèle HW3/120NG de Visaton. Photo : Visaton.

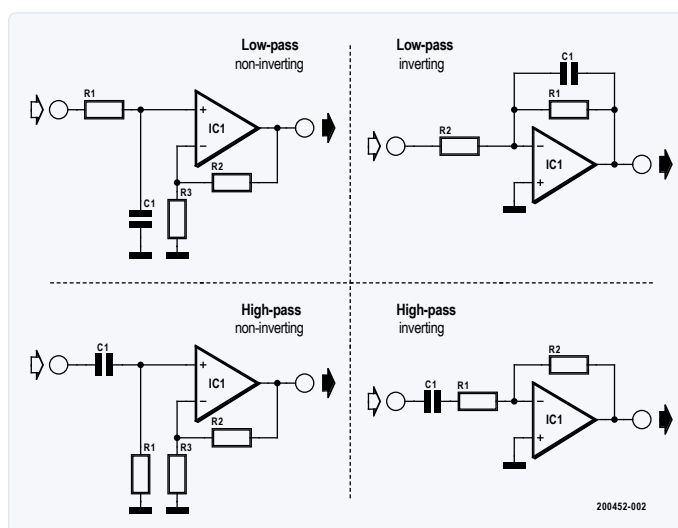


Figure 2. Principaux circuits de passe-haut et passe-bas du 1^{er} ordre en version inverseuse et non-inverseuse.

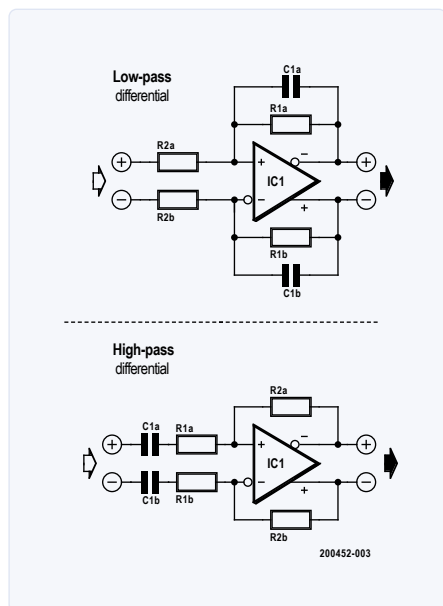


Figure 3. Principaux circuits de passe-haut et passe-bas différentiels du 1^{er} ordre.

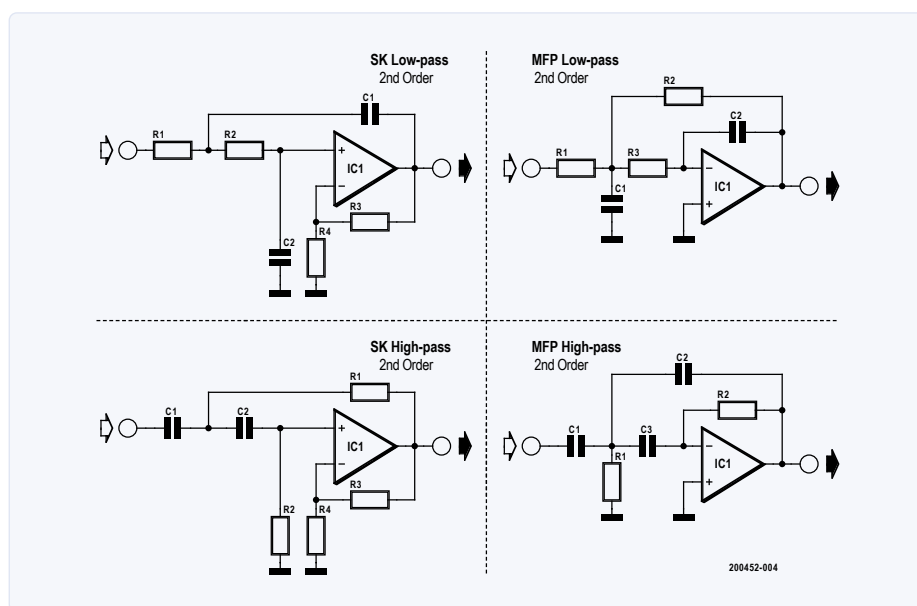


Figure 4. Principaux circuits de passe-haut et de passe-bas du 2^e ordre de type Sallen Key (SK) et Multiple Feedback (MFB).

n'est pas impossible en principe, mais en pratique l'effort requis est trop grand pour un bénéfice limité.

Au lieu de cela, par exemple, un filtre connu sous le nom de son inventeur Siegfried Linkwitz est souvent utilisé pour les filtres séparateurs dans le registre audio. Il permet d'obtenir une réponse en fréquence globale rectiligne des branches passe-bas, passe-moyen et passe-haut d'un filtre séparateur. Un Linkwitz du 4^e ordre consiste en deux filtres Butterworth du 2^e ordre en série – ce n'est pas la même chose qu'un Butterworth du 4^e ordre ! Sur un filtre de Linkwitz, la fréquence de coupure est définie par le point -6 dB, et non -3 dB comme d'habitude. Une réponse en fréquence plate peut être obtenue, mais le retard de groupe est tout sauf plat. Il en résulte de fortes distorsions d'impulsion, notamment avec les signaux carrés.

Structure et ordre

La figure 2 montre les principaux circuits des filtres passe-bas et passe-haut de 1^{er} ordre avec amplis op. Les deux filtres de gauche sont constitués d'un filtre passe-bas et d'un filtre passe-haut RC, suivis d'un ampli op comme tampon non-inverseur. Les deux résistances R2 et R3 définissent le gain de manière traditionnelle. La fonction de filtrage n'est pas influencée par le gain. Dans les deux cas, l'impédance d'entrée est formée par la combinaison R et C.

L'entrée positive d'un ampli op présente une impédance très élevée. Elle peut donc être négligée. La sortie de l'ampli op est également la sortie du filtre et son impédance est donc très faible. Un éventuel deuxième étage de filtrage peut donc être connecté directement à cette sortie au prix d'une faible rétroaction.

Les deux filtres de droite sont basés sur un amplificateur inverseur. Ici, le gain et la fréquence de coupure ne peuvent pas être réglés indépendamment. La fréquence de coupure dépend de R1 et le gain du rapport des deux résistances. L'outil de filtrage que j'utilise ne permet pas de calculer l'amplificateur inverseur. Cela serait redondant, car les voies passe-haut et passe-bas se comportent de la même manière et la fréquence de coupure est toujours donnée par $f_c = 1 / (2 * \pi * R1 * C1)$.

Nous avons déjà mentionné qu'il n'est pas facile de construire un filtre passe-bande ou un coupe-bande de premier ordre. En revanche, une sorte de filtre passe-bande de premier ordre pourrait être réalisée par une connexion en série passe-bas et passe-haut (ou vice versa). Cependant, on ne peut filtrer des bandes étroites avec des filtres qui ne sont pas très raides et dont les fréquences de coupure sont très rapprochées. L'avantage de ces méthodes est le réglage séparé des fréquences de coupure inférieure et supérieure. Attention : si $f_{cPH} > f_{cPB}$, on n'obtient malheureu-

sement pas de réjection de bande, mais un atténuateur à large bande avec une réponse en fréquence pentue.

Filtres différentiels

Les amplificateurs entièrement différentiels (Fully Differential Amps) sont de plus en plus utilisés pour l'audio, mais aussi à des fréquences plus élevées, de plusieurs centaines de MHz. Ils peuvent également faire des filtres actifs (fig. 3). Cependant, cela ne fonctionne que dans une seule topologie. Si cela vous intéresse, pour IC1, on pourrait par exemple utiliser l'amplificateur différentiel LME48724.

Il existe deux structures différentes pour les filtres actifs du second ordre : le filtre Sallen-Key (SK), du nom de ses inventeurs, et un arrangement connu sous le nom de multiple feedback (MFB) à rétroaction multiple. La fig. 4 montre les deux structures pour un filtre passe-bas et un filtre passe-haut. Contrairement au filtre SK, un filtre MFB inverse le signal. Le gain dans la bande passante est déterminé par le rapport entre R3 et R4 dans les filtres passe-haut et passe-bas SK. Avec le passe-bas MFB, le gain de la bande passante est déterminé par le rapport entre R1 et R2 ; avec le passe-haut MFB, cependant, le gain de la bande passante dépend du rapport entre C1 et C2. Les filtres différentiels actifs du 2^e ordre ne peuvent être mis en œuvre que sous forme de structure MFB (fig. 5). Les principaux

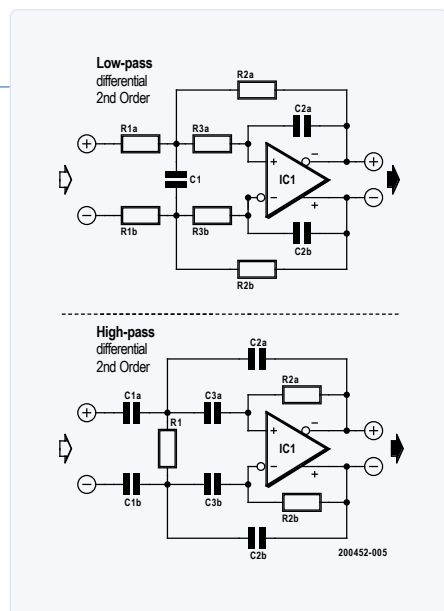


Figure 5. Principaux circuits de passe-haut et passe-bas différentiels du 2^e ordre. Elles ne sont possibles que pour le type MFB.

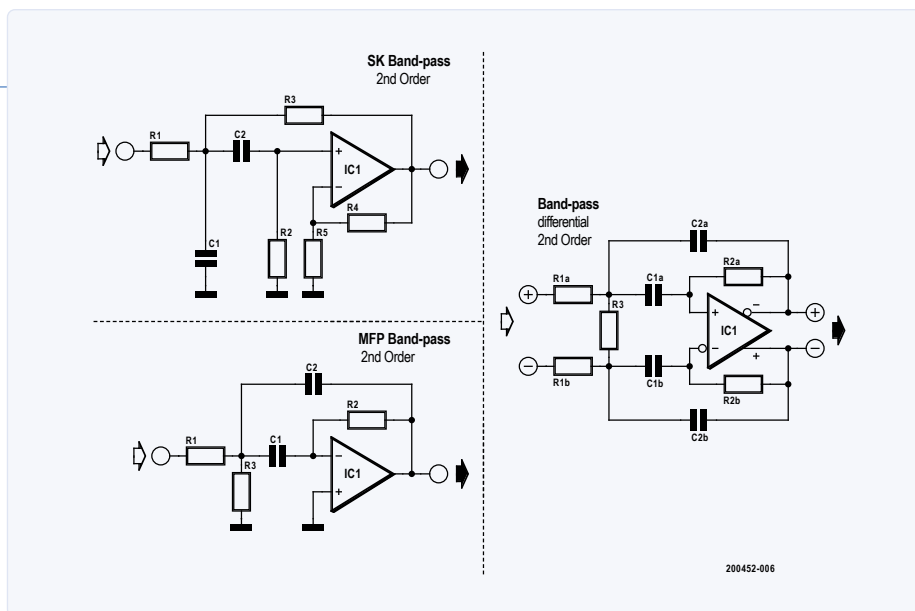


Figure 6. Principaux circuits passe-bande du 2^e ordre.

circuits des filtres passe-bande sont sur la **figure 6**. La bande passante différentielle est de type MFB.

Passe-bande d'ordre supérieur

Les passe-bande à partir du 4^e ordre peuvent être conçus directement comme circuit unique ou comme une combinaison de passe-bas et de passe-haut. Ce dernier présente à nouveau l'avantage que les deux branches du filtre peuvent être réglées séparément.

Avec de tels arrangements en série, il est

même possible de combiner des passe-haut et bas de différents ordres. Cela convient bien aux filtres d'un détecteur de chauves-souris, par exemple : les fréquences „basses“ relativement fortes < 20 kHz peuvent être supprimées avec un passe-haut de 4^e ordre. Un passe-bas de 2^e ordre est suffisant pour bloquer les fréquences > 80 kHz.

Réjection de bande

Pour un filtre coupe-bande de 2^e ordre, il existe, outre SK et MFB (**fig. 7**), la struc-

ture dite *bainter*. La mise en œuvre de SK s'en sort également avec un seul ampli op et s'adapte à un simple dispositif en double T composé de trois résistances et trois condensateurs. Malheureusement, ce concept est sensible aux tolérances des composants. C'est pourquoi les résistances et surtout les condensateurs doivent provenir d'un même lot. Même le vieillissement peut modifier considérablement la réponse en fréquence.

La réponse en fréquence d'un coupe-bande SK pour 8 kHz est illustrée par la **figure 8**.

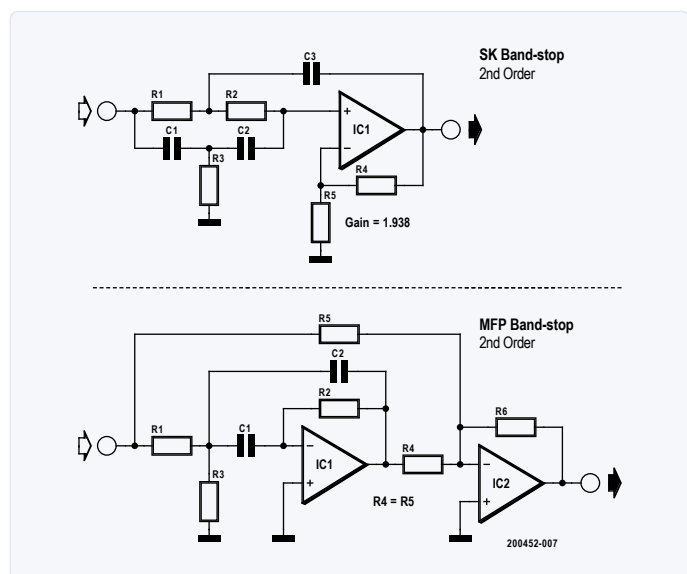


Figure 7. Principaux circuits des réjecteurs de bande du 2^e ordre.

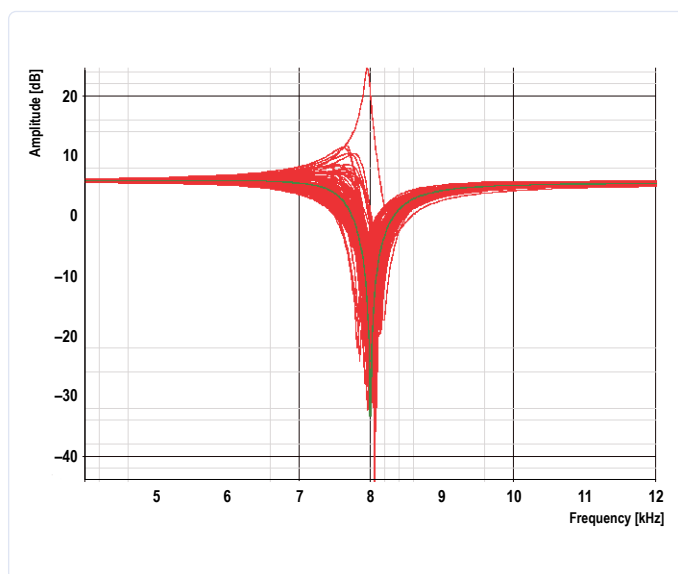


Figure 8. Réponse en fréquence simulée du coupe-bande SK de 2^e ordre. La courbe verte correspond à la réponse idéale. L'essaim de courbes rouges est le résultat de la simulation de Monte Carlo dans les limites des tolérances données des composants.

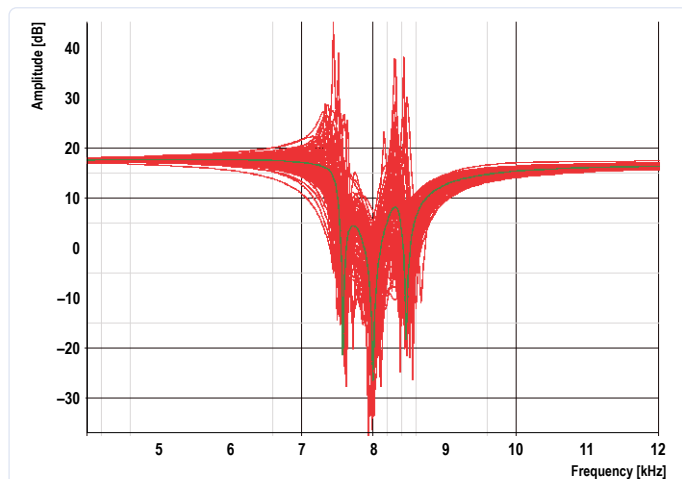


Figure 9. Réponse en fréquence simulée d'un coupe-bande SK du 6^e ordre. Légende : vert = réponse idéale ; rouge = simulation Monte Carlo des tolérances des composants.

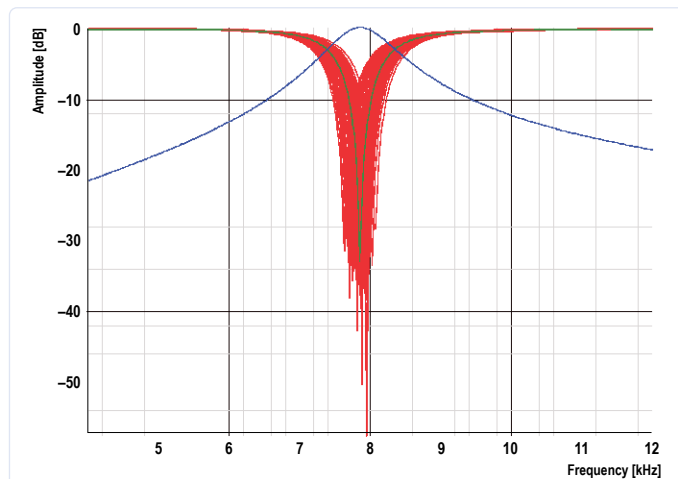


Figure 10. Réponse en fréquence simulée d'un coupe-bande MFB de 2^e ordre. Légende : vert = réponse idéale ; rouge = simulation de Monte Carlo des tolérances des composants ; bleu = courbe passe-bande (voir texte).

La courbe idéale et hypothétique est en vert. Les écarts aléatoires des tolérances des composants de 1 % pour les résistances et de 5 % pour les condensateurs sont particulièrement intéressants. Comme le montre de façon impressionnante la réponse en fréquence de la **figure 9**, les bandes SK d'ordre supérieur y sont encore plus sensibles. En outre, il y a plusieurs vallées dans la bande de réjection. Le gain n'est pas sélectionnable librement. Il est d'un facteur de 2 environ par étage. Dans l'ensemble, on peut dire qu'il vaut mieux choisir un autre type de coupe-bande que la mise en œuvre de Sallen-Key.

Le coupe-bande MFB est en fait un passe-bande dont le signal de sortie est soustrait du signal d'entrée, créant ainsi un coupe-bande de manière quasi indirecte. Cette variante est moins sensible aux tolérances de la valeur des composants, comme le montre la réponse en fréquence d'un coupe-bande MFB de 2^e ordre ainsi construit (**fig. 10**). Ici, la courbe passe-bande idéale est en bleu.

Un filtre Bainter a besoin de trois amplis op à la fois. Vous trouverez de plus amples informations sur cette variante [2][3]. Le circuit de la **figure 11** montre les valeurs concrètes d'une réjection de bande à un peu moins de 8 kHz. La réponse en fréquence correspondante peut être admirée sur la **figure 12**. Les courbes sont plus belles et l'atténuation n'est pas affectée par la dispersion des valeurs des composants.

Il est intéressant de noter que l'ancienne version pour ordinateur de bureau de mon programme de filtrage ne calcule que l'implémentation SK et MFB. Texas Instruments préfère la nouvelle version en ligne [3], qui ne prend en charge que la version *Bainter*.

Passe-tout

Les filtres examinés jusqu'à présent cherchent une variation d'amplitude en fonction de la fréquence. La dépendance en fréquence de la réponse de phase ou du retard de groupe est plutôt secondaire, voire indésirable (sauf pour le filtre de Bessel).

Dans les filtres passe-tout, l'amplitude du signal ne dépend pas de la fréquence, mais elle est constante. La définition d'une fréquence de coupure liée à l'amplitude n'aurait pas de sens. Au lieu de cela, la caractérisation du filtre est le point auquel le retard de groupe diminue de -3 dB (à 70,7 %).

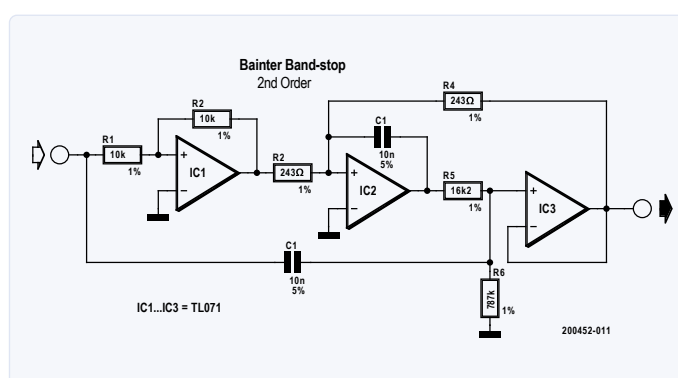


Figure 11. Circuit principal d'un verrouillage de bande de Bainter du 2^e ordre. Trois amplis op sont nécessaires.

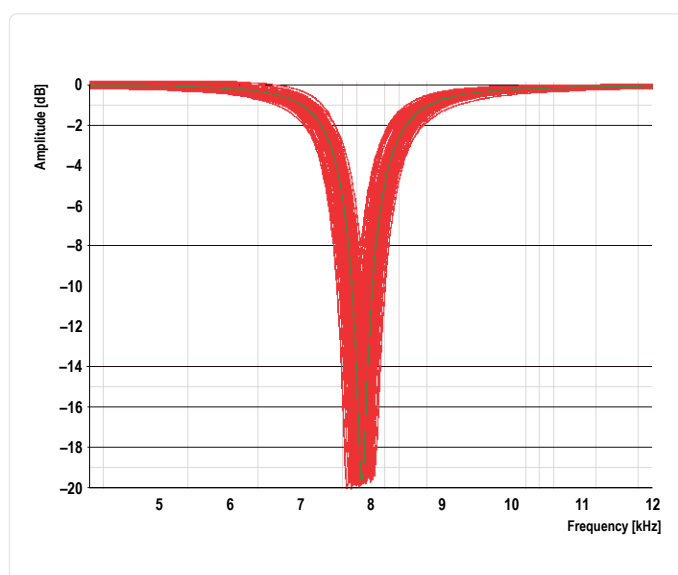


Figure 12. Réponse en fréquence simulée d'un coupe-bande Bainter du 2^e ordre. Légende : vert = réponse idéale ; rouge = simulation Monte Carlo des tolérances des composants.

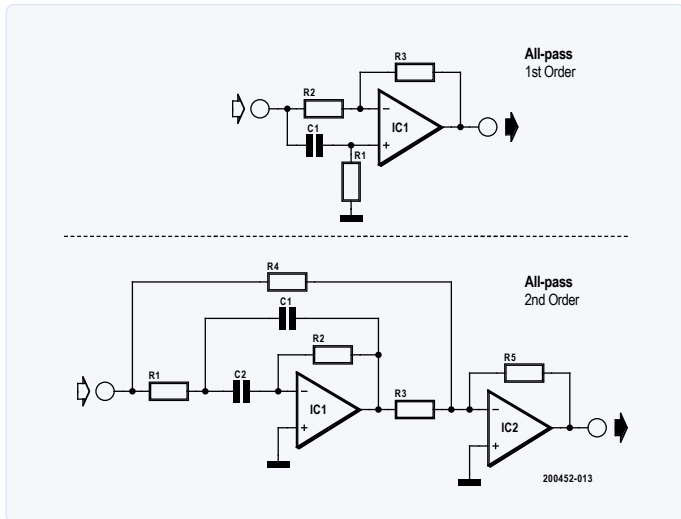


Figure 13. Principaux circuits des passe-tout de 1^{er} et 2^e ordre

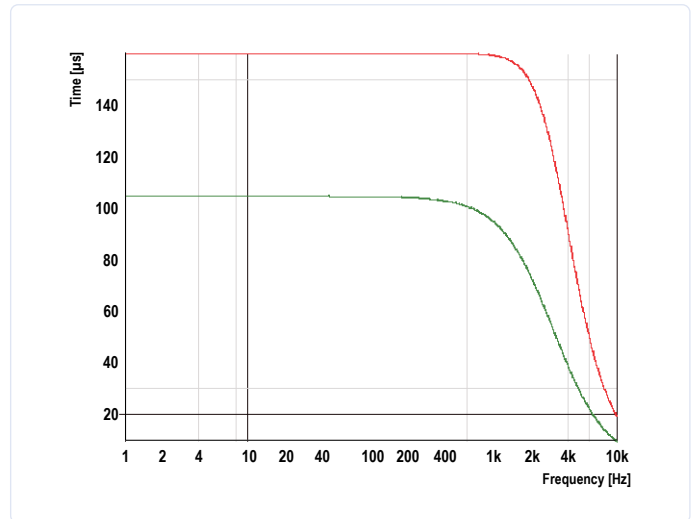


Figure 14. Réponses en fréquence du retard de groupe des passe-tout de 1^{er} (vert) et 2^e ordre (rouge) selon la fig. 13.

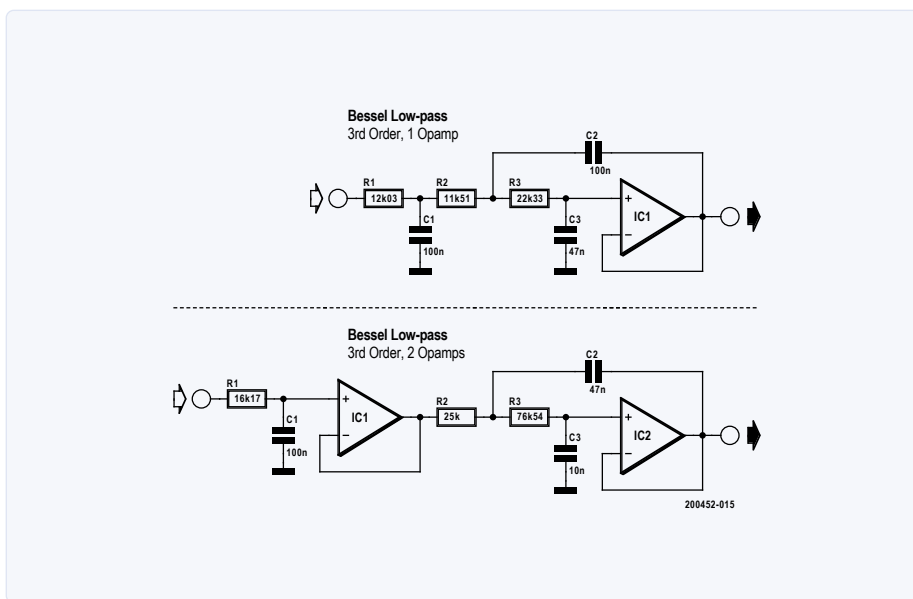


Figure 15. Circuits dimensionnés des filtres passe-bas de Bessel à un et deux amplificateurs opérationnels.

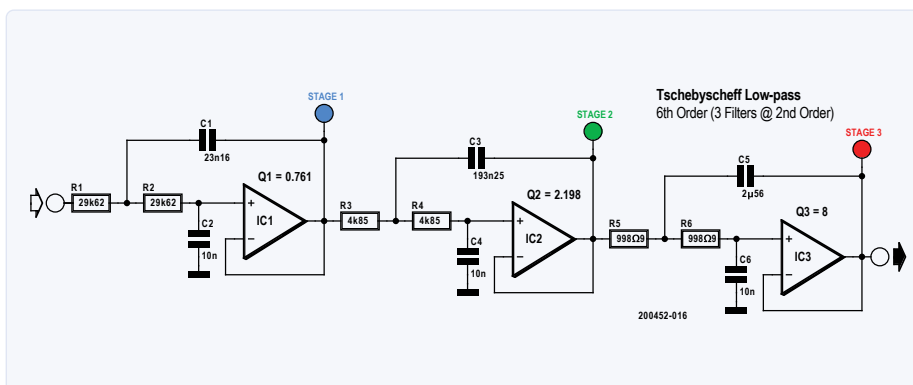


Figure 16. Schéma dimensionné d'un passe-bas Tchebyscheff du 6^e ordre.

La **figure 13** montre les principaux circuits passe-tout du 1^{er} et du 2^e ordre. La **figure 14** montre les retards de groupes correspondants selon les fréquences.

Ordres supérieurs

Les filtres d'ordre supérieur peuvent généralement aussi être mis en œuvre simplement en connectant deux ou plusieurs filtres de premier et de second ordre en série. Cependant, un filtre Tchebyschev de 4^e ordre n'est pas fait de deux filtres Tchebyschev identiques du 2^e ordre en série. Le dimensionnement des composants est différent et, logiquement, le comportement du filtre aussi.

Pour chaque étape de filtrage de premier ou de second ordre, vous avez généralement besoin d'un ampli op — c'est du moins ce que suggèrent les outils de calcul de filtre courants. Néanmoins, des filtres de troisième et même de quatrième ordre peuvent être réalisés avec un seul ampli op (avec des limitations). Au moins, la nouvelle version web de l'outil *TI Filter* n'offre plus ces possibilités, car les amplis op sont bon marché et les efforts les plus importants sont récompensés par des filtres moins sensibles. La **figure 15** montre un filtre passe-bas du 3^e ordre avec un et deux amplis op pour comparaison. Les valeurs des composants des deux filtres ont été calculées par l'outil de filtrage pour une fréquence de coupure de 133 Hz.

Rien ne s'oppose à rendre hétérogène un filtre d'ordre supérieur en combinant différents filtres tels qu'un filtre passe-bas

SK et un second filtre passe-bas MFB. Les filtres partiels peuvent également être mis en œuvre de manière inversée ou non inversée, comme c'est le cas des filtres SK et MFB. Les seules limites sont celles de votre imagination. Pour de tels arrangements plutôt exotiques, avant de le construire, la simulation du circuit est recommandée [4]. Vous saurez tout de suite si les propriétés de filtrage souhaitées sont peu ou prou à votre portée.

Pour les filtres d'ordre supérieur, il y a un autre aspect pertinent : Q , le facteur de qualité du filtre. Vous pouvez demander à l'outil de filtrage de le produire séparément pour chaque étage. En principe, vous pouvez choisir librement l'ordre des étages de filtrage, puisque les sorties des filtres sont toutes à faible impédance. Si les étages avec un Q élevé viennent d'abord, les surcharges sont plus susceptibles de se produire que si l'ordre est inversé. L'outil que j'utilise [5] trie les étages par ordre croissant de Q .

La **figure 16** montre un exemple de filtre à plusieurs étages sous la forme d'un passe-bas Tchebychev d'ordre 6 pour une fréquence de coupure de 1 kHz. La **figure 17** montre la réponse en fréquence après chacun des trois stades dans des couleurs différentes. Si les trois étages du filtre sont simulés individuellement, on obtient les réponses en fréquence individuelles hypothétiques de la **figure 18**. Permutons le premier et le troisième étage du circuit de la figure 16, et nous obtiendrons les réponses en fréquence de la **figure 19**. Notez que les couleurs de chaque étage se déplacent avec lui. Le premier étage reste rouge et le signal global à la fin reste bleu. La courbe rouge de la fig. 17 et la courbe bleue de la fig. 19 sont donc presque identiques. Sur la fig. 19 on voit bien qu'à des niveaux élevés autour de la fréquence de coupure, une surmodulation peut facilement survenir dans la première (rouge) et la deuxième phase (verte). C'est pourquoi la disposition de la figure 16 est préférée à l'inverse.

Gain

En plus de la fonction de filtrage (qui est une atténuation), les filtres actifs peuvent introduire un gain >1 . Avec certains filtres, le gain est lié à la valeur de composants déterminant la fréquence de coupure. Par exemple, avec les filtres SK (fig. 4), d'un ordre >1 , vous ne pouvez pas changer R_3 ou R_4 sans influencer également les propriétés du filtre. Avec l'augmentation du gain, la dispersion des valeurs (différence entre la plus petite et la plus grande valeur d'un composant dans la plage de tolérances admises) des résistances et des condensateurs augmente généralement. Ce fait doit être pris en compte lors de la conception, car cette répartition influence à son tour la tolérance requise des composants. Heureusement, la plupart des outils de filtrage répartissent le gain global d'un filtre d'ordre supérieur sur les amplis op disponibles. Une simulation avec les valeurs de tolérance des composants utilisant la méthode aléatoire (mot-clé : Monte Carlo) permet d'évaluer la sensibilité d'un filtre à la propagation des valeurs.

Dimensionnement des composants

La même réponse en fréquence peut être obtenue avec des valeurs différentes des composants. Si vous prenez des valeurs de résistance plus importantes, les valeurs des condensateurs deviennent plus petites et vice versa. Des valeurs de résistance plus élevées augmentent le bruit, mais des valeurs trop faibles peuvent provoquer des distorsions, car les amplis op doivent fournir des courants de plus forte intensité. Pour les filtres audio, des valeurs de 2 à 4 k Ω sont un bon compromis.

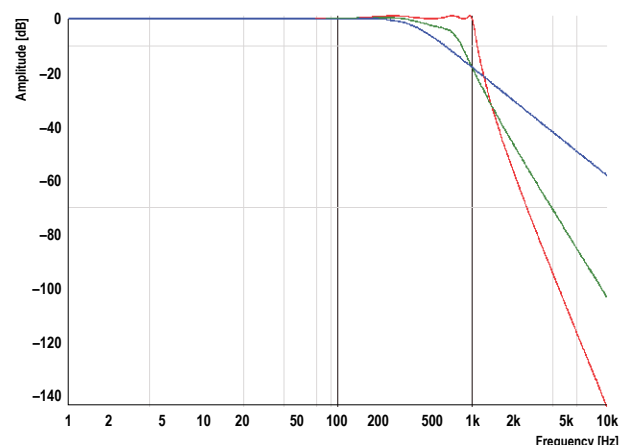


Figure 17. Réponses en fréquence des trois étages du filtre passe-bas de Tchebychev du 6^e ordre de la figure 16. Légende : bleu = étage 1 ; vert = étage 2 ; rouge = étage 3 (sortie).

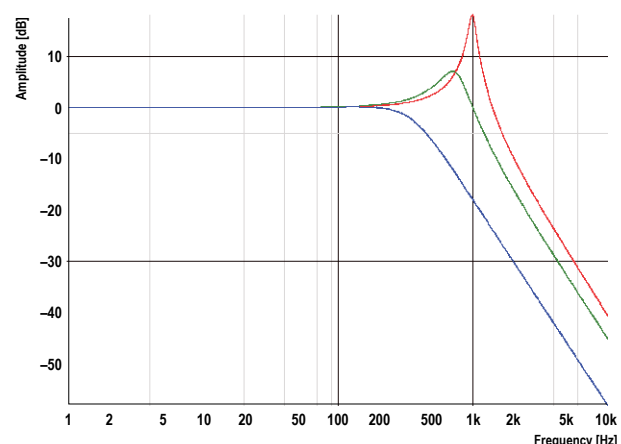


Figure 18. Réponses en fréquence simulées des trois sous-filtres individuels du filtre passe-bas de Tchebychev du 6^e ordre de la figure 16. Légende : bleu = stade 1 ; vert = stade 2 ; rouge = stade 3.

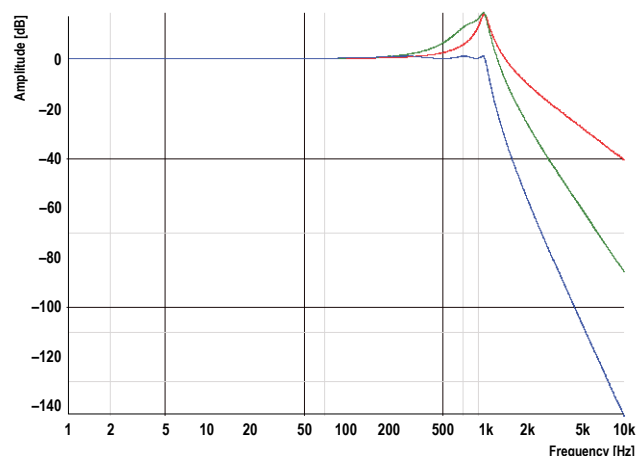


Figure 19. Réponses en fréquence des trois étages du filtre passe-bas de Tchebychev du 6^e ordre avec inversion des 1^{er} et 3^e étages par rapport à la figure 16. Légende : rouge = étage 1 ; vert = étage 2 ; bleu = étage 3 (sortie).

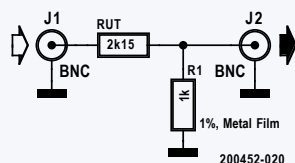


Figure 20. Circuit de test de la distorsion d'une résistance (RUT) avec un analyseur de spectre audio.

Un bon outil de filtrage permet de faire varier les valeurs des composants. Il ne faut pas se contenter de valeurs de composants idéales (fig. 15 et 16), rares notamment pour les condensateurs. Il est agréable de pouvoir spécifier séparément la série E disponible pour les résistances et les condensateurs (fig. 11). Parfois, quand tout cela est vain, vous pouvez essayer de vous rapprocher le plus possible d'une valeur „tordue“ en connectant des valeurs standard en parallèle et/ou en série. L'outil TI montre également l'écart de la courbe réelle du filtre par rapport à l'idéal pour les valeurs des composants sélectionnés. Vous pouvez essayer de réduire les écarts en apportant des modifications.

Une simulation de Monte Carlo prend en compte les tolérances des composants sélectionnés et utilise des ensembles de courbes pour montrer de manière dynamique comment des valeurs aléatoires peuvent altérer la réponse du filtre. Si de grands écarts de la réponse sont détectés, il est recommandé de passer à un filtre moins sensible.

Sélection des composants

Une fois le filtre calculé, il reste à réfléchir au type de composants à utiliser, car il aura un impact sur la qualité et les coûts du filtre. Dans l'exemple d'un détecteur de chauve-souris donné ici, vous n'avez probablement pas besoin de vous soucier de la distorsion des filtres. Si, en revanche, vous souhaitez produire un signal sinusoïdal pur à des fins de mesure audio, il faudra se donner plus de mal.

Résistances :

Lors du tri des composants, il faut tenir compte du fait que telle résistance n'est pas toujours égale à... telle autre résistance de même valeur. Une résistance de bonne facture à film métallique produit nettement moins de distorsion et de bruit qu'un spécimen à couche 0815. La **figure 20** montre un circuit de test qui mesure l'effet d'une certaine résistance (RUT = Resistor Under Test) en la connectant à un analyseur de spectre audio de pointe. R1 doit bien sûr être une

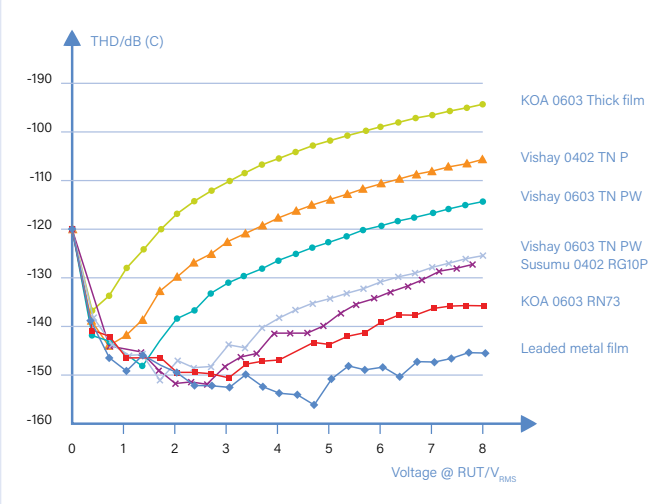


Figure 21. Distorsions des différents types de résistances à 1 kHz en fonction de l'amplitude du signal.

résistance à film métallique avec des données caractéristiques connues et très bonnes.

La **figure 21** montre la distorsion de différents types de résistances en fonction de l'amplitude du signal. Comme vous pouvez le voir, pour la distorsion et le bruit, les résistances SMD à couche épaisse sont certainement un mauvais choix. Pour les résistances CMS à couche mince, des modèles plus grands sont préférables. Les résistances à film de métal au plomb donnent les meilleurs résultats. Devinez quel type de résistance est utilisé dans l'analyseur de précision audio que j'utilise.

Condensateurs :

Des considérations similaires s'appliquent aux condensateurs. Les condensateurs SMD avec ≤ 1 nF sont principalement constitués de matériau NPO et sont donc très bons. À des valeurs plus élevées, d'autres matériaux courants tels que X7R, X5R etc. ne conviennent pas aux filtres à faible distorsion. Des condensateurs allant de 10 nF à ≤ 1 μ F sont disponibles en matériau CoG ; malheureusement, les condensateurs plus grands sont très chers. Les types au plomb comme MKT ou Styroflex restent le premier choix pour les condensateurs. Surtout avec les versions SMD et les valeurs élevées, il faut garder un œil sur la rigidité diélectrique.

Amplificateurs opérationnels :

Le choix du bon amplificateur opérationnel n'est pas aussi facile qu'on pourrait le croire. D'abord il y a les tensions d'alimenta-

LIENS


- [1] Conception de filtre analogique (1), Elektor sept/oct. 2020 : <http://www.elektormagazine.fr/magazine/elektor-156/58994>
- [2] Halbleiter-Schaltungs-Technik 13. Auflage, U. Tietze, Ch. Schenk
- [3] Bandstop Filters and the Bainter Topology, Bonnie C. Baker : <http://www.ti.com/lit/an/slyt613/slyt613.pdf>
- [4] Simetrix : <http://www.simetrix.co.uk/>
- [5] Outil de filtrage de TI, version en ligne : <https://webench.ti.com/filter-design-tool/>

Votre avis, s'il vous plaît...

Vous pouvez contacter l'auteur en allemand ou en anglais par courriel :
alfred_rosenkraenzer@gmx.de

tion. La tendance est à la baisse. Au lieu des ± 15 V habituels, vous verrez de plus en plus de ± 5 V, même moins. Plus les tensions sont basses, plus elles limitent le niveau possible du signal et détériorent le rapport signal/bruit. Les niveaux élevés augmentent le risque de distorsion ou d'écroulement. Ce n'est pas sans raison que ± 15 V reste obligatoire dans l'appareillage audio professionnels.

En général, il convient bien sûr de choisir des circuits intégrés à faible bruit et à faible distorsion. Un autre aspect crucial est le produit gain-bande passante d'un ampli op requis par la conception du filtre. Si vous prenez un ampli op trop lent, vous risquez de ne pas obtenir la courbe de filtrage prévue. Par conséquent, il est préférable de simuler le filtre calculé avant d'en construire un prototype. Et bien sûr, avant

de passer à la production en quantité, non seulement simuler, mais construire et mesurer un prototype. 

200452-03



DANS L'E-CHOPPE D'ELEKTOR

- > **Oscillo numérique OWON SDS1102 à 2 voies (100 MHz)**
www.elektor.fr/18782
- > **Générateur de formes d'ondes arbitraires Siglent SDG2042X (40 MHz)**
www.elektor.fr/17649
- > **Analyseur de spectre OWON XSA1015-TG (9 kHz - 1,5 GHz)**
www.elektor.fr/18978

Une contribution de

Idée, circuits et texte :

Alfred Rosenkränzer

Schémas : **Patrick Wielders**

Rédaction : **Thomas Scherer**

Traduction : **Frédéric Handle**

Maquette : **Giel Dols**

Publicité

Vous souhaitez publier votre montage dans le magazine ?

Rendez-vous sur la page du labo d'Elektor :
www.elektormagazine.fr/labs pour y enregistrer votre projet.

Cliquez sur « Créer un projet ». Connectez-vous (créez un compte gratuit si vous n'en avez pas encore). Remplissez les différents champs du formulaire.

Votre proposition de montage sera examinée par l'ensemble des rédacteurs du magazine. Si votre projet est retenu pour sa publication dans le magazine, un rédacteur prendra contact avec vous pour vous accompagner dans la rédaction de l'article.



Labo d'Elektor :
www.elektormagazine.fr/labs
créer > partager > vendre



dis-moi comment tu ranges ton labo...

Visite guidée dans le saint des saints

... et je te dirai qui tu es !

On peut être passionné d'électronique et avoir une vie de famille, avec les plaisirs et toutes les obligations qui vont de pair... et de grand-père.

Ont collaboré à cet article :

Idee, schéma et photos de **Rainer Schuster** (Allemagne)

Texte : **Eric Bogers**

Maquette : **Giel Dols**

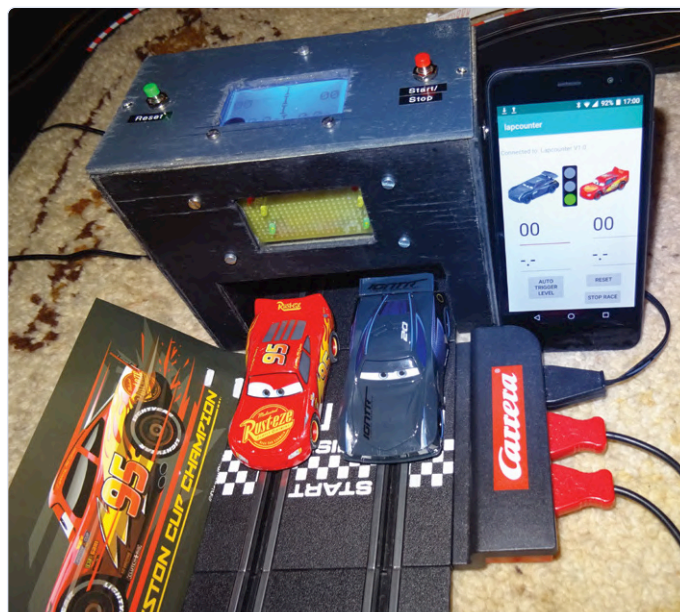


Figure 1. Le compteur de tours s'adapte parfaitement à une piste de course à deux voies.

Pour Noël dernier, Rainer Schuster, qui n'est pas un inconnu de nos lecteurs fidèles, a offert à ses petits-enfants un modèle réduit de piste de course automobile. D'abord ravis, les enfants n'ont pas tardé à se plaindre de l'absence de compteur de tours de piste.

Un grand-père ordinaire se serait rendu au magasin de jouets pour faire l'acquisition d'un tel compteur. Rainer, lui, a pris un papier et un crayon pour dresser la liste des fonctions souhaitées :

- écran graphique pour l'affichage du nombre de tours et du temps ;
- signalisation (rouge/jaune/vert) pour le départ et l'arrêt d'une course ;
- signal sonore (buzzer) ;
- affichage du déroulement d'une course sur un téléphone (Bluetooth).

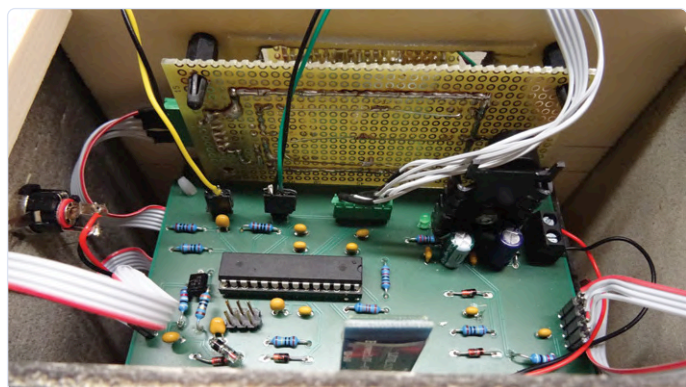


Figure 2. L'électronique est répartie entre deux vrais circuits imprimés et un morceau de circuit à pastilles.



Figure 3. L'afficheur et les deux boutons poussoirs sont montés dans le couvercle du boîtier.

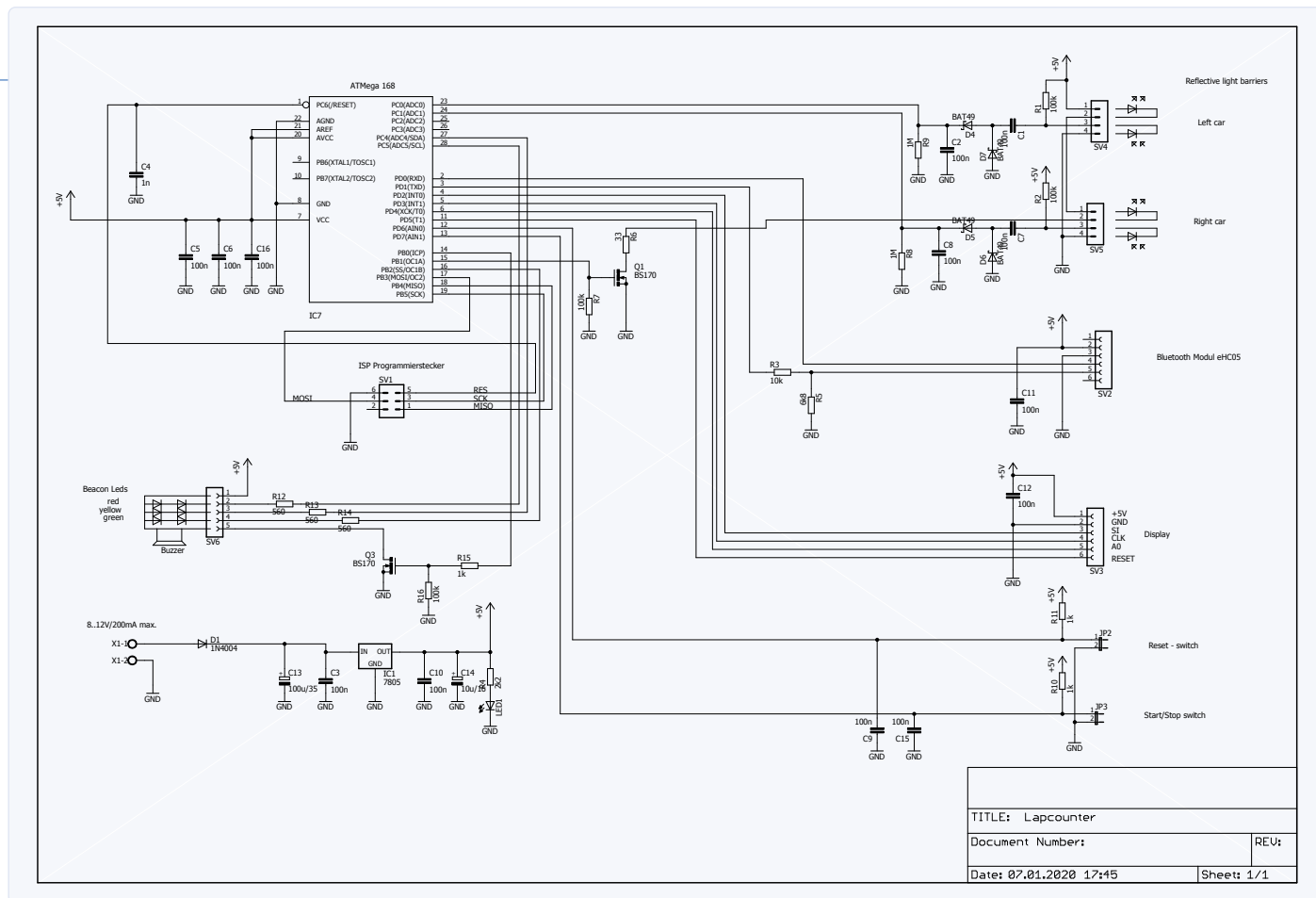


Figure 4. L'électronique est remarquablement simple.

VOTRE LABO, VOTRE ANTRE

Votre labo d'électronique, peu importe qu'il soit grand ou petit, bien organisé ou bien désordonné, ne mérite-t-il pas aussi un coup de projecteur ? Peu importe que vous y conceviez des projets ambitieux ou modestes, envoyez-nous en une courte description avec quelques photos (redaction@elektor.fr). Avec un peu de chance, c'est de vos créations que parlera un prochain épisode de cette série d'articles !

Ce compteur de tours est conçu pour un circuit *Carrera Go*, mais il est facile de l'adapter à d'autres modèles. Le nombre de tours est compté (pour deux voitures) avec des barrières infrarouges réfléchissantes ; le temps de parcours est chronométré pour chaque voiture. Les informations sont affichées sur un écran graphique de 128 x 64 pixels. Les mêmes données peuvent être affichées sur un téléphone (Android). L'alimentation est assurée par un adaptateur secteur (8 à 12 V_{CC}).

Le boîtier du compteur de tours (**fig. 1**) a été conçu pour durer plutôt que pour séduire par son aérodynamisme. Il s'adapte parfaitement à une piste à deux voies ; les barrières à infrarouge sont situées à l'intérieur des montants de part et d'autre.

L'électronique est logée sur deux cartes : l'une pour le μ C, un ATmega168, l'autre pour l'afficheur monté dans le couvercle avec les boutons marche/arrêt et remise à zéro (**fig. 2**, sans afficheur). Les LED de signalisation sont soudées sur un morceau de circuits à pastilles. L'afficheur apparaît sur la **figure 3**.

Les deux émetteurs-récepteurs IR sont du type ITR9904. Pour éliminer les interférences de la lumière ambiante, les LED IR sont commandées par des impulsions de 2 ms, fournies sur sa sortie PWM par le microcontrôleur.

Le schéma (**fig. 4**) montre que le matériel se réduit à peu de chose.

Le plus gros du boulot, c'est le microcontrôleur qui le fait. L'auteur a préparé une vidéo du compteur de tours [1].

Si ce projet vous intéresse, faites-le nous savoir par courriel à l'adresse redaction@elektor.fr ; votre intérêt nous incitera à revenir éventuellement sur le sujet pour l'approfondir. En passant par cette adresse, vous pouvez aussi vous adresser directement à l'auteur.

200045-02

LIEN

- [1] Vidéo de présentation du compteur de tours : <https://bit.ly/2ZxPwh0>



Dans l'e-choppe d'Elektor :

> Joy-Car Educational Robot (incl. BBC micro:bit)
www.elektor.fr/19408

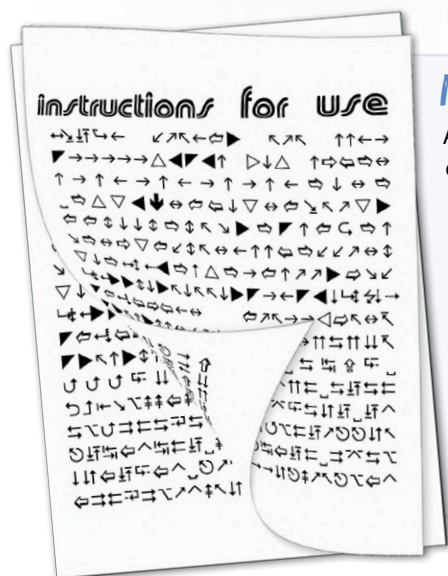
bureau d'études – Zone D

D comme développement, débrouille et dur-à-cuire !
Trucs et astuces, bonnes pratiques et autres informations utiles

Clemens Valens (Elektor Labs)

DE L'IDÉE AU PRODUIT – 7^E PARTIE (SUITE ET FIN)

Nous avons déjà abordé dans cette série le processus de mise sur le marché d'un nouveau produit. Nous avons commencé par préparer un plan d'affaires, puis nous avons conçu le produit et nous sommes allés jusqu'à le produire et le tester. Maintenant, il est enfin prêt à être vendu, n'est-ce pas ? Presque...



MANUELS D'UTILISATION ET DE SERVICE

Aucun produit n'est fini tant qu'il manque la notice. Pour vous, tout est clair et logique, mais ce n'est pas la vision qu'en a l'utilisateur qui le découvre. Vous lui devez des explications. Un manuel d'utilisation demande des compétences, car un bon manuel est concis, clair et complet. Vous sentez-vous à la hauteur de la tâche ou serait-il judicieux d'engager un rédacteur technique pour cela ? Une fois que vous avez le manuel, il faut faire le nécessaire pour qu'on le trouve facilement, en le joignant au produit et/ou en en faisant une version consultable en ligne ou téléchargeable.

Pour certains produits, il faut aussi un manuel d'entretien, destiné à ceux qui, sur le terrain, assurent l'entretien de vos produits. Il explique p. ex. comment remplacer des pièces et comment ça doit fonctionner ou quel type de signal est censé apparaître sur quel point de test. La production d'un tel manuel est aussi une tâche considérable que vous pouvez sous-traiter.



Dilemme : emballer votre produit dans un carton ordinaire ou faire un effort pour séduire l'utilisateur ?

EMBALLAGE

Les spécialistes du marketing vont confirmer que l'emballage peut faire ou défaire un produit. L'emballage inclut le produit lui-même, son manuel, et souvent quelques accessoires, et il permet d'identifier ce qu'il contient. Un bon emballage permet de stocker et de transporter le produit en toute sécurité. Un bon emballage donne envie d'acheter le produit. Comme le produit, son emballage nécessite un effort de conception. Les dimensions, la forme et le matériau ne doivent pas être le fruit du hasard, pas plus que le graphisme et le texte imprimés. Faudra-t-il un matériau de rembourrage ? Un emballage sur mesure ou un carton ordinaire ? S'il s'agit d'un emballage personnalisé, qui va le fabriquer ? Quel en sera le coût ? Affectera-t-il le prix de détail du produit ? Au fait, qui va emballer votre produit ?



EXEMPLES D'UTILISATION

Si vous produisez des amplificateurs audio ou des voitures, il n'y a pas lieu d'expliquer à vos clients à quoi ils servent. Pour d'autres produits, il faut donner des exemples d'utilisation. Le fait que l'utilisateur ait acheté votre produit n'implique pas qu'il sache comment en exploiter toutes les fonctions intéressantes. Dans votre manuel de l'utilisateur, la ligne «*Bouton TRUC : appliquer un TRUC*» ne rime à rien si vous n'expliquez pas ce qu'est un «TRUC» et pourquoi vous voudriez l'appliquer (et à quoi). Donnez toujours non pas un, mais plusieurs exemples intéressants, qui sortent du cadre de l'utilisation élémentaire.

EXPÉDITION, DISTRIBUTION ET COMMERCIALISATION

Avez-vous réfléchi assez tôt à la distribution de votre produit ? Le vendrez-vous directement ou préférez-vous laisser cela à des spécialistes (distributeurs, revendeurs...) ? Si c'est vous, vous n'aurez pas à payer de commission, mais il faudra peut-être engager quelqu'un pour aller à la poste tous les jours. Ou passer un accord avec un service de coursier ? Les frais d'expédition seront-ils forfaitaires ou fixés selon la destination ? Si vous optez pour un détaillant, comment saurez-vous s'il fait les efforts suffisants ? Qui s'occupe du marketing ? De combien êtes-vous prêt à augmenter le prix au détail pour financer cela ? Ou êtes-vous disposé à réduire votre marge ?



FACTURATION ET COMPTABILITÉ

C'est surprenant, mais de nombreuses jeunes entreprises dépérissent faute de prendre le temps de suivre les questions financières, le suivi des factures impayées ou des demandes de devis ou même la comptabilité en général. Que ce soit par excès de gentillesse envers ses clients ou sous la pression d'une demande écrasante, si l'aspect financier est ignoré trop longtemps, l'entreprise ne survivra pas longtemps. Vous avez laissé impayées des factures de vos fournisseurs en vous disant « ils ont de l'argent ». Vous voici devenu fournisseur à votre tour, et devez traiter maintenant avec des clients mauvais payeurs. Il est ennuyeux d'avoir à chasser les débiteurs, mais si vous n'envoyez pas de factures, vous n'aurez pas de débiteurs à chasser, ce qui est encore bien pire. L'établissement soigneux de devis prend du temps, mais de bons devis attirent de nouveaux clients. Nul n'ignore l'importance d'un suivi rigoureux des dépenses et des recettes pour maintenir l'équilibre. Il faut donc lui accorder un peu de temps. On tergiverse souvent sur ce point, parce qu'il y a, croit-on, tant d'autres choses (plus) importantes à faire pour faire tourner la boîte. Dans un sens, il est efficace de tergiverser longtemps, car ça finit par faire disparaître vos problèmes... et tous vos rêves avec.

AIDE ET SERVICE APRÈS-VENTE

La qualité de votre service après-vente affecte directement la réputation de votre entreprise et se répercute sur les ventes (sauf si vous êtes Apple). L'après-vente peut devenir un processus long, alors demandez-vous qui va communiquer avec vos clients et comment ? De nombreuses entreprises s'efforcent de garder secrets leur(s) numéro(s) de téléphone et leur adresse électronique. Leur idée est de rediriger toutes les demandes de service vers une page web contenant soit un formulaire de contact, soit une liste de questions fréquentes (FAQ) (n'oubliez pas les réponses !), soit un robot *chatteur*, soit un forum ou autre. Même si vous mettez en place le meilleur centre de services d'entraide imaginable, il y aura toujours des clients qui voudront parler à quelqu'un en chair et en os. Et il faudra toujours une vraie personne pour lire les courriels qui arrivent à info@crise.com, traiter les formulaires de contact, s'occuper des rapports produits par le robot *chatteur* et ouvrir le courrier distribué par le facteur ce matin.



RETOURS, RÉPARATIONS ET GARANTIES

Même si vous avez fait de votre mieux pour vous assurer que tous vos produits ont été construits selon les normes de qualité les plus élevées et fonctionnent exactement comme prévu, un pourcentage d'entre eux finira par vasouiller. Il faut espérer que ce seront des exceptions, mais il en reviendra certains à vos bureaux accompagnés d'une plainte – probablement vague. Comment traitez-vous les retours de produits ? Faites-vous des réparations ? Ou est-il moins cher de les remplacer ? Qui s'en chargera ? Si vous remplacez, que ferez-vous des exemplaires défectueux ? Qu'est-ce qui est garanti et qu'est-ce qui ne l'est pas ? Pour quelle durée ? Certains pays fixent des durées minimales pour certains types de produits, vous n'aurez donc pas le choix.

ÉLIMINATION DES DÉCHETS

Dans différents pays, les fabricants sont responsables de l'élimination des produits jetés par leurs utilisateurs. Cela signifie que les utilisateurs peuvent (et vont) renvoyer les produits au fabricant pour destruction ou recyclage. Plus vous vendrez de produits, plus vous recevrez de déchets. Un tas de déchets électroniques peut s'accroître rapidement. Certains de ces articles peuvent-ils être remis à neuf ? Certaines pièces sensibles doivent-elles être éliminées avec un soin ? Comme pour tout, quelle que soit la manière dont vous traiterez les déchets, cela impliquera des coûts.



PENSER À L'AVENIR

Bien sûr, votre produit est génial et il résout tous les problèmes pour lesquels il a été conçu, mais tôt ou tard les clients commenceront à demander d'autres fonctions. Vous-même trouverez peut-être une idée géniale pour faire évoluer votre produit ou pour en élargir la gamme. Malheureusement, vous êtes maintenant pieds et poings liés à votre production, à vos essais, à l'expédition, au conditionnement, au marketing, à la comptabilité, aux ventes et au service après-vente, et vous n'avez donc plus une minute à consacrer à l'ingénierie et à la conception de nouveaux produits. Là aussi la procrastination fera disparaître le problème, et votre entreprise avec.

BUREAU ET EMPLOYÉS

Tant que vous travaillez seul, cela ne vous dérange probablement pas de travailler dans votre garage ou dans la cuisine, mais dès que vous commencerez à embaucher, il vous faudra plus d'espace. Il y a bien le télétravail, popularisé par qui vous savez, mais convient-il lorsque la communication est intense ou lorsque les fumées de soudure filtrent la lumière du jour ? Vous louerez peut-être un bureau. Quelque chose de bon marché au fond d'une zone industrielle ? Ou plutôt dans un incubateur, séduisant par sa situation en ville, offrant des services utiles, avec éventuellement une aide pour les questions financières par exemple ?... Avant de signer quoi que ce soit, assurez-vous de lire attentivement tous les petits caractères ! Les employés, il ne suffit pas de les payer, il faut encore les gérer, ce qui prend du temps. Vous serez tenté d'engager un ami pour cela, mais l'ami sera peut-être encore plus difficile à gérer. Si vous voulez garder vos amis, placez plutôt une annonce sur une plateforme de médias sociaux.



Tous vos employés tiendront-ils dans votre cuisine ?

LA FIN

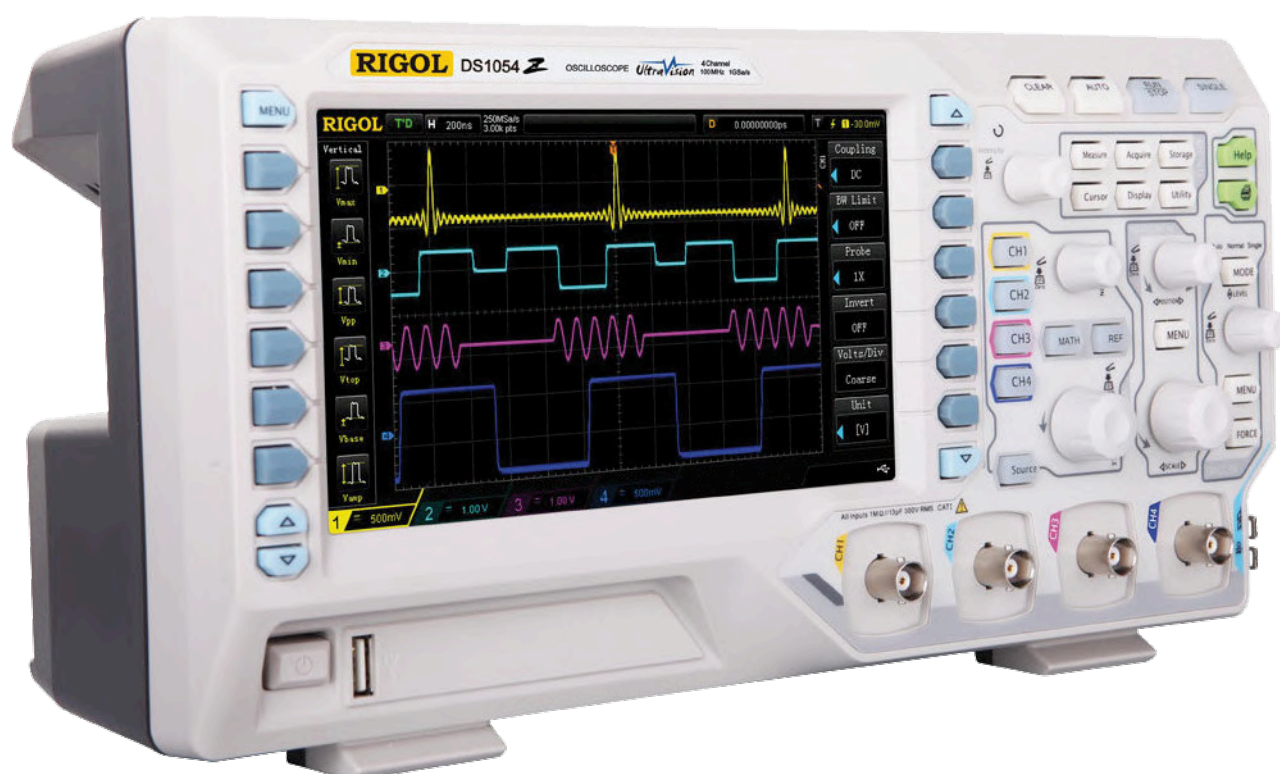
C'est tout pour cette série. Il y aurait encore beaucoup à dire, et je suis sûr d'avoir oublié l'un ou l'autre détail pas négligeable. Vous aurez compris l'idée générale, n'est-ce pas ? Si vous avez suivi les lignes directrices de ces articles et combiné tout cela avec votre propre bon sens, vous êtes peut-être déjà en affaires. Espérons que votre produit aura du succès. Ne vous croyez pas obligé d'avoir tout parfaitement en place dès le premier jour. L'évolution sera organique, mais préparez-vous toujours à ce qui pourrait suivre. Bonne chance !

Votre avis, s'il vous plaît...

Vous pouvez adresser vos questions ou vos commentaires à clemens.valens@elektor.com ou à redaction@elektor.fr

oscilloscope à 4 voies

Rigol DS1054Z



Harry Baggen

Si vous cherchez un oscilloscope à quatre voies fiable mais abordable, vous tomberez tôt ou tard sur le populaire Rigol DS1054Z. Pour moins de 100 € par voie, cet appareil offre en plus de ces 4 voies un grand écran, de nombreuses fonctions de mesure et bien d'autres possibilités qui justifient qu'on s'y intéresse de près pour un banc d'essai.

Le Rigol DS1054Z n'est pas un modèle récent, mais il reste l'un des oscillos à 4 voies les plus populaires dans la gamme de prix d'environ 400 €. Ses spécifications les plus importantes sont bien sûr le nombre de canaux (4 !), le taux d'échantillonnage (1 Géc/s pour un canal) et la largeur de bande (50 MHz). Par rapport aux concurrents, ce sont de très bonnes spécifications pour 400 €. En plus, vous bénéficiez de quatre options logicielles gratuites, à

savoir une extension de mémoire à 24 Mpts, une option de déclenchement avancée, des décodeurs de protocole pour RS232, I²C et SPI, et un module d'enregistrement. Voilà qui rend l'appareil très attrayant pour ce prix.

Matériel

Le DS1054Z tient dans un boîtier assez compact de 31x16x12 cm et pèse un peu plus de 3 kg. La plus grande partie de la

façade est occupée par un grand écran de presque 18 cm (7 pouces) avec une résolution de 800x480 pixels, ce qui semble rester la norme dans cette gamme de prix. La plupart des commandes sont réunies sur la droite. Comme l'espace est limité, il n'y a qu'un jeu de boutons de réglage vertical pour les 4 voies. Quatre boutons poussoirs vous permettent de l'assigner à la voie concernée. À droite se trouvent aussi les réglages horizontaux, le déclen-

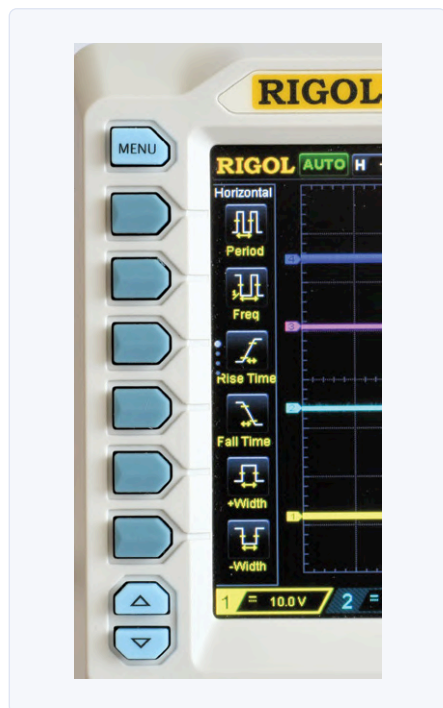


Figure 1. Les boutons de gauche du menu de fonction du DS1054Z.



Figure 3. Les boutons de droite du menu de fonction du DS1054Z.

chement et des touches de menu qui déterminent les fonctions affichées sur le bord droit de l'écran : *Measure* (mesure), *Acquire* (acquisition), *Storage* (mémoire), *Cursor* (curseur), *Display* (affichage) et *Utility*. Un sélecteur rotatif multifonctionnel permet

de parcourir différents menus. Vous aurez remarqué de part et d'autre de l'écran les nombreux boutons de fonction. Ceux de gauche servent au menu de mesure, avec un ensemble d'options de mesure pour le calibre horizontal et un

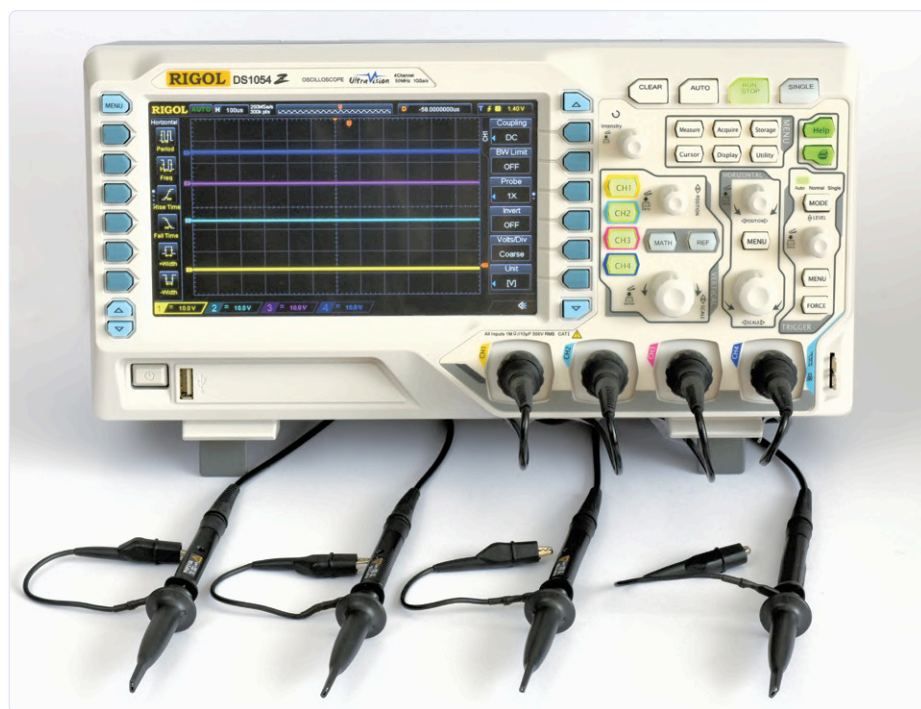


Figure 2. Moins de 100 € par voie pour l'oscilloscope DS1054Z.

autre ensemble pour le calibre vertical. Les valeurs de mesure choisies apparaissent au bas de l'écran. L'ensemble de boutons à droite sert aux menus de fonctions.

Le boîtier robuste est muni à l'avant de deux plots escamotables. À l'arrière, en plus de la fiche pour le cordon du secteur, il y a un connecteur USB et LAN et une sortie de déclenchement pour les mesures en série. Le ventilateur du DS1054Z n'est pas inaudible, mais ce n'est pas dérangeant.

Au travail !

Du fait de la pléthore de boutons, la manipulation des commandes de l'oscilloscope implique quelques tâtonnements au début. En fait, leur organisation est similaire à celle d'autres appareils dans cette gamme de prix. Cela s'applique également à la sélection des voies à l'aide des quatre poussoirs. Au bas de l'écran, le numéro de la voie active est indiqué ; avant de modifier un réglage vertical, on apprend vite à y jeter un coup d'œil pour s'assurer d'intervenir sur la bonne voie !

La combinaison de la largeur de bande disponible et de la fréquence d'échantillonnage maximale est bien choisie. Certains trouveront un peu étroite la bande de 50 MHz, mais cela suffit pour la plupart des applications tant que vous ne travaillez pas vraiment en HF. N'oubliez pas que le taux d'échantillonnage est divisé par deux ou quatre lorsqu'on utilise 2, 3 ou 4 voies simultanément. À 50 MHz, il ne reste alors que 5 échantillons/période, ce qui interdit la restitution du signal. Cette combinaison de largeur de bande et de taux d'échantillonnage est donc vraiment très bien !

L'écran du DS1054Z est facile à lire, même vu de côté. J'ai remarqué qu'en l'utilisant j'avais tendance à augmenter un peu la luminosité. Les informations affichées sont très nombreuses, grâce notamment aux menus commandés par les poussoirs de part et d'autre de l'écran. Du fait de la présence d'informations en haut et en bas de l'écran, l'ensemble paraît assez chargé. L'espace qui reste pour les formes d'onde elles-mêmes est donc restreint.

Pas moins de 15 fonctions de déclenchement différentes sont disponibles sur le DS1054Z : *pattern* (patron), *time-out* (délai d'attente), *window* (fenêtre), *delay* (retard) et aussi les protocoles de communica-

tion sériele pris en charge. La mémoire interne d'échantillons de 24 Mpts est assez grande mais, comme nous l'avons vu pour le convertisseur A/N, cette mémoire est partagée par les voies actives. Même quand elles le sont toutes les quatre, la capacité reste amplement suffisante.

Rigol a intégré de nombreuses fonctions de mesure dans cet appareil. Les boutons à gauche de l'écran permettent de choisir rapidement parmi 32 paramètres, dont les valeurs mesurées (max. 5 à la fois) apparaissent alors en bas de l'écran. Il y a aussi un fréquencemètre „normal“, et vous pouvez choisir d'afficher 20 paramètres simultanément et un maximum de 5 paramètres une fonction statique montre les écarts sur un certain nombre de périodes. Vous pouvez aussi mesurer avec le curseur, automatiquement ou manuellement.

Sous le bouton *Math*, nous trouvons de nombreuses opérations mathématiques, de la simple addition au logarithme, en passant par l'exposant, l'intégration et la différenciation ... et la fonction FFT qui permet d'afficher le spectre harmonique d'un signal. Un petit atout a priori appréciable, mais dont j'ai eu du mal à régler tous les paramètres pour obtenir une image utilisable. Il me semble qu'il reste là pour Rigol une marge d'améliorations possibles de son logiciel.

Le DS1054Z offre pour ce prix des extras considérables, dont je n'ai encore décrit ici que celles que j'ai essayées ou remarquées. Il est possible également de connecter l'oscillo à un ordinateur pour le commander. On trouve sur le site de Rigol les programmes Windows *UltraSigma* et *UltraScope*.

Ma conclusion

Par rapport à son prix, l'oscilloscope polyvalent Rigol DS1054Z a beaucoup à offrir, d'autant plus qu'il est livré désormais avec divers logiciels supplémentaires. Il est sur le marché depuis plusieurs années, mais tient la dragée haute à des concu-

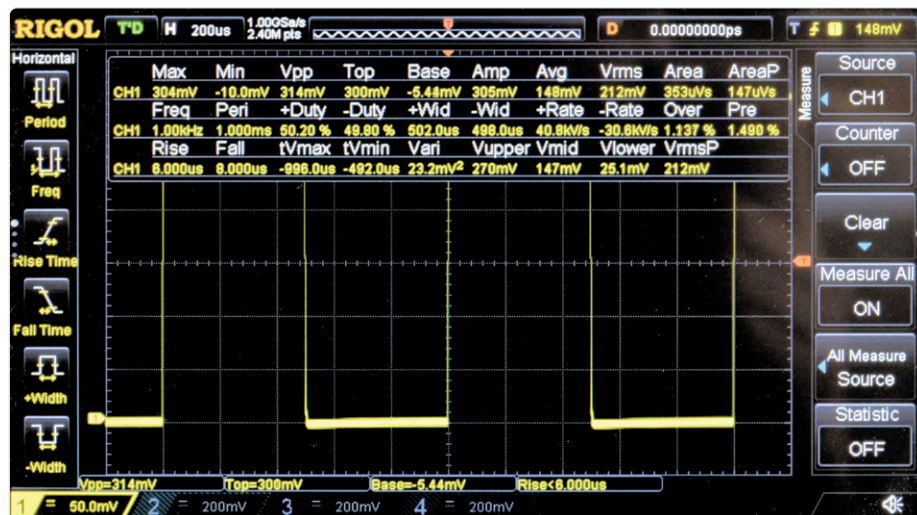


Figure 4. Affichage simultané de toutes les valeurs mesurées par le DS1054Z.

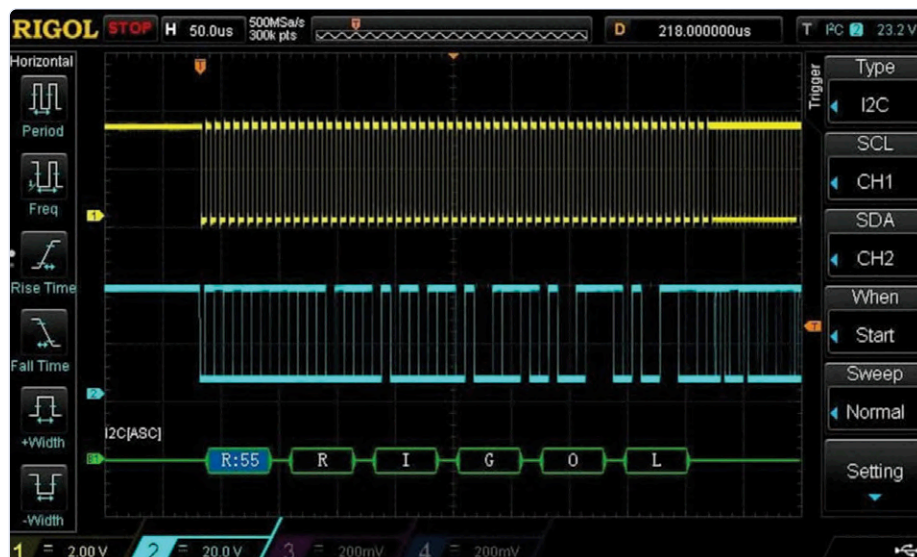


Figure 5. Le décodage du protocole I²C est standard sur le Rigol DS1054Z.

rents plus récents dans cette gamme de prix. Pour 400 €, c'est un appareil fiable à 4 voies, avec de multiples fonctions, qui vous rendra service pendant de nombreuses années. ◀

200463-04



DANS L'E-CHOPPE D'ELEKTOR

➤ **Oscillo à 4 voies Rigol DS1054Z**
www.elektor.fr/17821

bougie de Noël électronique

Soufflez-la comme une vraie !

Chaque année, Elektor fait de son mieux pour proposer un nouveau gadget de Noël. Ce n'est pas une tâche facile. Il faut quelque chose d'agréable, si possible drôle, forcément abordable, évidemment unique, assez facile à réaliser et le petit Jésus tant attendu doit en plus combiner au mieux toutes ces qualités. Or les étals débordent de gadgets à des prix (presque) imbattables ! L'originalité, ou l'unicité, est peut-être le plus grand défi, que nous relevons cette fois en combinant deux anciens projets d'Elektor.

Ont contribué à cet article :

Idée : **Clemens Valens, Luc Lemmens**

Conception et texte : **Luc Lemmens**

Illustrations : **Jan Visser, Luc Lemmens**

Schéma : **Patrick Wielders**

Rédaction : **Jens Nickel, C. J. Abate**

Mise en page : **Giel Dols**

Version française : **Helmut Müller**

INFOS SUR LE PROJET

Mot clés

Noël, vacances, maison & jardin

Niveau

débutant – connaisseur – expert

Temps nécessaire

environ 4 h

Outils

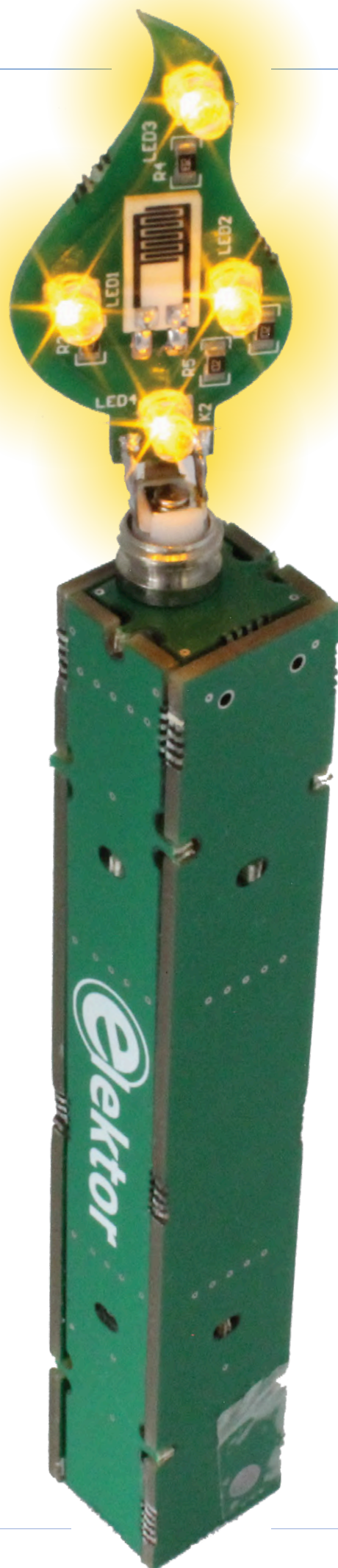
Outils de soudure (pour CMS et traversants)

Coût

~20 €

Le premier projet recyclé pour le gadget de Noël de cette année date de décembre 2011 : la bougie électronique à LED [1]. De cet article, seule l'idée de «souffler une bougie LED électronique» a été retenue pour le nouveau projet ; l'électronique a été revue de fond en comble. En 2011, pour obtenir l'effet de scintillement, c'est un μC PIC16F1827 qui pilotait des LED SMD standard, de couleur jaune ; aujourd'hui tout cela est remplacé par des LED de 3 mm à électronique intégrée.

Le CAN intégré dans le μC mesurait la tension aux bornes d'une CTN pour détecter le souffle censé éteindre la bougie. La température de la respiration est à peu près la même que la



température ambiante, donc le fait de souffler sur une CTN dans des conditions normales n'a qu'un effet minime sur sa résistance. L'astuce de l'ancien projet était de réchauffer la CTN par sa propre dissipation de puissance pour la refroidir par le courant d'air du souffle, ce qui se traduit par un changement de tension détectable sur l'entrée analogique du μC . Nous voulions que la bougie de notre nouveau projet se contente d'une seule batterie de 1,5 V, dont l'énergie serait le plus possible consacrée à l'effet lumineux, au lieu de gaspiller 15 mA à chauffer le capteur.

De plus, l'abandon du PIC (et de son logiciel) utilisant l'électronique analogique a facilité la conception et la construction de la bougie.

Le TAPIR, fameux détecteur de rayonnement électromagnétique (*Totally Archaic but Practical Interceptor of Radiation*) [2] date de 2012. Nous avons surtout réutilisé ce projet pour l'originalité de sa construction mécanique : un boîtier fait de circuits imprimés et contenant, entre autres, un support de pile pour une pile AAA et une prise jack stéréo. Dans notre nouveau projet, la prise jack connecte un petit circuit imprimé en forme de flamme qui contient un capteur et quatre LED scintillantes de 3 mm, imitant une bougie. À condition d'accepter qu'un boîtier parallélépipédique puisse passer pour une bougie. Rien n'empêche de l'habiller mieux si cette apparence ne vous convainc pas. Il suffit que le bouton et le compartiment de la pile restent accessibles.

LED spéciales

Les LED scintillantes utilisées dans ce nouveau projet de Noël sont bon marché et facilement disponibles sur l'internet et à des prix microscopiques dans les boutiques asiatiques en ligne. L'électronique intégrée imite la lumière scintillante d'une vraie flamme de bougie, seule une résistance externe de limitation du courant est nécessaire. De l'extérieur, elles ressemblent à des LED ordinaires de 3 ou 5 mm et sont disponibles en différentes couleurs. Pour notre bougie de Noël, c'est la version 3 mm, de préférence jaune ou orange, qu'il nous faut.

Ces composants sont utilisés dans (presque) toutes les bougies électroniques à piles bon marché qui ne contiennent que les LED, une résistance et une pile bouton de 3 V ou deux piles AA ou AAA. Si vous ne trouvez pas de fournisseur pour les LED seules, achetez quelques-unes de ces bougies et récupérez les LED scintillantes pour les utiliser dans ce projet.

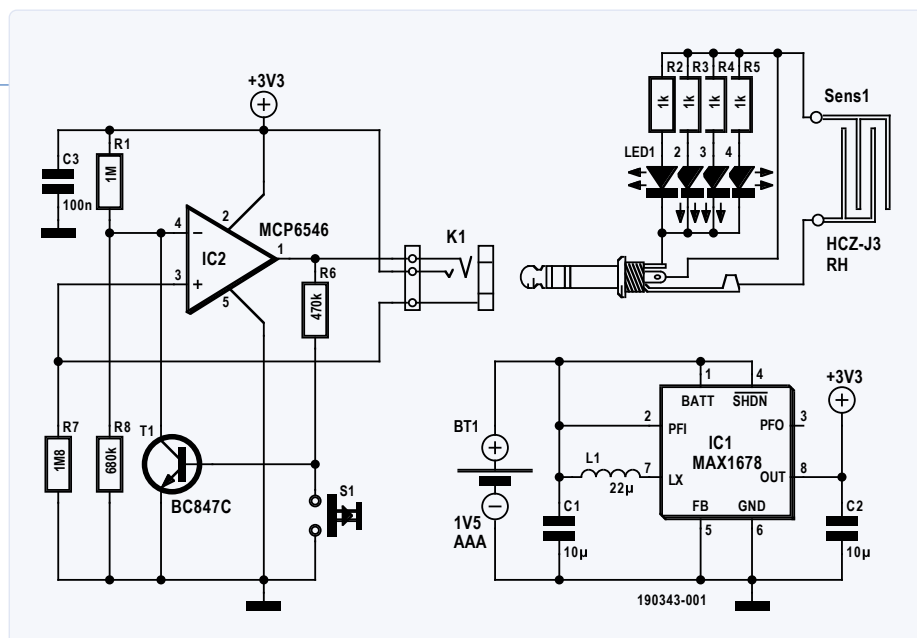


Figure 1. Le schéma de la bougie de Noël.

Et un peu plus de matériel

La tension de 1,5 V d'une seule batterie ne suffit pas pour allumer correctement les LED scintillantes. Un petit convertisseur élévateur (IC1, MAX1678) la porte à 3,3 V (fig. 1). Avec une tension d'entrée minimale de seulement 0,7 V, cette puce est parfaite pour épuiser les batteries déjà trop déchargées pour servir dans d'autres appareils. Sur notre prototype, une tension d'entrée

de seulement 300 mV suffit pour allumer la bougie ! Idéal pour essorer les vieilles piles que vous croyiez à plat.

Pour un appareil à piles, la mesure de la température pour détecter le souffle censé éteindre la bougie n'est pas une solution. Pour que l'objet final reste simple, petit et abordable, c'est un capteur d'humidité HCZ-J3 (Multicomp) qui détecte le souffle.

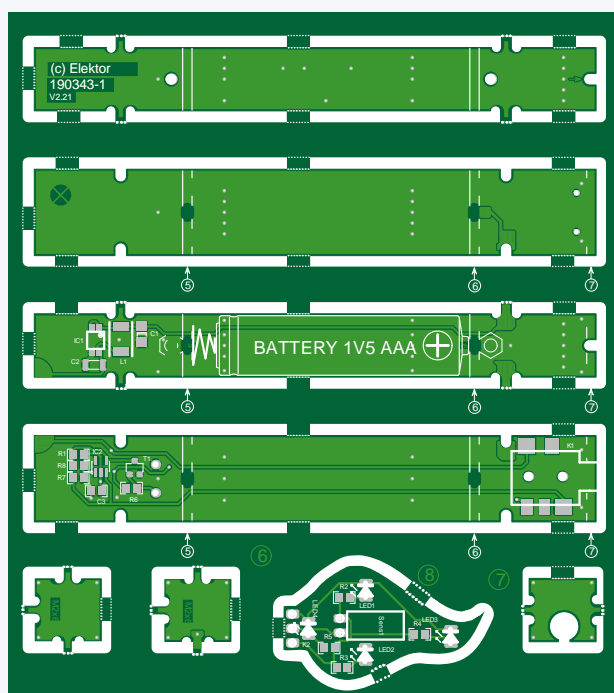


Figure 2. Huit circuits sont combinés en une seule carte.

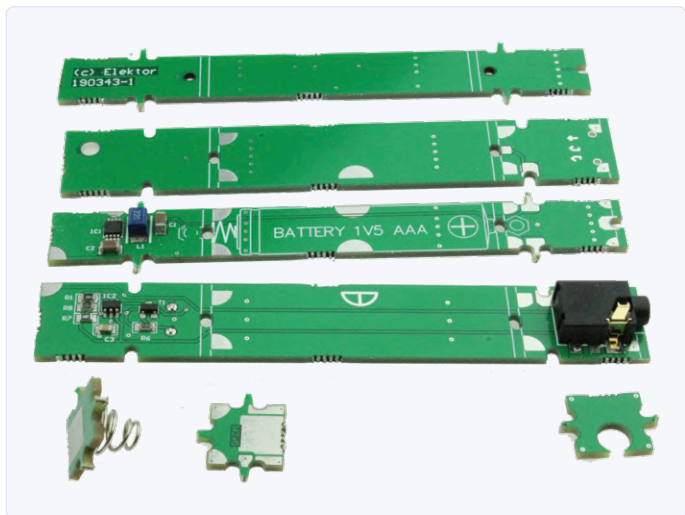


Figure 3. Soudez la plupart des composants avant de découper la carte.



Figure 4a. Le circuit imprimé de la flamme, les LED et le capteur avec la prise jack connectée.

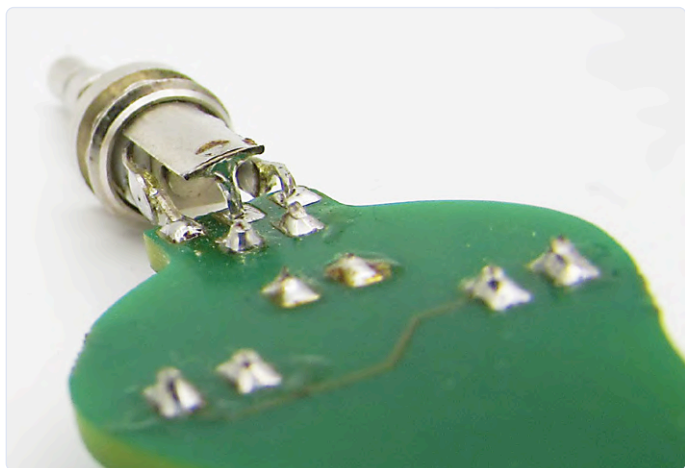


Figure 4b. Interconnexion du circuit des LED avec la prise jack.

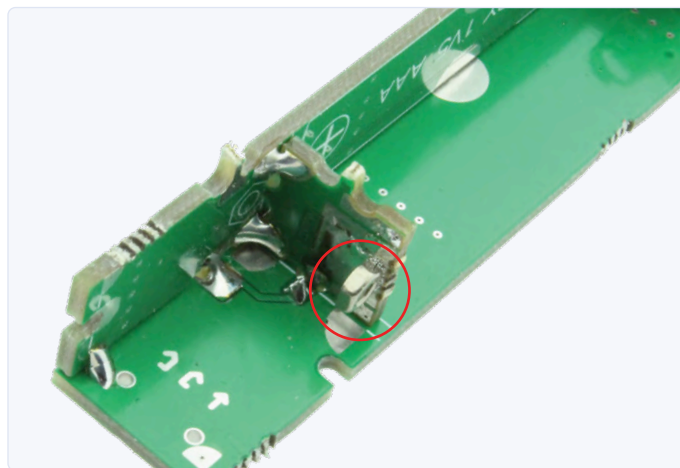


Figure 5. Circuits de contact de la batterie montés, remarquez l'écrou pour la fixation du couvercle.

Ces composants sont conçus pour être utilisés dans des circuits à courant alternatif, mais fonctionnent aussi ici. Le seul inconvénient d'un capteur d'humidité est qu'il faut quelques secondes avant que sa résistance ne revienne au niveau d'humidité ambiante. Il faut donc attendre un peu avant de rallumer la bougie.

Le reste de l'électronique est simple : un comparateur à faible consommation (IC2, MCP6546) surveille sur son entrée non-inverseuse la tension qui suit l'accroissement du taux d'humidité mesuré par le capteur quand on souffle dessus. Si elle devient supérieure à la tension sur l'entrée inverseuse ($\pm 1,35$ V du diviseur de tension R1 et R8), la sortie du comparateur passe au niveau haut. Le transistor T1 met alors l'entrée inverseuse au

niveau de la masse, ce qui verrouille les LED à l'état éteint jusqu'à ce que le poussoir S1 soit actionné, à condition, bien entendu, que la tension sur l'entrée non-inverseuse soit repassé sous le seuil de 1,35 V.

Réalisation de la bougie

Note sur le comparateur MCP6546 : lorsque vous commandez ce composant, assurez-vous d'obtenir le MCP6546R. Ce suffixe est très important, car Microchip produit trois versions SOT23-5 dont les brochages ne sont PAS compatibles ! La version 'R' est celle qu'il vous faut ici.

Pour en simplifier la production, la bougie et son boîtier sont constitués de huit petits circuits imprimés réunis en une seule carte (fig. 2). Téléchargez les fichiers Gerber sur

le site d'Elektor et commandez la carte au fournisseur de PCB de votre choix.

Pour votre confort, laissez cette carte intacte jusqu'à ce que (la plupart) des composants soient soudés (fig. 3). Les circuits imprimés sont numérotés (de un à huit) et il y a une sorte de légende sur la carte qui indique la place de chaque circuit.

Le masque de soudure vert comporte des ouvertures de couleur argentée sur plusieurs bords et coins des circuits. Ce sont ces zones qui permettent de relier les cartes par soudure. Certains joints ne servent qu'à maintenir les cartes ensemble, d'autres assurent les connexions entre circuits.

La plupart des composants utilisés dans ce projet sont montés en surface, mais il est possible de les souder avec un fer normal à pointe fine et du fil de soudure fin. Le travail

sera facilité si vous disposez d'un distributeur de pâte à souder, d'un outil de soudure à air chaud ou d'un four de refusion, mais il est réalisable aussi avec des outils moins spécialisés. Dans tous les cas, la tresse de dessoudage et le flux de soudure seront utiles pour éliminer l'excès de soudure et les courts-circuits presque inévitables, en particulier entre les broches des deux minuscules puces. Commencez par elles et vérifiez les soudures avant d'ajouter les autres puces CMS. Vous finirez par les composants traversants.

Avant de souder K2, la fiche jack stéréo de 3,5 mm qui relie le circuit imprimé de la flamme à la bougie, ce circuit doit être séparé de l'ensemble. Utilisez de courts morceaux de fil pour souder le circuit imprimé à la fiche et vérifiez l'interconnexion (fig. 4).

Pour fermer le «couvercle» du compartiment de la batterie, vous devez en principe souder deux colonnettes ou écrous sur les plus petits circuits (les contacts de la pile) (fig. 5). Cependant, nos prototypes ont montré que cela n'était pas indispensable, le boîtier est suffisamment serré pour que le couvercle reste en place sans vis.

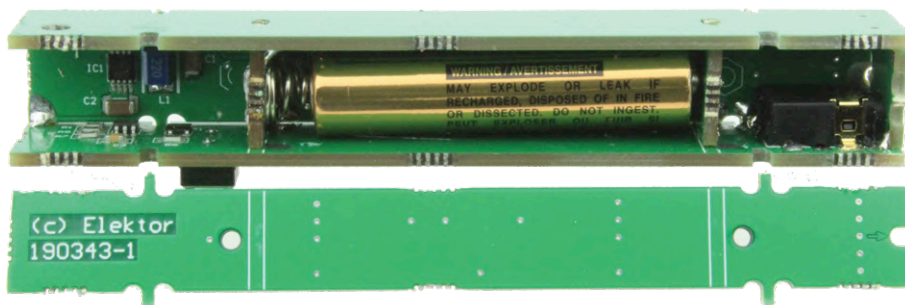
Une fois les composants implantés, il est temps de détacher toutes les cartes, mais ne soudez pas encore le boîtier ! Commencez par disposer les circuits sans les connecter, mais vérifiez où et comment chaque circuit va prendre sa place. Pour obtenir un agencement parfait, il faudra peut-être ébarber les bords de la carte ou la limer.

Si vous avez soudé des colonnettes ou des écrous pour fermer, c'est le moment de vérifier leur alignement avec les trous du couvercle. Ajustez-les si nécessaire. Ensuite, soudez quelques fils pour interconnecter les circuits et utilisez une alimentation de labo pour les tests. Vous aurez plus d'espace pour les mesures et d'éventuelles reprises de soudure, si nécessaire. Il faudra retirer ces fils plus tard. Puis, lorsque la bougie fonctionnera correctement, les circuits seront enfin soudés pour de bon.

Votre bougie est prête. Pour créer une ambiance de Noël, rien ne vaut une vraie bougie, mais nous espérons que vous appré-

cierez cette version électronique. Nous vous souhaitons beaucoup de plaisir à construire ce gadget et de toute manière un très joyeux Noël ! ❄️

190343-04



Votre avis, s'il vous plaît...

Vous pouvez adresser vos questions ou vos commentaires à la rédaction : redaction@elektor.fr



LISTE DES COMPOSANTS

Résistances

(SMD 0805)

R1 = 1 MΩ

R2,R3,R4,R5 = 1 kΩ

R6 = 470 kΩ

R7 = 1,8 MΩ

R8 = 680 kΩ

Inductance

L1 = 22 μH, 1812, TDK NLC453232T-220K-PF

Condensateurs

C1,C2 = 10 μF, 10%, 25 V, X7R, 1206

C3 = 100 nF, 10%, 50 V, X7R, 0805

Semi-conducteurs

LED1,LED2,LED3,LED4 = LED jaune 3 mm pour bougie scintillante (voir texte)

T1 = BC847C

IC1 = MAX1678EAU+
régulateur élévateur de tension

IC2 = MCP6546RT-E/OT
comparateur analogique

Divers

K1 = connecteur jack stéréo 3,5 mm SMD
(CUI SJ1-3514-SMT-TR)

K2 = prise jack stéréo 3,5 mm
contact de batterie (Keystone Electronic 211)

S1 = bouton-poussoir 6 mm
(Alcoswitch FSM2JRT)

Sens1 = capteur d'humidité HCZ-J3
(Multicomp)

PCB 190343-1 V2.21



Dans l'e-shop d'Elektor :

➤ Fichiers Gerber PCB à télécharger gratuitement : www.elektormagazine.com/190343-01

LIENS

- [1] Bougie électronique à LED, Elektor décembre 2011 : www.elektormagazine.fr/magazine/elektor-201112/12016
- [2] Elektor TAPIR, détecteur ultrasensible d'e-smog à large bande ! : Elektor juillet-août 2012 : www.elektormagazine.fr/magazine/elektor-201207/12196
- [3] Téléchargement :: <http://www.elektormagazine.fr/190343-04>

générateur sinusoïdal accordable à tubes

de 20 Hz à 20 kHz sur la même gamme !

Martin Ohsmann (Allemagne)

Dans ma jeunesse, j'ai construit ce que l'on appelle un récepteur oV2 à tubes. Nostalgique de l'époque d'avant les transistors, j'ai décidé de construire un autre circuit utilisant des tubes. Les études de récepteurs simples et d'amplificateurs haut de gamme se comptent déjà par dizaines, voire par centaines. Cela m'a poussé vers quelque chose d'inhabituel.

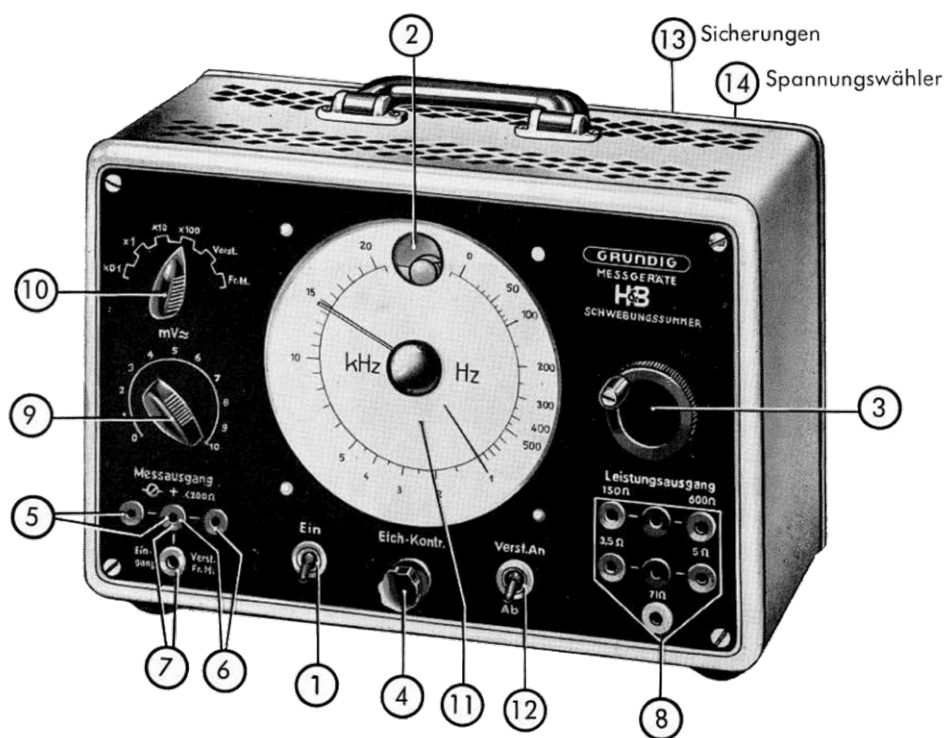


Figure 1. Oscillateur à fréquence de battement, modèle 295A.

C'est dans cet état d'esprit que je suis tombé sur le manuel de l'oscillateur à fréquence de battement modèle 295A de Grundig/Hartmann et Braun (**fig. 1**), qui peut être accordé en un seul balayage de toute la bande audio de 20 Hz à 20 kHz sans changer de gamme. Ce rapport de fréquences énorme de 1000:1 est irréalisable avec de simples générateurs d'ondes sinusoïdales. Ce qui nous conduit au...

Principe de l'hétérodyne

La **figure 2** en illustre l'idée de base. L'oscillateur 1 délivre une fréquence fixe f_1 , ici environ 235 kHz. L'oscillateur 2 est réglable et délivre une fréquence f_2 variable entre 235 kHz à 270 kHz, soit une variation de la fréquence de moins 20 %. Il est facile de mettre en œuvre une gamme de fréquences

aussi étroite dans un simple oscillateur à l'aide d'un condensateur variable.

Les signaux de sortie des deux oscillateurs sont envoyés à un mélangeur à la sortie duquel on trouve, entre autres produits, la fréquence de **différence** $f_1 - f_2$. Les autres produits, comme la somme de f_1 et f_2 , ainsi que les sommes et les différences de multiples entiers de ceux-ci, peuvent être éliminés à l'aide d'un filtre passe-bas approprié. Enfin, nous amplifions le signal de sortie pour obtenir une source à faible impédance. Il est possible d'utiliser ce principe dit *hétérodyne* (en grec, *hetero* = différent et *dynamis* = force) pour construire des oscillateurs RF accordables jusque dans les GHz.

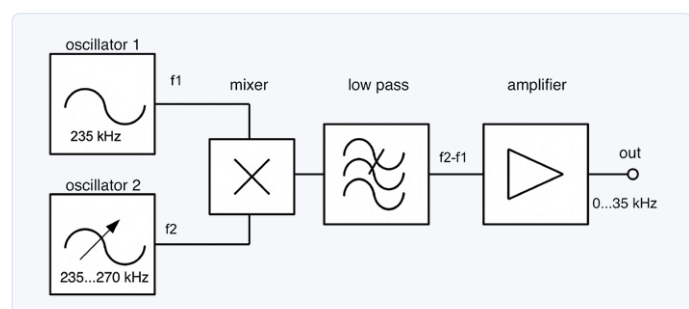


Figure 2. Principe de fonctionnement du générateur de signal hétérodyne.

Figure 3. Extrait du schéma du modèle 295A.

L'original

La **figure 3** montre un extrait du schéma, comprenant les deux oscillateurs, le mélangeur, le filtre passe-bas et l'étage d'amplification final. Le tube appelé Rø2 produit la fréquence fixe. Son anode est connectée à un circuit résonnant qui détermine la fréquence d'oscillation, et il y a un transformateur pour fournir une rétroaction à l'électrode de grille. La fréquence variable est obtenue de la même manière en utilisant la section triode d'un ECH81 (partie gauche du tube Rø1). La fréquence est ajustée à l'aide de la self variable L5. Les sorties des deux oscillateurs sont ensuite acheminées vers les grilles G1 et G3 de la section heptode de l'ECH81, et les produits du mélangeur apparaissent à l'anode. Le filtre passe-bas est constitué de L1, L2 et L3 ainsi que de C1, C2, C3 et C4. Le signal de sortie filtré passe ensuite à travers un potentiomètre pour permettre le réglage du niveau du signal du tube Rø3, configuré en suiveur de cathode. Le signal de sortie sinusoïdal à faible impédance est disponible à la cathode.

Ma version

Pour en faciliter la construction, j'ai essayé des modifications. Au lieu d'une inductance variable pour ajuster la fréquence, j'ai utilisé un condensateur variable ; et j'ai modifié le circuit de rétroaction dans les oscillateurs pour utiliser des inductances à prises intermédiaires au lieu de transformateurs (**fig. 4**).

Sur le schéma complet (**fig. 5**) la triode V1 est l'oscillateur à fréquence variable, et son signal de sortie passe via un potentiomètre à la grille G3 du tube mélangeur V2. La section triode de V2 est utilisée pour former l'oscillateur à fréquence fixe, et son signal de sortie est acheminé vers la grille G1 de l'heptode. La sortie du mélangeur apparaît sur l'anode de l'heptode.

Figure 4. Mes expérimentations en cours.

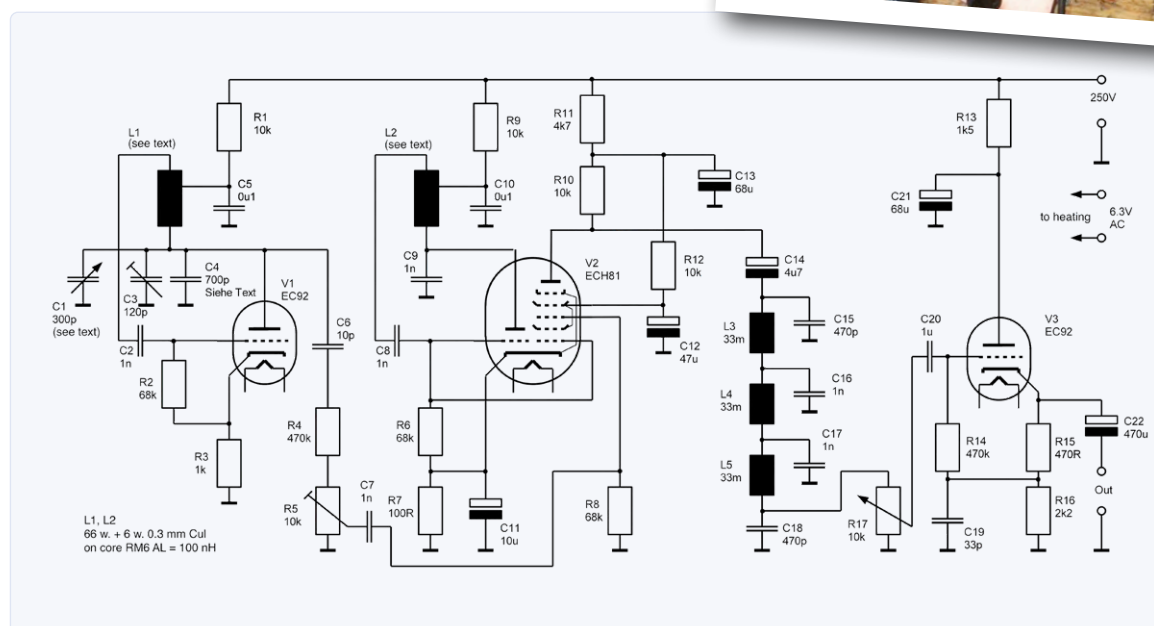
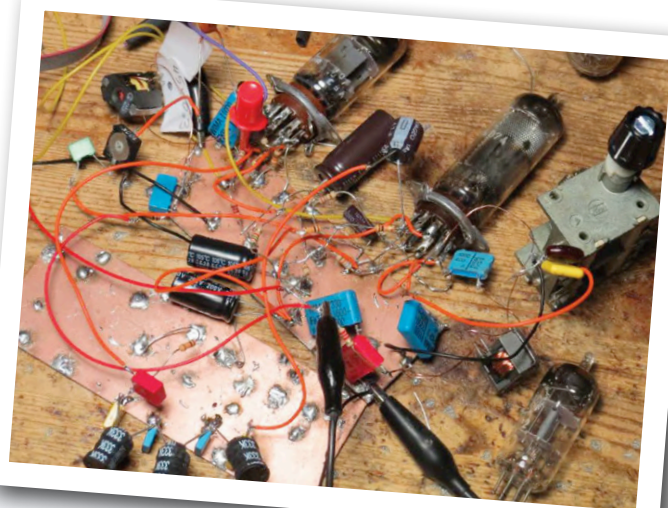
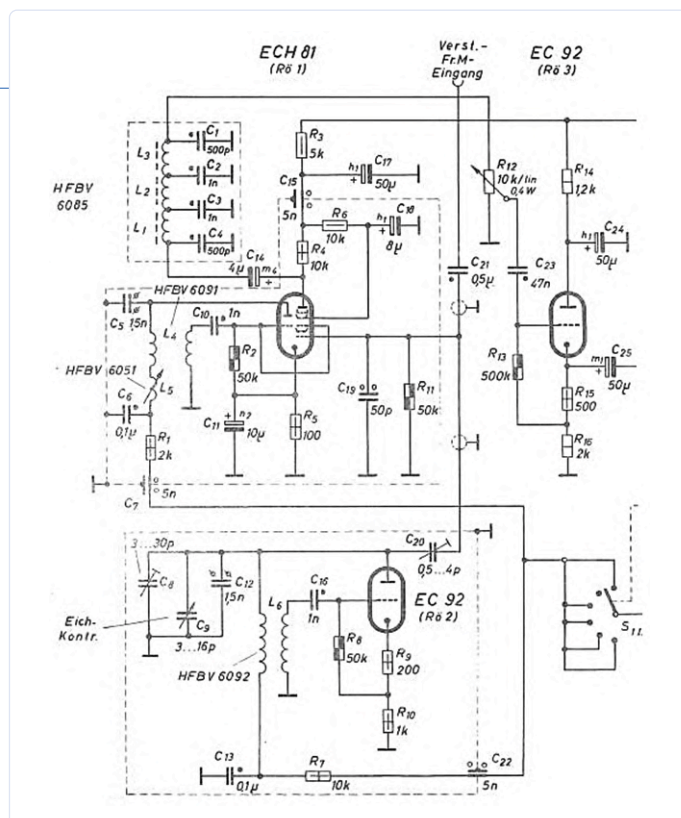


Figure 5. Schéma du circuit de ma version optimisée. Voir le proto (sans alimentation) sur la figure 6.

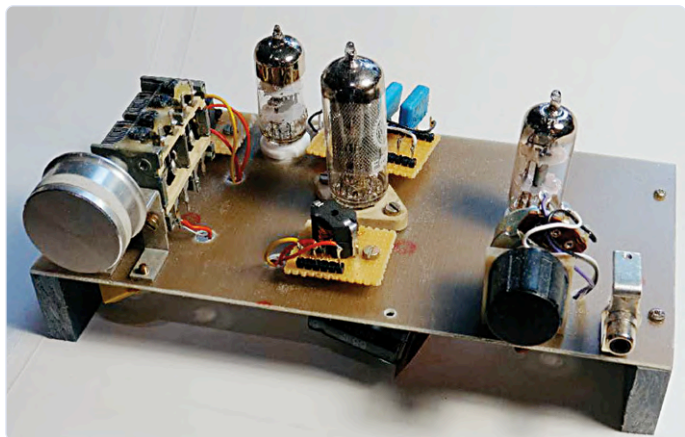


Figure 6. Prototype final du générateur d'ondes sinusoïdales.

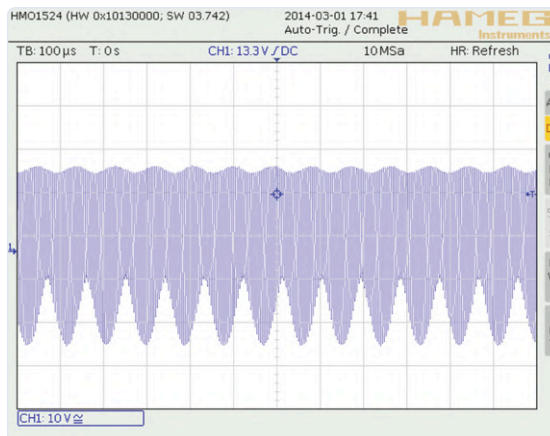


Figure 7. Signal à l'anode du mélangeur.

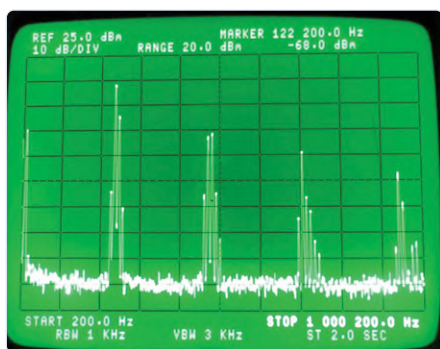


Figure 8. Spectre du mélangeur.

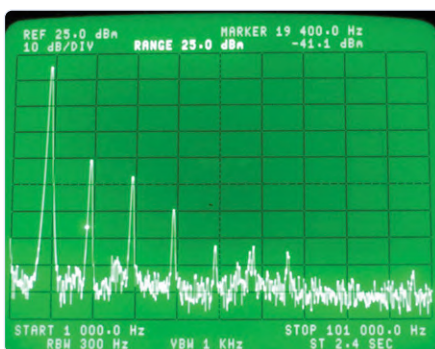


Figure 9. Spectre à haut niveau de modulation du mélangeur.

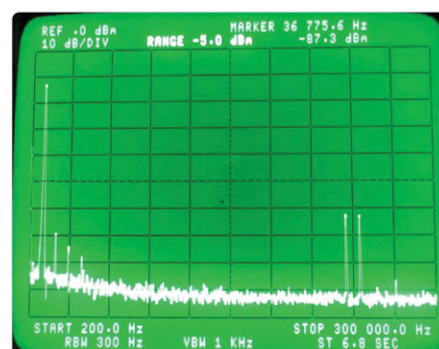


Figure 10. Spectre à bas niveau de modulation du mélangeur.

Mesures

Pour vérifier le fonctionnement de base et la qualité du signal de mon prototype final (fig. 6), j'ai effectué quelques mesures. J'ai mesuré une amplitude de signal d'environ $200 V_{c\grave{a}c}$ sur les anodes de l'oscillateur. Regardez bien le signal de l'anode sur le tube mélangeur (fig. 7). Les oscillations rapides sont les deux fréquences élevées, tandis que l'enveloppe inférieure représente le signal sinusoïdal différentiel : c'est celui-ci que nous extrayons ensuite avec le filtre passe-bas. Dans le spectre du signal de l'anode (fig. 8) sur le tube mélangeur, on voit les nombreux produits du mélange des deux oscillateurs. Le réglage du potentiomètre R5 modifie l'amplitude du signal de sortie, mais en même temps, il modifie le niveau de modulation du tube de mélange. Si nous réglons R5 pour que l'amplitude du signal soit par exemple de $8 V_{c\grave{a}c}$ sur la grille G3, nous obtenons une amplitude de sortie d'environ $7 V_{c\grave{a}c}$. À un niveau de sortie aussi élevé, nous obtenons bien sûr une distorsion considérable du signal de sortie, comme le montre le spectre de la fig. 9. La deuxième harmonique est déjà à environ 35 dB seulement sous la fondamentale, et le taux de distorsion de plus de 1 % n'est pas particulièrement impressionnant.

Limitons l'amplitude du signal sur G3 à $1 V_{c\grave{a}c}$, et nous aurons un signal de sortie de $2 V_{c\grave{a}c}$ dont le spectre (fig. 10) est beaucoup plus présentable. La deuxième harmonique est maintenant à plus de 50 dB de la fondamentale, et la distorsion réduite à environ 0,3 %, ce qui est satisfaisant. Les résidus à 235 kHz et 250 kHz résultent d'un couplage capacitif du signal de sortie via le filtre passe-bas, que l'on pourrait atténuer en améliorant le blindage RF du circuit.

Il est donc possible, avec seulement trois tubes et des bricoles, de réaliser un générateur d'ondes sinusoïdales à fréquence variable sur toute la plage audio.

PS : Précisons pour les électroniciens qui ne connaissent peut-être pas le jargon RF des radioamateurs, que dans l'abréviation **0V2** le premier chiffre indique le nombre d'étages de préamplification RF avant le détecteur, tandis que le chiffre après le V indique le nombre d'étages d'amplification audio. Ainsi, un récepteur 0V2 est un modèle *audion* à un tube, sans préampli RF, et deux étages d'amplification audio. ◀

190312-03



@ WWW.ELEKTOR.FR

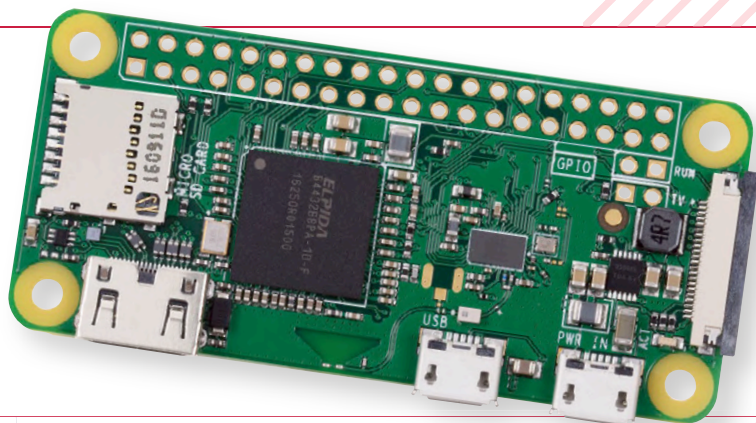
➤ **Röhren-Projekte von 6 bis 60 V** (= « Projets de circuits à tubes de 6 V à 60 V » - Ce livre au format PDF n'existe qu'en allemand)
www.elektor.de/roehren-projekte-von-6-bis-60-v-pdf

➤ **Röhrenverstärker-Schaltungen** (= « Circuits d'amplis à tubes » - Ce livre au format PDF n'existe qu'en allemand)
www.elektor.de/roehrenverstaerker-schaltungen-pdf

➤ **JOY-iT JDS6600 Générateur de signaux (60 MHz) et fréquencesmètre (100 MHz)**
www.elektor.fr/joy-it-jds6600-signal-generator-frequency-counter

ABONNEZ-VOUS ET RECEVEZ

Raspberry Pi Zero W GRATUIT



**TOUS LES 2 MOIS, LES
DERNIÈRES NOUVELLES DU
RASPBERRY PI ET LES
MEILLEURS PROJETS !**

**SEULEMENT
54,95 €
PAR AN
(6 NUMÉROS)**



**Souscrivez dès maintenant
un abonnement d'un an
au magazine MagPi, nous
vous offrons :**

- Six numéros du magazine MagPi
- Une carte Raspberry Pi Zero W
- Boîtier Pi Zero W

Vos avantages :

- Prix au numéro réduit
- Chaque numéro directement dans votre boîte aux lettres
- Tous les numéros disponibles sous forme numérique (PDF)
- Cadeau de bienvenue d'une valeur de 23,90 €
- Découverte de chaque nouveau numéro avant sa sortie en kiosque



ABONNEZ-VOUS : WWW.MAGPI.FR

générateur DDS Rigol DG4162



Philippe Demerliac | Cyrob (France)

Vous voudriez un générateur de formes d'onde fiable, moderne, avec beaucoup de possibilités ? La série DG2000 de Rigol pourrait être une bonne solution. Son interface tactile hybride et son excellente facture industrielle en font une option sérieuse pour votre labo. L'Ethernet et l'USB standard vous permettront de piloter votre instrument par programmation. Examinons de plus près le générateur DG2072.

Philippe Demerliac anime Cyrob.org et publie des vidéos sur sa chaîne https://youtu.be/tm_VqCWMY34.



Figure 1. La façade du DG2072.

Figure 2. L'arrière du DG2072.

Ma note globale pour l'instant est de **18/20**. J'ai hésité à mettre ce généré dans ma catégorie **"Pro"** puisque son prix et ses performances le placent dans le matériel de labo professionnel. Pour un amateur, c'est un gros investissement, mais il devrait couvrir longtemps la plupart des besoins.

J'avais acheté il y a quelques années le générateur DDS Rigol DG4162, déjà d'excellente facture. Le DG2072 est encore meilleur !

Il possède une interface tactile. C'est devenu la norme sur les instruments de mesure, et pour cause ! Il m'arrive souvent d'appuyer sur l'écran d'appareils qui en sont dépourvus. Quelle frustration ! Les mesures que j'ai faites montrent qu'il est bien calibré et qu'il tient ses spécifications !

La série DG2000 innove, avec le mode *DualTone*, la production de signaux logiques et le mode *séquence* qui sont autant d'atouts utiles. Si vous cherchez un générateur DDS de qualité, la série DG2000

de Rigol sera un excellent investissement. Le prix dépasse certes celui de modèles concurrents, mais il est justifié par la variété des fonctions et la qualité des signaux.

Selon la limite de fréquence, le prix change. À mon sens, la version 50 MHz devrait suffire à la plupart des amateurs.

Certains percevront le carénage comme agressif, c'est une question de goût.

Plus et moins

J'ai apprécié :

- l'interface tactile agréable et rapide
- les possibilités de modulation
- la grande variété des fonctions



Figure 3. L'écran de démarrage.



Figure 4. La voie 2 en mode harmonique.



Figure 5: Le marqueur bien commode en mode wobulation.

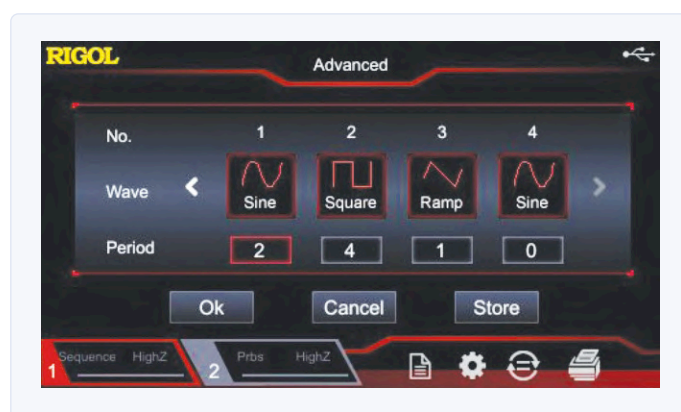


Figure 6. Le mode séquence enchaîne plusieurs formes d'ondes en mode Direct.

- > le grand nombre de formes d'ondes standard
- > le choix de l'impédance de sortie
- > le choix de l'amplitude en V_{pk} , V_{rms} , dBm, etc...
- > la fonction Marker en mode Wobulation
- > les modes avancés (Bit, RS232, séquence)
- > la fonction Dual Tone
- > la technologie SiFi II pour une meilleure restitution
- > le "vrai" interrupteur en face avant
- > le silence (pas de bruit de ventilateur)
- > la qualité de l'écran tactile et la lisibilité parfaite
- > la résolution de 16 bits (65536 possibilités, soit 152 μV par point sous 50 Ω)
- > 16 mégapoints de longueur d'échantillon et la vitesse de 250 MSa/s
- > le faible bruit de phase (10 MHz : < -105 dBc/Hz)
- > la qualité de l'ensemble

J'ai regretté ou moins aimé

- > les prises à l'arrière
- > les prises à l'arrière tantôt en entrées tantôt en sorties selon le mode
- > l'interrupteur bon marché en façade (pourquoi pas un poussoir ?)
- > la dureté des boutons en face avant
- > les restrictions du mode fréquencesmètre:
- > l'affichage trop petit de la fréquence, l'impossibilité de définir un offset, l'impossibilité de faire des copies d'écran, la dynamique d'entrée limitée

Premiers contacts

Voici la face arrière (fig. 1). Attention, selon le mode, les BNC seront des entrées ou des sorties (fig. 2).

Et voici l'écran lors de la mise en route (fig. 3), la voie 1 est rouge, la 2 est bleue. La voie 2 en mode harmonique (fig. 4). Le Marker est dispo en mode Wobulation ce qui en facilite l'utilisation (fig. 5). Le mode séquence permet d'enchaîner plusieurs formes d'ondes en mode Direct ce qui élimine les défauts de l'accumulation de phase (fig 6). La simulation de séquences pseudo aléatoires de bits série est bien utile pour la mise au point de systèmes de transmission numériques (fig. 7).

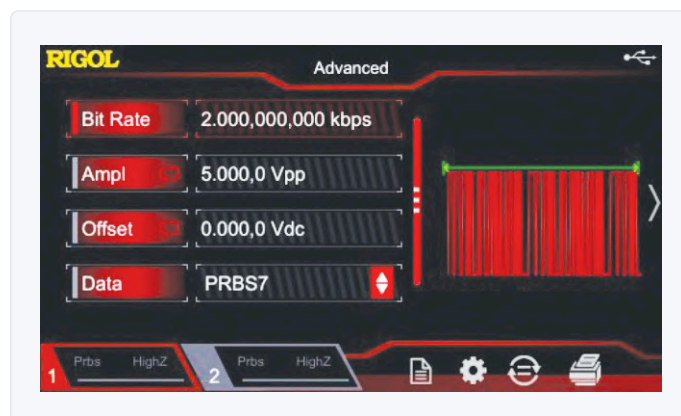


Figure 7. La simulation de séquences pseudo aléatoires de bits série est bien utile pour les transmissions numériques.

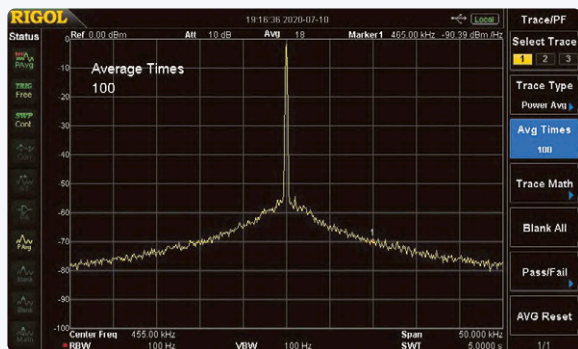


Figure 8. Analyse spectrale

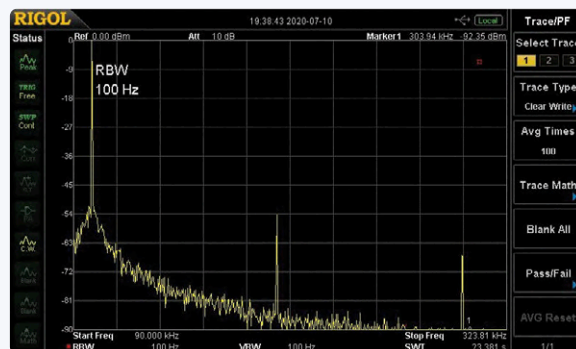


Figure 9. Spectre pour une sinusoïde de 100 kHz à 0 dBm.

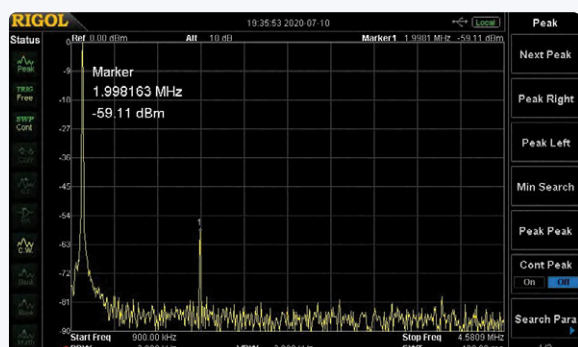


Figure 10. Spectre pour une sinusoïde de 1 MHz à 0 dBm.

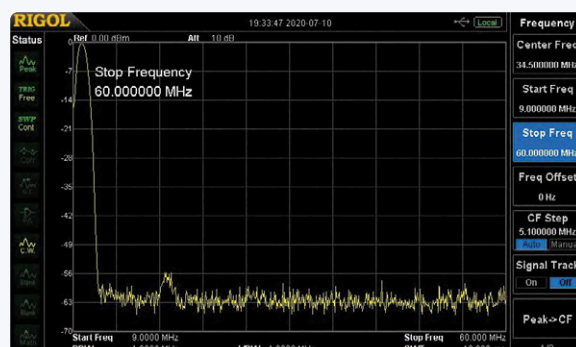


Figure 11. Spectre pour une sinusoïde de 10 MHz à 0 dBm

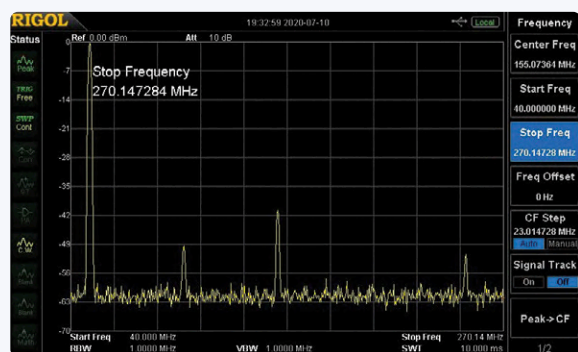


Figure 12. Spectre pour une sinusoïde de 50 MHz à 0 dBm

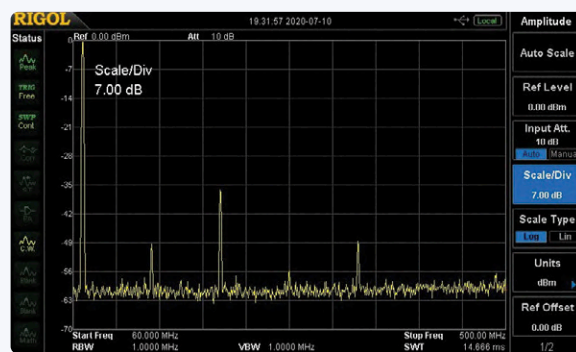


Figure 13. Spectre pour une sinusoïde de 70 MHz à 0 dBm

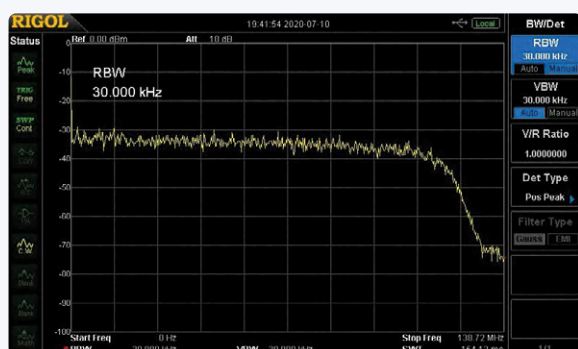


Figure 14. En mode bruit blanc, le niveau est quasi stable jusqu'à 100 MHz.

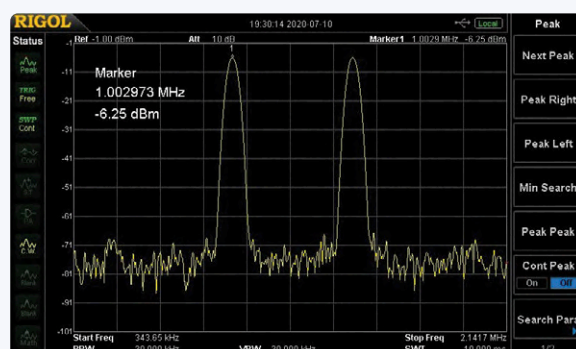


Figure 15. Le dos de chameau du mode Dual Tone.



Figure 16. J'ai ramené les connecteurs vers l'avant.

Analyse spectrale

J'ai fait quelques mesures avec l'analyseur de spectre DG815TG Rigol. Tentative de mesure du bruit de phase normalisé à 10 kHz de la porteuse, mais mon analyseur manque de sensibilité... (fig. 8).

Dans le spectre de sortie pour une sinusoïde de **100 kHz** à 0 dBm (fig. 9), la première harmonique est à -54 dB. Dans le spectre de sortie pour une sinusoïde de **1 MHz** à 0 dBm (fig. 10), la première harmonique est à -59 dB. Dans le spectre de sortie pour une sinusoïde de **10 MHz** à 0 dBm (fig. 11), la première harmonique est à -56 dB. Dans le spectre de sortie pour une sinusoïde de **50 MHz** à 0 dBm (fig. 12), la première harmonique est à -49 dB, la 3e plus forte à -40 dB. Dans le spectre de sortie pour une sinusoïde de **70 MHz** à 0 dBm (fig. 13), la première harmonique est à -49 dB, la 3e plus forte à -37 dB.

En mode générateur de bruit blanc, le niveau est quasi stable jusqu'à 100 MHz, ce qui est excellent (fig. 14)! Et enfin le mode Dual Tone avec ses bosses de chameau (fig. 15).

Modifications

Je n'aime pas les connecteurs à l'arrière et me suis donc empressé de les ramener à l'avant ! (fig. 16)! Je n'ai rien modifié dans l'appareil, juste profité des vis de fixation des protections latérales (fig. 17). Il a fallu usiner une pièce un peu zarbi (fig. 18 et 19). Pour le repérage, les gaines thermo imprimables sont idéales (fig. 20). ❏

200453-04



Figure 17. Les connecteurs fixés sur le côté.

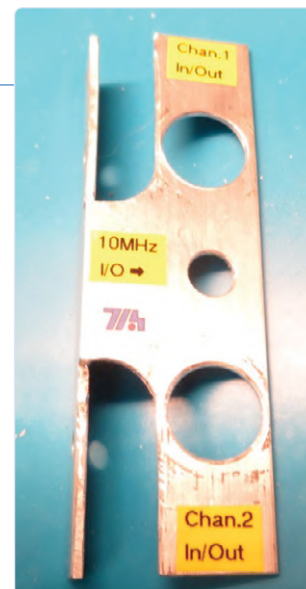


Figure 18. Un peu de serrurerie...



Figure 19. Fixation de la pièce usinée à la main.



Figure 20. Les gaines thermo imprimables.



DANS L'E-CHOPPE D'ELEKTOR :

> Rigol DG2072 Function/Arbitrary Waveform Generator
www.elektor.fr/dg2072

Votre avis, s'il vous plaît...

Vous pouvez adresser vos questions et vos commentaires à l'auteur (philippe.demerliac@free.fr) ou à la rédaction d'Elektor (redaction@elektor.fr)

Banc d'essai vidéo

Philippe Demerliac, alias Cyrob a publié un banc d'essai vidéo du DG2072 : https://youtu.be/tm_VqCWMY34

LIEN

[1] Cyrob, "Revue du DG2072 Rigol, merci Elektor!," July 11, 2020 : https://youtu.be/tm_VqCWMY34

Ont contribué à cet article :

Auteur : **Philippe Demerliac**
 Rédaction : **C. J. Abate**

Traduction : **Catherine Rappin**
 Maquette : **Giel Dols**

greffons et modules de



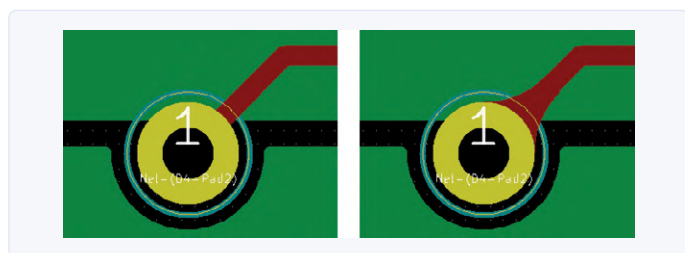
KiCAD, outil gratuit et libre pour la conception de schémas et de circuits imprimés, dont je vous ai déjà vanté bien des mérites, est extensible. Ce n'est pas la moindre de ses qualités. En plus, n'importe qui peut créer des greffons et des modules complémentaires en Python. En voici quelques-uns.

KiCAD est un outil gratuit et à code source ouvert d'assistance à la création de schémas et à la conception de circuits imprimés (PCB). Depuis sa sortie en 1992, il gagne en popularité. Chez Elektor aussi. Qu'il soit extensible nous paraît très intéressant, puisque tout(e) pythonneur ou pythonneuse averti(e) peut concevoir pour lui des greffons et des modules complémentaires. Cela a permis de partager en ligne une multitude d'outils et de fonctions auxiliaires. Selon votre expérience et vos besoins, certains vous paraîtront plus utiles que d'autres.

Notre sélection

Voici quelques greffons et boîtes à outils choisis pour vous. Une belle liste de greffons et d'utilitaires pour KiCAD est tenue à jour sur <https://github.com/xesscorp/kicad-3rd-party-tools>

Stretch - Plug-in d'action pour faire de jolis PCB. Il a beau être un programme de dessin, KiCAD n'est pas très doué pour cela. Pour les lignes droites oui, mais pas les courbes ni les formes complexes. Pour y remédier, il permet d'échanger des tracés de



Greffon d'action "Teardrops" avant (à gauche) et après (à droite). Vous pouvez le télécharger ici : https://github.com/NilujePerchut/kicad_scripts

PCB avec *Inkscape*. Les dessins typiques de PCB sont réalisés dans *Pcbnew*, puis la touche artistique et d'autres opérations complexes sont effectuées dans *Inkscape* avant de réexporter le dessin vers *Pcbnew* qui donnera les fichiers de production.

<https://github.com/JarrettR/Stretch>

KiKit - Boîte à outils Python pour la production automatique de panneaux, l'exportation de fichiers de production et la création de pages de présentation de panneaux. Pour faire bon usage de cette boîte à outils, une certaine connaissance de la programmation Python n'est pas inutile. En raison de problèmes Python liés à KiCAD, cette boîte à outils ne fonctionne pas sous Windows (cf. encadré).

<https://github.com/yaqwsx/KiKit>

BOM HTML interactif - Plugin d'action pour créer non seulement une nomenclature, mais aussi un document interactif d'implantation des composants, avec mise en évidence graphique de l'emplacement d'un composant sur le PCB. Les composants peuvent être recherchés par valeur ou individuellement. Ce greffon sait non seulement où mettre les composants, mais connaît tous les *nets*. Cela accélère considérablement la localisation d'un composant ou d'un signal pour l'assemblage, le test ou la réparation de la carte.

<https://github.com/openscopeproject/InteractiveHtmlBom>

RF-Tools for KiCAD - Cette collection de greffons d'action facilite la conception de cartes HF avec des fonctions comme l'arrondi des angles des pistes, le calcul de la longueur de piste, du blindage et des masques de soudure. Elle comporte également des utilitaires pour la création d'empreintes RF.

<https://github.com/easyw/RF-tools-KiCAD>

sonde de courant différentielle pour oscillo – 2.0

Mesurer les courants avec l'oscilloscope

Alfred Rosenkränzer

Voici une sonde de courant différentielle pour oscilloscopes à sortie asymétrique et à large bande passante. Le gain standard de 2 peut être augmenté en modifiant une seule résistance !

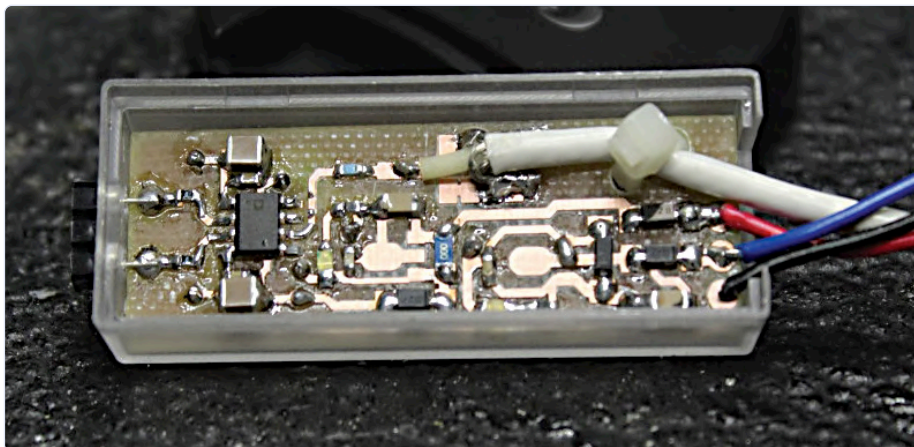


Figure 1. Prototype de la première version de la sonde [1].

INFOS SUR LE PROJET

Mots clés

Mesure, oscilloscope, courant

Niveau

débutant – connaisseur – expert

Temps nécessaire

environ 2 h

Outils

Outils de soudure (CMS et câblés)
Outillage mécanique

Coût

30 à 40 €

Le «2.0» du titre l'indique : ceci n'est pas la première sonde de l'auteur. La **fig. 1** montre le prototype de la sonde d'Alfred Rosenkränzer, publiée dans Elektor il y a quatre ans [1], mais qui traîne loin derrière celle présentée ici en termes de largeur de bande. La motivation de ce nouveau projet est donc évidente : cette sonde version 2.0 constitue un net progrès par rapport à la précédente dont la bande passante était limitée à 130 kHz. Ce gain se paie par une réduction de l'amplitude du mode commun acceptable sur les entrées, mais voyez vous-même !

Sonde de courant pour oscillo

L'oscilloscope, n° 2 incontesté dans la hiérarchie des instruments de mesure, présente non seulement l'avantage par rapport au multimètre n°1) de pouvoir être utilisé pour évaluer la dynamique d'un signal, mais aussi un inconvénient décisif : normalement, un oscilloscope ne peut pas mesurer directement les courants. Sauf avec une sonde de courant !

Si vous voulez mesurer le comportement dynamique du courant dans un circuit, vous avez un problème : bien sûr, vous pourriez insérer une résistance shunt quelque part (ou utiliser une résistance existante) et mesurer la chute de tension à ses bornes avec une sonde de tension normale. La sensibilité de l'amplificateur vertical des oscilloscopes, même bon marché, descend jusqu'à au moins 5 mV par division, et suffit pour de telles mesures. Malheureusement, le conducteur de masse de l'oscilloscope est alors relié à cette résistance, ce qui peut entraîner des perturbations gênantes. De plus, surtout avec un oscilloscope à deux canaux, il y a le danger de court-circuiter des parties du circuit en cas de mesure de tension simultanée sur le deuxième canal.

La solution du problème est un petit amplificateur différentiel qui mesure une différence de potentiel à l'abri du mode commun. Ce n'est pas un problème pour ces amplificateurs capables de rejeter les tensions de mode commun jusqu'à une certaine valeur. La sortie est alors la tension différentielle rapportée au potentiel de la masse de l'oscilloscope. De cette manière, de faibles tensions aux bornes des résistances dans un circuit sont accessibles à la mesure et, dans certaines limites, indépendamment d'une mesure de tension simultanée.

L'avantage de ce concept est l'existence de circuits intégrés conçus pour ça. Il suffit de quelques composants supplémentaires pour obtenir la sonde de courant.

Caractéristiques

En raison de sa large bande passante soulignée par le fabricant, jusqu'à 6 MHz, nous avons choisi l'amplificateur d'instrumentation AD8421 d'Analog Devices. À moins de 10 €, cette puce est non seulement peu coûteuse, mais convainc également par son faible niveau de bruit, sa faible consommation d'énergie et son rejet élevé du mode commun. Les caractéristiques détaillées se trouvent

dans l'encadré «**AD8421 - Caractéristiques techniques**», et plus encore dans la fiche technique du fabricant [2].

Malheureusement, l'AD8421 présente également un inconvénient majeur par rapport à la puce AD8479 utilisée dans la première version : alors que dans celle-ci, la puce avait une plage de mode commun de ± 200 à 600 V, les entrées de l'AD8421 utilisées ici doivent rester dans la plage de sa tension d'alimentation. Une tension d'alimentation de ± 15 V donne donc une plage de mode commun utilisable d'environ ± 12 V. À première vue, c'est peu, mais la puce qui coche toutes les cases reste à inventer. Il faudra donc choisir l'une ou l'autre version de la sonde, la 1.0 de 2016, pour les applications à tension d'alimentation élevée, et la 2.0 pour les signaux rapides et une basse tension d'alimentation. C'est la situation qui dictera le choix du bon outil.

Schéma de la sonde

Le concept d'Alfred Rosenkränzer a convaincu Elektor. Ton Giesberts, du labo d'Elektor, a donc repris son idée et l'a légèrement retravaillée. Le circuit résultant de la **fig. 2** est configuré pour un gain de 2, fixé par R5, une résistance unique. Le gain s'obtient par l'équation suivante :

$$A = 1 + (9,9 \text{ k}\Omega / R5)$$

Si l'on omet R5, la fiche technique indique un gain unitaire avec une largeur de bande de 10 MHz, puisque R5 est pratiquement infini dans ce cas. Avec la valeur choisie de $10 \text{ k}\Omega$ pour R5, on obtient un gain d'exactement 1,99, soit une erreur de $0,5 \%$ par rapport au gain de 2, ce qui est plus précis que les résistances utilisées. La largeur de bande à ce gain dépasse 6 MHz – mesurée, elle était même d'environ 8 MHz. Si vous préférez d'autres gains : même avec un gain de 100 , la largeur de bande de l'AD8421 atteint encore 2 MHz ! Les filtres d'entrée (R1/C1 et R2/C2) offrent une protection simple contre les surtensions et les transitoires pour les conditions de surcharge de courte durée. De plus, ils corrigent la légère augmentation du gain de l'AD8421 près de la fréquence de coupure, ce qui permet de linéariser la réponse en fréquence de la sonde.

Bien que l'AD8421 ait des entrées très robustes, les surtensions plus longues doivent être évitées. Répétons que la plage utilisable de mode commun sur les entrées est limitée par l'alimentation de ± 15 V à un peu plus de ± 12 V.

La sonde possède ses propres régulateurs de tension ainsi qu'un filtre CEM avec C13, C14 et la self en mode commun L1 pour supprimer

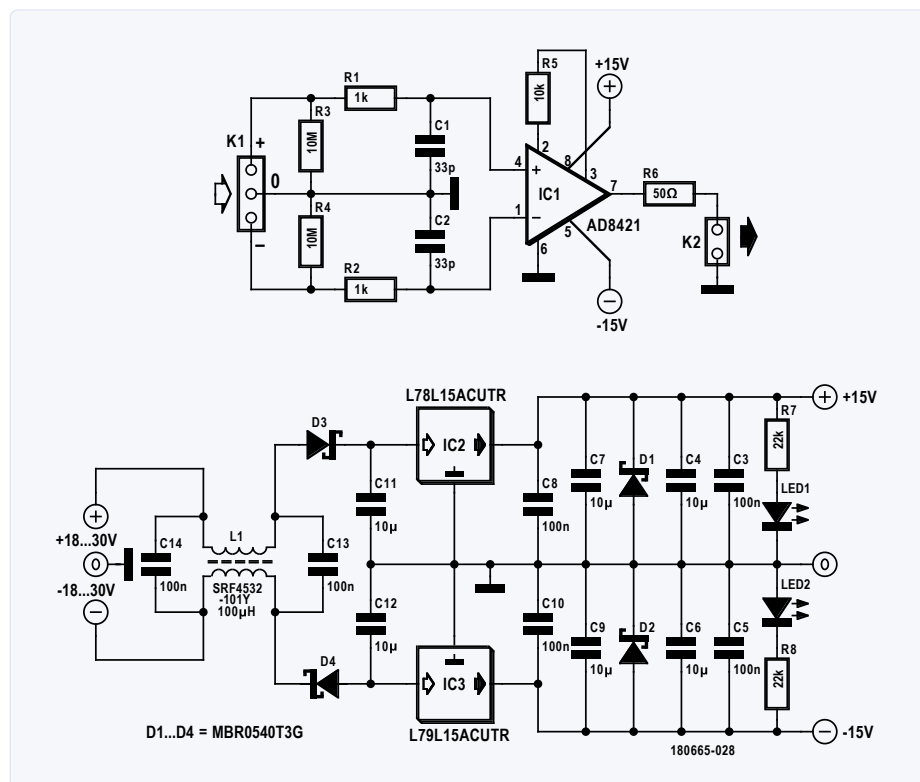


Figure 2. Circuit complet de la sonde 2.0. Ici on ne lésine pas sur les moyens pour obtenir des tensions d'alimentation propres !

les éventuels parasites de l'alimentation utilisée pour la sonde - ce qui est important si la sonde doit être alimentée par une alimentation à découpage. Pour un usage occasionnel, elle peut être alimentée par une alimentation électrique symétrique de laboratoire ou mieux, par une petite alimentation électrique propre. D'ailleurs, D3 et D4 ne sont pas destinés à redresser une tension alternative, mais à protéger contre une inversion de polarité ! Les LED1 et LED2 indiquent que les deux tensions d'alimentation sont présentes et que la sonde est active. Pour éviter de vous éblouir, leurs résistances série R7 et R8 ont une valeur élevée. Les deux résistances R3 et R4 de $10 \text{ M}\Omega$ servent à mettre les entrées ouvertes au potentiel de la masse, ce qui garantit que la sortie de la puce fournit alors 0 V . Il en résulte également une impédance d'entrée élevée et définie. La résistance R6 de 50Ω est nécessaire pour l'adaptation à un câble coaxial, car la sortie de IC1 est à faible impédance.

Alimentation

Comme il ne circule que de très faibles courants, un transformateur de $1,5 \text{ VA}$ avec $2 \times 12 \text{ V}$ ou $2 \times 15 \text{ V}$ au secondaire, un redresseur double alternance et deux condensateurs de filtrage de $22 \mu\text{F} / 35 \text{ V}$ chacun sont

CARACTÉRISTIQUES TECHNIQUES DE L'AD8421

Alimentation :

$\pm 2,5 \dots \pm 18 \text{ V}$

Consommation de courant :

$2,3 \text{ mA max}$

Bruit :

max. $3,2 \text{ nV}/\sqrt{\text{Hz}}$ @ 1 kHz

max. $200 \text{ fA}/\sqrt{\text{Hz}}$ @ 1 kHz

Bande passante :

10 MHz @ gain = 1

2 MHz @ gain = 100

CMRR :

94 dB @ DC, gain = 1

80 dB @ 20 kHz , gain = 1

Vitesse de montée :

$35 \text{ V}/\mu\text{s}$

Dérive :

$0,2 \mu\text{V}/^\circ\text{C}$

Tension de décalage d'entrée :

$1 \text{ ppm}/^\circ\text{C}$ de l'amplification @ facteur 1

Protection d'entrée :

max. 40 V entre « +IN » et « -VS »

max. 40 V entre « -IN » et « +VS »

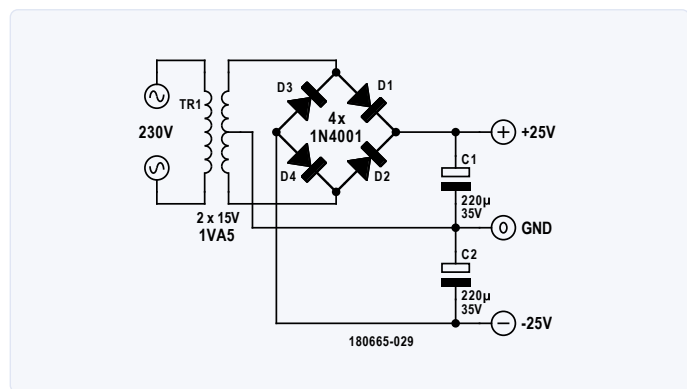


Figure 3. Circuit d'une simple alimentation électrique comprenant un transformateur, un redressement double alternance et des condensateurs de filtrage.

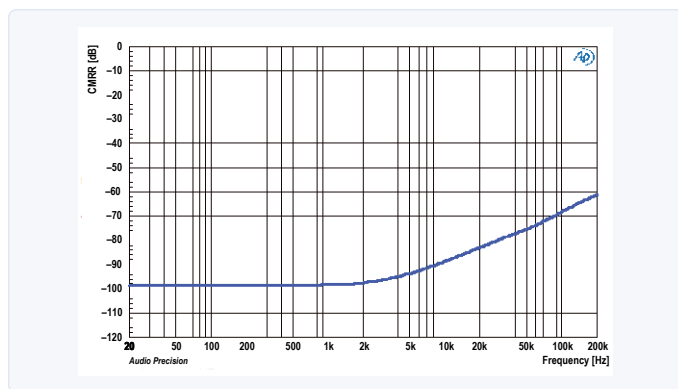


Figure 4. Le rejet de mode commun en fonction de la fréquence.

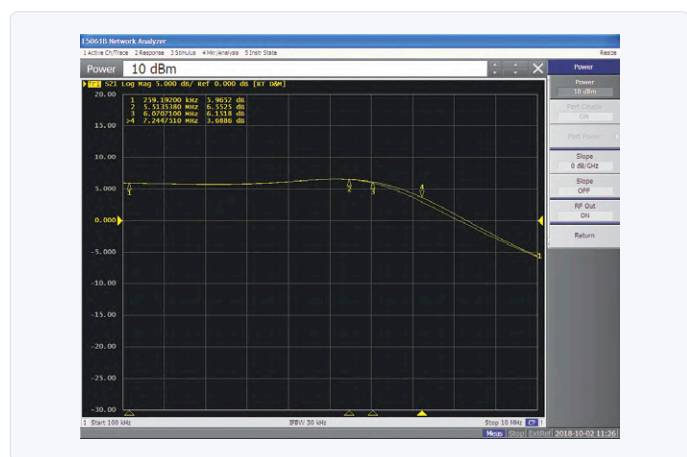


Figure 5. La réponse en fréquence de la sonde atteint 8 MHz.

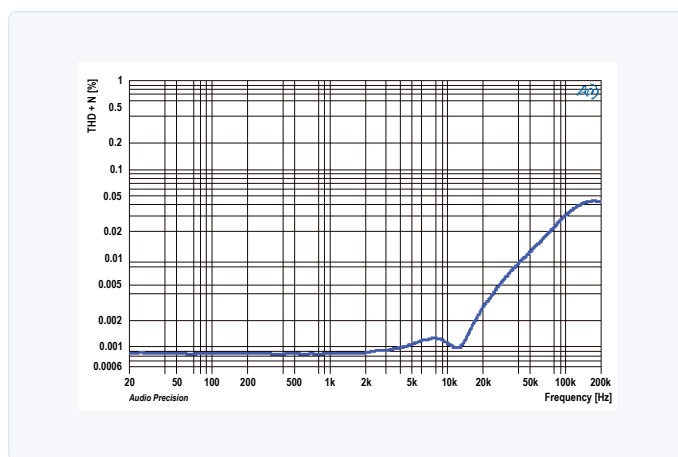


Figure 6. La distorsion augmente fortement au-dessus de 10 kHz, mais reste à un niveau très faible à 200 kHz.

suffisants (**fig. 3**). Les mesures n'ont montré aucune différence dans l'alimentation avec une alimentation aussi petite par rapport à une alimentation de laboratoire. L'ondulation résiduelle des tensions d'alimentation est assez faible à environ 200 mV_{CC} en raison de la faible charge.

Les régulateurs de tension de 15 V utilisés peuvent supporter des tensions d'entrée de 17,5 à 35 V. À travers les deux diodes Schottky D3 et D4, la sonde peut donc être alimentée avec des tensions de ± 18 V. Pour TR1, un petit transformateur de 12 V de tension nominale peut suffire, car les petits transformateurs de faible puissance sont à haute impédance et leur tension à vide ou à faible charge est nettement plus élevée. Même après avoir déduit la perte de tension de $\pm 0,7$ V à travers D1 à D4, on a souvent jusqu'à ± 20 V à la sortie de l'alimentation. Il vaut mieux le vérifier par une mesure. Par contre, avec un petit transformateur de 18 V on peut, dans certaines

conditions, se retrouver avec plus de ± 35 V à la sortie, ce qu'il faut absolument éviter. Avec un petit transformateur de 15 V, attendez-vous à environ ± 26 V à la sortie.

Des tests ont été effectués pour déterminer le comportement avec une tension d'alimentation réduite. Un signal d'entrée de 1 kHz avec une amplitude de 3 V_{CC} a été utilisé, ce qui, avec le gain de 2, a donné 6 V_{CC} à la sortie. La petite alimentation était connectée à un transformateur variable et sa tension d'entrée a été progressivement réduite. Tant que la tension d'alimentation fournie par la source d'alimentation restait supérieure à ± 12 V, on n'a observé aucun changement, mais en dessous de cette valeur, le signal à la sortie de la sonde commençait à faiblir. Rien d'étonnant, car avec les 2 V de chute de tension à travers D3 et D4 et les régulateurs, il ne reste guère de marge sur la sortie de IC1. Le fait que même à juste un peu plus de ± 12 V, il n'y ait pas eu de perturbation, alors que les

deux régulateurs de tension avaient cessé de réguler, prouve l'excellente suppression des effets de la tension d'alimentation de l'AD8421 (≥ 100 dB @ 100 Hz).

Mesures

Pour un signal de 1 kHz avec 3 V_{CC} aux deux entrées, la réjection de mode commun est supérieure à 98 dB. À partir de 2 kHz, elle diminue régulièrement jusqu'à 61 dB à 200 kHz (**fig. 4**). C'est mieux que prévu, bien que les composants du filtre d'entrée constitué de R1/C2 et R2/C2 ne soient pas identiques malgré une tolérance étroite.

Une mesure asymétrique avec un générateur de fonctions sur une entrée (l'autre étant mise à la masse) a donné une largeur de bande réelle d'environ 8 MHz (par rapport à 1 kHz). Lorsque le gain est réglé à un niveau bas avec R5, on observe une légère augmentation du gain près de la fréquence de coupure, la valeur maximale étant de 7 MHz (**fig. 5**).



Figure 7. Câble coaxial RG316 avec extrémité dénudée et connecteur serti à l'autre extrémité.

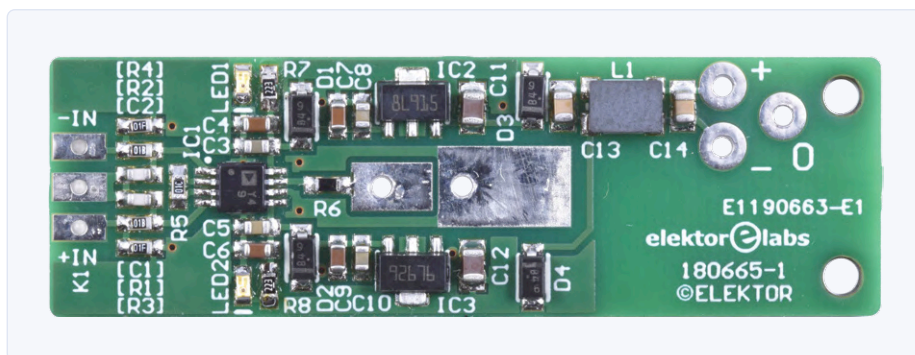


Figure 8. La carte du prototype entièrement équipée. Pour souder les CMS 0603, il faut une main ferme.

Les filtres d'entrée ont une fréquence de coupure de 4,8 MHz (sans tenir compte de l'impédance de la source), ce qui compense cette augmentation responsable du fait que la largeur de bande de la sonde est plus élevée que prévu. Cependant, à 8 MHz, le déphasage du signal de sortie est supérieur à 180° . Les ondes sinusoïdales dans la gamme des MHz sont plutôt triangulaires. La distorsion augmente progressivement pour les signaux supérieurs à 10 kHz (fig. 6).

Câbles

Un câble coaxial RG316, disponible au mètre, peut être utilisé pour connecter la sortie. Cependant, le raccordement d'un connecteur BNC à l'autre extrémité nécessite un outil à sertir adéquat. S'il n'est pas disponible, vous pouvez simplement acheter un câble coaxial BNC complet, couper un connecteur à une extrémité, le dénuder (fig. 7) et souder le câble sur la carte. Prenez un câble BNC long, coupez-le en son milieu, vous aurez deux câbles pour deux sondes.

Pour le soudage direct du câble RG316, la carte comporte en son milieu de grandes pastilles idoine (fig. 8), auxquelles le conducteur central et le blindage peuvent être soudés. Les fils pour l'alimentation électrique peuvent être soudés sur les trois pastilles en haut à droite. Les cinq plaquettes ont des trous qui accepteraient également des picots de 1,3 mm. Il faudra raccourcir un peu ces picots, afin que la carte tienne dans le coffret recommandé. Avec un petit serre-câbles passant par les trous de 2,5 mm tout à fait à droite, on peut fixer tous les fils sortant de la sonde (fig. 9). La colle chaude peut rendre de bons services ici. Toutefois, avant de brancher les câbles, veuillez lire d'abord la section sur les instructions de montage ci-dessous.

Comme «prise» pour les cordons de test, le plus simple est de souder, à plat sur les trois pastilles du côté gauche de la carte (fig. 8), une section de 3 pattes d'un support SIL, en raison de la qualité de ses contacts tournés. La fig. 9 montre l'aspect du montage terminé.

Instructions de montage

Un coffret approprié est spécifié dans la liste des composants. On commence par percer sur le côté gauche du coffret un trou rectangulaire approprié pour le connecteur SIL tripo-

laire, puis on introduit la carte entièrement assemblée et testée dans le coffret, on glisse la prise SIL tripolaire par la gauche à travers la fente sur la carte et on la soude. C'est maintenant au tour du câble coaxial. Utilisez de préférence du câble RG316. Le connecteur BNC doit être adapté au RG316 ! Il existe de nombreuses variantes de BNC pour différents câbles. Veuillez vous référer à la nomenclature. Le câble utilisé dans le prototype a un diamètre extérieur de 2,9 mm. Si c'est votre première prise BNC, vous devriez

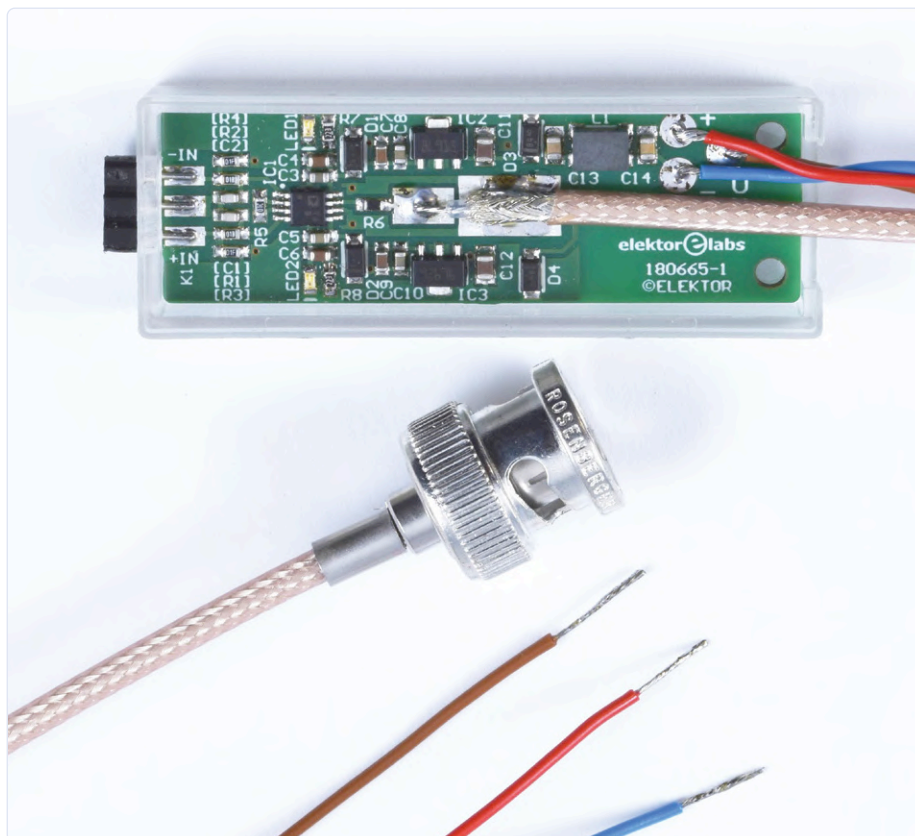


Figure 9. Prototype dans son coffret avec câbles soudés et prise SIL.



LISTE DES COMPOSANTS

Résistances :

(toutes 1%, 100 mW, SMD 0603)

R1,R2 = 1 k Ω

R3,R4 = 10 M Ω

R5 = 10 k Ω

R6 = 50 Ω

R7,R8 = 22 k Ω

Condensateurs :

C1,C2 = 33 p / 50 V, 1%, C0G/NP0, SMD 0603

C3,C5,C8,C10 = 100 n / 50 V, 10%, X7R,
SMD 0603

C4,C6,C7,C9 = 10 μ / 25 V, 20%, X5R,
SMD 0603

C11,C12 = 10 μ / 35 V, 10%, X5R, SMD 0805

C13,C14 = 100 n / 100 V, 10%, X7R, SMD 0805

Bobine :

L1 = 100 μ H @ 100 kHz, self de choc, 200 mA,
SMD, p. ex. SRF4532-101Y, Bourns

Semi-conducteurs :

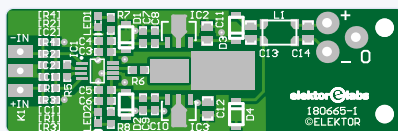
IC1 = AD8421ARMZ, MSOP 8

IC2 = L78L15ACUTR, SOT-89

IC3 = L79L15ACUTR, SOT-89

LED1,LED2 = LED, rouge, SMD 0603

D1..D4 = MBR0540T3G, SOD-123



Le côté composants du circuit imprimé
conçu au labo Elektor.

Divers :

K1 = Prise SIL 3 broches, verticale, RM 1/10

Prise SIL 3 broches comme fiche pour les
câbles de mesure*

Câble coaxial RG316, 50 Ω , \varnothing 2,9 mm, 1 m,
p. ex. Conrad 606450*

Connecteur à sertir BNC pour RG316, 50 Ω ,
p. ex. Conrad 1490723*

Boîtier USB1KL, 56x20x12 mm,
p. ex. Conrad 531276*

Autre boîtier : TC-USB1 KL203, 56x20x12 mm

Circuit imprimé n° 180665-1 v1.0*

* voir le texte

consulter le site du fabricant pour savoir
comment dénuder correctement le câble
avant de le fixer.

Pour souder le câble coaxial aux pastilles sur
la carte, il faut d'abord dénuder 12 mm de
la gaine. Ensuite, raccourcissez la tresse de
blindage à 6 mm. Enlevez maintenant 4 mm
de l'isolation du conducteur central. Le brin du
conducteur central se trouve alors à 2 mm du
blindage tressé. Soudez d'abord le conducteur
central, puis le blindage. Avec ce dernier, vous
devrez travailler rapidement. La fig. 9 montre
à quoi doit ressembler la connexion soudée.
Maintenant, c'est le tour des trois fils multi-
brins de l'alimentation. Il ne manque plus
que les conducteurs de mesure. Pour cela,
on soude trois brins fins à un connecteur
SIL à trois broches. Sur le prototype, nous
avons soudé deux fils fins de cuivre émaillé
à une embase SIL (fig. 10). Le couvercle peut
maintenant être mis en place et la sonde est
prête à l'emploi.

Mesures

L'alimentation de la sonde doit être isolée
galvaniquement du circuit testé – il est donc
préférable d'utiliser une petite alimentation
propre à la sonde comme celle de la fig. 3.
Lors de la mesure, il est impératif de relier la
masse de l'entrée à la masse du circuit ! Les
points de mesure peuvent avoir une tension de
mode commun jusqu'à ± 12 V par rapport
à la masse – mais pas plus.

Pour la connexion des fils de mesure : la
durée de vie mécanique des broches SIL
tournées est généralement de quelques
centaines d'embrochages. Elles devraient
donc pouvoir servir pendant des années. Si
nécessaire, la prise soudée à la carte peut
être facilement remplacée, car ses broches
ne sont pas enfichées dans des trous mais
soudées à plat sur les pastilles.

Au lieu d'utiliser la patte centrale de cette prise
comme masse, vous pouvez aussi simplement
connecter la pastille «0» de la carte à la masse

MESURES SUR LE PROTOTYPE

Tension d'alimentation avec le bloc d'alimentation selon la fig. 3 (transfo 12 V) : $\pm 18,9$ V.
La consommation par branche était de 5 à 6 mA. Les régulateurs de tension et les LED
se taillent la part du lion.

Valeurs maximales pour un signal de 1 kHz de 4,8 V_{CC} :

U_{out} : 9,52 V @ charge = 100 k Ω)

THD+N : 0,1%,

Valeurs typiques pour un signal de 1 kHz de 3 V_{CC} :

U_{out} : 5,97 V @ charge = entrée de l'oscilloscope

THD+N 0,00025 % @ B = 22 kHz

0,00083 % @ B > 500 kHz

Rejet du mode commun à 3 V_{CC} sur les deux entrées

CMRR : >98 dB @ 20..1.000 Hz

>60 dB @ 200 kHz

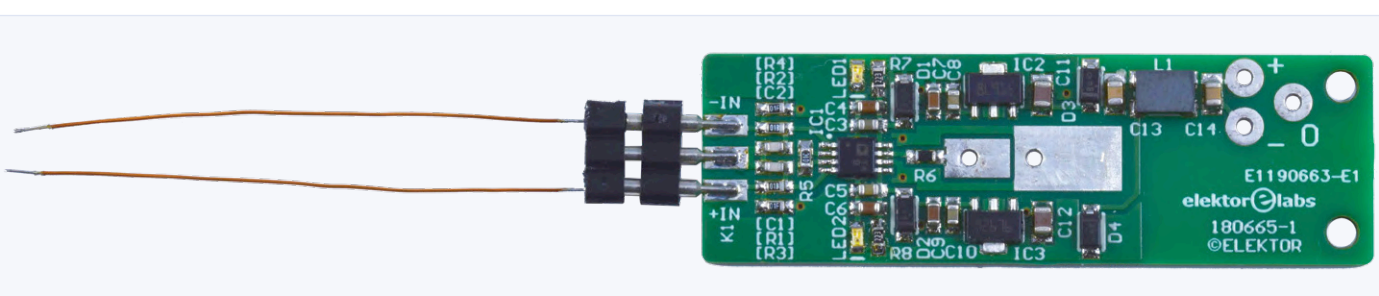


Figure 10. À titre d'essai, deux minces fils de cuivre émaillé, utilisés comme fils de mesure, ont été soudés à une prise SIL à trois pattes et embrochés. La masse doit alors être connectée ailleurs.

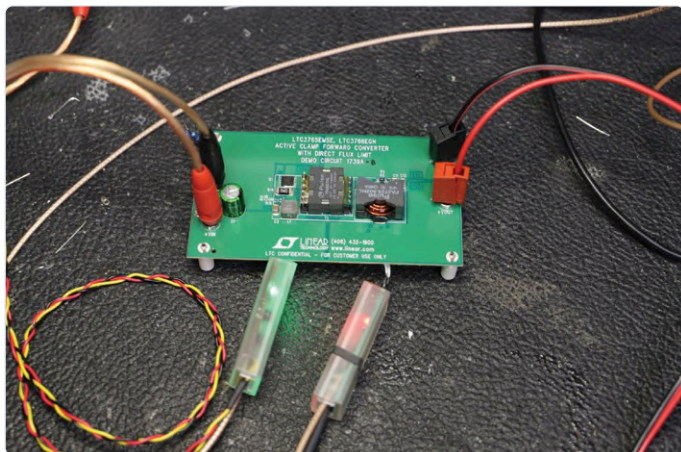


Figure 11. Alfred Rosenkränzer utilise deux sondes sur une carte de régulation à découpage.

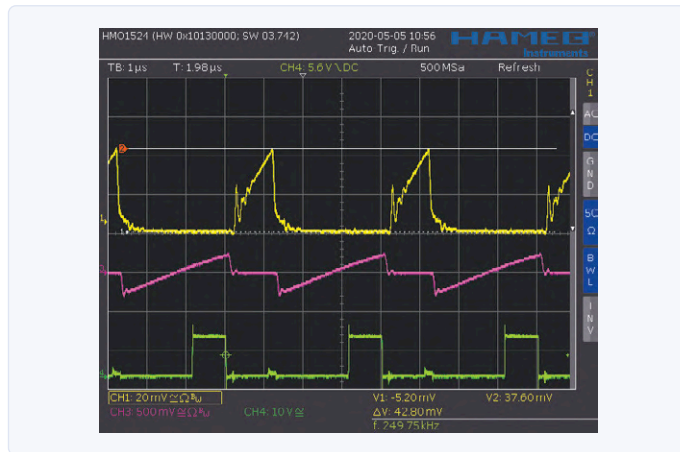


Figure 12. La sonde 2.0 peut également mesurer des variations rapides de courant.


du circuit à mesurer. Grâce à la forte réjection de mode commun de la sonde, il est peu probable que cela entraîne des interférences, du bruit ou des parasites.

Lorsque vous utilisez plusieurs sondes alimentées par la même alimentation, vous ne pouvez bien sûr établir qu'une connexion à la masse du circuit à mesurer – sinon vous créeriez une vilaine boucle de masse.

Important : si la tension d'alimentation du

circuit étudié est de plus de 12 V, avant de raccorder les câbles de mesure à une résistance shunt, il faut d'abord vérifier avec un multimètre si la tension à ses bornes reste dans la plage de mode commun de ± 12 V.

La **fig. 11** montre l'utilisation de deux sondes sur le circuit d'un petit régulateur à découpage. La **fig. 12** prouve que cette sonde peut mesurer des variations très rapides de l'intensité du courant.

Comme toujours, sur le site d'Elektor [3], vous pouvez télécharger gratuitement les fichiers de conception de la carte présentée dans cet article. 

180665-03

Votre avis, s'il vous plaît...

Vous pouvez vous adresser à l'auteur en allemand ou en anglais :

alfred_rosenkraenzer@gmx.de



Dans l'e-shoppe d'Elektor

- > **Pince de courant PeakTech 4350**
www.elektor.fr/peaktech-4350-clamp-meter
- > **Convertisseur de courant pour oscilloscopes (150170-1)**
www.elektor.fr/potentialfreie-strommessung-fur-oszilloskope
- > **Oscilloscope numéro à 2 voies OWON SDS1102 DSO (100 MHz)**
www.elektor.fr/owon-sds1102-2-ch-digital-oscilloscope-100-mhz

Ont contribué à cet article :

Idée et circuit : **Alfred Rosenkränzer**

Révision et texte : **Ton Giesberts**

Schémas : **Patrick Wielders**

Rédaction : **Thomas Scherer**

Traduction : **Helmut Müller**

Maquette : **Giel Dols**

LIENS

- [1] Article «Sonde de courant pour oscilloscopes», A. Rosenkränzer, Elektor juillet-août 2016 : www.elektormagazine.fr/magazine/elektor-201607/29098
- [2] Fiche technique de l'AD8421 : www.analog.com/media/en/technical-documentation/data-sheets/AD8421.pdf
- [3] Page de cet article : www.elektormagazine.fr/180665-03
- [4] Page de ce projet sur Elektor Labs : www.elektormagazine.fr/labs/differential-current-probe-180665

la boucle d'interface

Hewlett-Packard

Connecter le monde (en 1981) !



Blue Globe by Torty3, Wikimedia Commons under Creative Commons Attribution-Share Alike 3.0 Unported (https://commons.wikimedia.org/wiki/File:Blue_globe_icon.svg).

Karl-Ludwig Butte (Allemagne)

À l'ère de l'internet des objets, nos appareils sont petits, portables, alimentés par des piles et forcément interconnectés par un réseau, afin qu'ils puissent assurer une surveillance automatique et autonome de notre environnement grâce à leurs capteurs, et agir avec leurs actionneurs pour effectuer les réglages nécessaires. Cette idée n'est-elle apparue qu'après la diffusion par Mark Weiser de sa vision de l'informatique ubiquitaire dans un article très lu (et à relire!) de *Scientific American* [1] en septembre 1991? Non, dix ans avant, il existait déjà la *Hewlett-Packard Interface Loop* que je vous propose de (re)découvrir.

La caractéristique la plus remarquable de la calculatrice Hewlett-Packard HP-41C, lancée en juillet 1979 [2], était les quatre ports pour les périphériques tels qu'un lecteur de code à barres, une imprimante thermique, un lecteur de carte magnétique ou des modules d'extension de RAM et de ROM. Il n'existait rien de comparable. Les utilisateurs ont rapidement appris à l'apprécier et ont demandé de nombreuses autres options pour connecter le HP-41C à d'autres appareils.

Hewlett-Packard avait déjà acquis une expérience pertinente dans l'interconnexion et le pilotage de ses instruments de test et de mesure par le bus d'interface Hewlett-Packard (HP-IB), officialisé comme la norme industrielle IEEE-488. Cependant, le bus parallèle HP-IB de 8 bits, avec ses câbles épais à 18 conducteurs au moins et ses volumineuses fiches Amphenol à 24 broches, était difficilement compatible avec les petits appareils à batterie de la famille HP-41C. Pour concilier ces deux mondes, les ingénieurs de HP ont sérialisé le trafic de données. Au lieu de transmettre huit bits en parallèle, les bits ont été envoyés un par un, et l'acquittement (ou poignée de main) matériel a été remplacé par un acquittement logiciel. Le câble ne comptait plus que deux conducteurs.

Il restait cependant un détail à régler pour que le nouveau système d'interconnexion soit fiable et robuste comme les clients pouvaient l'attendre de la qualité légendaire de Hewlett-Packard.

Les ingénieurs de HP ont opté pour une topologie en boucle plutôt que pour une topologie en bus. Cela signifiait que le câble allait du pilote de boucle (par exemple un HP-41C) au premier périphérique, de là au périphérique suivant, et ainsi de suite jusqu'au dernier, après quoi il revenait au pilote de boucle. Cela permettait au contrôleur de boucle de vérifier directement si le nombre de bits reçus correspondait exactement au nombre de bits transmis, de sorte que toute erreur de transmission pouvait être corrigée immédiatement.

Le 14 décembre 1981 [3], Hewlett-Packard lançait officiellement la boucle d'interface Hewlett-Packard (HP-IL) pour le HP-41C sous la forme du module HP-IL 82160A (fig. 1). Il se branche simplement sur l'un des quatre ports du HP-41C. Le lecteur de cassette numérique HP-82161A HP-IL et l'imprimante thermique HP-82162A HP-IL (fig. 2) ont été lancés en même temps que le module HP-IL, dépassant les attentes les plus optimistes des clients. Ce système ingénieux a provoqué un choc et une crainte chez les concurrents de HP. Le nombre de périphériques a augmenté au fil du temps, ainsi que le nombre d'appareils pour lesquels un module HP-IL était disponible. Par exemple, la calculatrice HP-71B BASIC [4] et le HP-75C/D, en prenant en charge la boucle d'interface HP, pouvaient utiliser les mêmes périphériques que le HP-41C. Des cartes d'interface étaient disponibles pour les appareils de la série 80 (HP-85, 86 et 87) ainsi que pour les HP-110 et HP-150. En 1986, l'interface HP-IL HP-82973A est lancée sous forme de carte ISA pour les PC IBM et compatibles, et le kit de développement HP-IL HP-82166C permet à d'autres sociétés de concevoir leurs propres appareils compatibles avec HP-IL.

Les périphériques de HP

Le lecteur de cassettes n'était pas un simple enregistreur de cassettes audio standard comme celui de nombreux fabricants d'ordinateurs. HP a conçu un lecteur de précision parfaitement adapté au système HP-41C en termes de fonction, de taille et de conception du boîtier (fig. 3). Les cassettes étaient les mêmes que celles des mini-dictaphones. La figure 4 compare une mini-cassette à une cassette audio compacte standard. À gauche du compartiment à cassette sur le lecteur, un espace permet de ranger deux cassettes (fig. 3). La capacité de la cassette était de 131.072 octets, soit à peu près celle d'une disquette de 5¼ pouces sur un ordinateur Commodore PET 2001 [5], qui contenait 176.640 octets.

L'imprimante thermique HP-82162A ressemblait à son cousin plus ancien HP-82143A, qui pouvait être connecté directement à l'un des quatre ports d'entrée/sortie du HP-41C. L'imprimante thermique HP-IL, cependant, offrait une fonction attendue par les utilisateurs : l'impression des codes-barres HP-41C. Le lecteur de codes-barres HP-82153A, connectable directement à l'un des quatre ports d'entrée/sortie du HP-41C, permettait de lire le code en couverture des livres et des magazines beaucoup plus rapidement et plus sûrement qu'une fastidieuse saisie manuelle.

Ce n'était encore qu'un début. Le tableau 1 donne un aperçu des appareils compatibles HP-IL. Tout ce qu'un amoureux de HP peut désirer était disponible : stockage de données, imprimantes, interfaces, et même des coupleurs acoustiques. Cependant, la cerise sur le gâteau, c'était le parc des instruments de test et de mesure. Un système composé d'un multimètre HP-3468, d'un module de temps HP-82182A et d'un HP-41C, permettait d'automatiser entièrement les séances de mesure prolongées. L'adjonction à la boucle d'une



Figure 1. Le module HP-82160A HP Interface Loop.



Figure 2. Une installation HP-IL composée d'une HP-41C, d'un lecteur de cassettes numériques et d'une imprimante



Figure 3. Le lecteur de cassettes numériques HP-82161A.



Figure 4. Une cassette pour le lecteur numérique par rapport à une cassette audio standard.

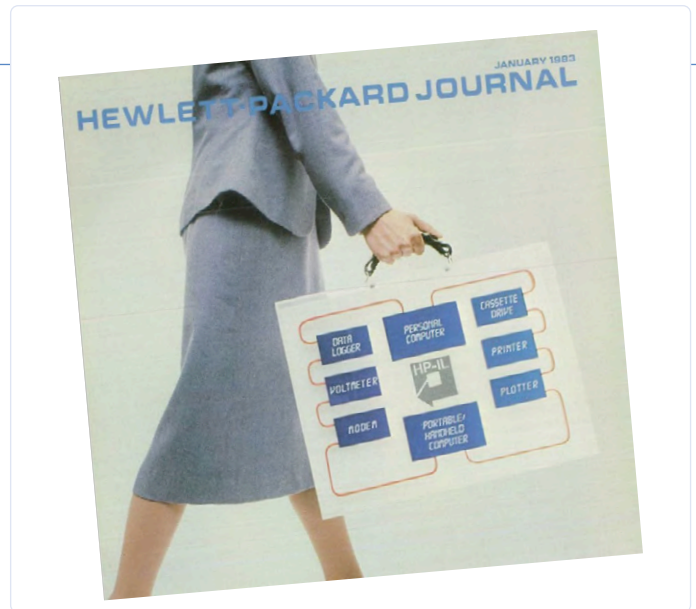


Figure 5. Illustration en couverture du Hewlett-Packard Journal, en janvier 1983.

Tableau 1. Périphériques compatibles HP-IL [6].

périphérique	fonction
HP 9114A/B	lecteur de disquettes
HP 82161A	lecteur de cassettes
HP 82162A	imprimante thermique (version HP-IL de l'imprimante HP 82143A pour HP-41C)
HP 82163A	interface vidéo
HP 92198A	interface vidéo à 80 lignes (Mountain Computer)
HP 82168A, 92205M	coupleur acoustique (Modems)
HP 82905A/B	imprimante
HP 2225B	imprimante ThinkJet
HP 7470A	table traçante graphique (interface HP-IL Opt. 003)
HP 2671A/G	imprimante thermique alphanumérique et graphique
HP 1630 et HP1631	analyseur logique (et contrôleur de boucle)
HP 3421A	saisie de données/unité de commande
HP 3468A/B	multimètre numérique (version HP-IL du HP 3478A avec GPIB)
HP 5384A et HP 5385A	fréquence-mètre avec Option 003 (HP-IL)
HP 82164A	interface série HP-IL-/RS-232C
HP 82165A	interface HP-IL-/GPIO-Schnittstelle (sortie parallèle polyvalente)
HP 82166A	convertisseur HP-IL (version compacte de l'interface GPIO pour applications embarquées)
HP 82166C	kit de prototypage de convertisseur HP-IL
HP 82169A	interface HP-IL-/HPIB (bidirectionnelle)
HP 82985A	interface HP-IL/NTSC (monochrome)

imprimante et/ou d'un lecteur de cassettes permettait d'enregistrer et de stocker les données de mesure. Avec le coupleur acoustique HP-82168A, ces données pouvaient être envoyées par téléphone à un ordinateur central pour un traitement ultérieur. D'un point de vue fonctionnel, nos applications IdO actuelles doivent beaucoup à la boucle d'interface HP.

Mode de fonctionnement

La boucle d'interface Hewlett-Packard applique une topologie en anneau comparable à celle de l'anneau à jeton conçue à l'université de Cambridge (Royaume-Uni) en 1974, puis améliorée et commercialisée par IBM au milieu des années 1980. La configuration en anneau de ces réseaux est très bien illustrée sur la couverture de janvier 1983 du Hewlett-Packard Journal (fig. 5), dans lequel la boucle d'interface a été décrite en détail.

Chaque système HL-IL était géré par un pilote de boucle et pouvait interconnecter jusqu'à 30 appareils. Chaque appareil se voyait attribuer automatiquement une adresse par le contrôleur de boucle, comparable à l'attribution automatique d'une adresse IP par DHCP dans un réseau Ethernet. Le contrôleur assurait l'application stricte du protocole, assisté par un ensemble de groupes de commande. Le pilote de boucle pouvait émettre des messages d'initialisation pour la boucle ou pour des appareils individuels afin de garantir des conditions initiales clairement définies. Un autre groupe de messages permettait au contrôleur de boucle de déterminer le locuteur ou l'auditeur actif, permettant ainsi aux appareils de communiquer directement entre eux. Les groupes de commande comprenaient également des requêtes d'état, des messages d'échange de données et d'autres messages pour l'administration et la synchronisation. La réponse d'un appareil particulier à un message spécifique pouvait varier considérablement et était déterminée par le concepteur de l'appareil périphérique. En réponse à un message de libération du périphérique (DCL), par exemple, l'imprimante thermique vidait son tampon d'impression interne, garait sa tête d'impression et faisait avancer le papier. L'interface



Figure 6. Le kit d'interface HP-IL HP-82166C.



Figure 7. Les composants spéciaux du kit d'interface HP-IL.

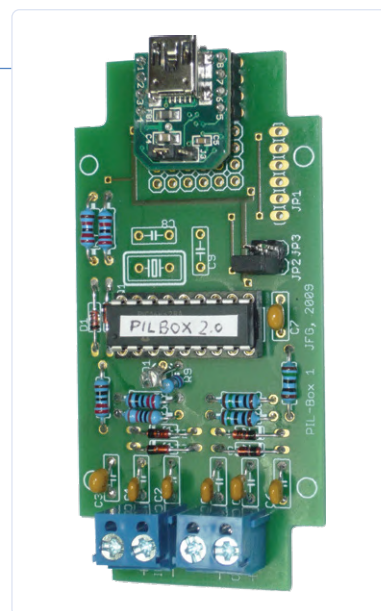


Figure 8. L'interface de la PIL Box (avec l'aimable autorisation de J.-F. Garnier, le détenteur des droits d'auteur).

vidéo réagirait à la même commande en vidant sa mémoire d'écran et en plaçant le curseur dans le coin supérieur gauche de l'écran.

Le kit d'interface HP-IL

HP a fortement encouragé la communauté des concepteurs de matériel à développer des appareils dotés d'une interface HP-IL et a publié des descriptions techniques détaillées à cet effet, de même qu'un kit d'interface (fig. 6). Les composants spéciaux du kit sont sur la fig. 7 : outre trois ASIC, le kit contenait deux modules de connexion pour les fiches spéciales du câble de connexion en boucle, de petits transformateurs DIL pour l'isolation galvanique des appareils, et un module enfichable pour le HP-41C pour le débogage des logiciels.

HP-IL n'est pas morte !

Il est difficile de retrouver la date de l'annonce de la disparition de la boucle d'interface HP. On sait que le HP-42S, lancé en 1988 pour succéder au HP-41C, n'avait pas de ports d'entrée/sortie, et encore moins d'interface HP-IL. Cependant, la flexibilité, la polyvalence

et la supériorité technique de la boucle d'interface HP ont rapidement conquis le cœur des concepteurs du monde entier, et cet enthousiasme reste intact.

Il existe un certain nombre de projets communautaires qui combinent l'interface HP-IL avec les techniques d'aujourd'hui. L'un des projets les plus connus est la PIL Box de Jean-François Garnier [7]. La fig. 8 en montre le circuit imprimé, qui remplit les fonctions d'une interface HL-IL vers USB. Les lecteurs de cet article peuvent facilement imaginer les possibilités nouvelles et inattendues que cela ouvre.

Comme souvent à cette époque, Hewlett-Packard était en avance sur l'état de l'art avec la boucle d'interface HP. Grâce à l'attention particulière accordée au développement, à la qualité de la production et à la documentation détaillée, nous pouvons encore nous amuser beaucoup avec cette technologie bien des années après l'abandon des produits associés. ◀

200329-03

LIENS

- [1] à lire : Weiser, Mark: « The Computer for the 21st Century ». Scientific American, Sept. 1991 : www.lri.fr/~mbl/Stanford/CS477/papers/Weiser-SciAm.pdf
- [2] calculatrice Hewlett-Packard HP-41C : www.finseth.com/hpdata/hp41c.php
- [3] calculatrices HP avec dates de mise sur le marché : www.vcalc.net/hp-date.htm
- [4] article « calculatrice Hewlett Packard 71B (1984) » par Karl-Ludwig Butte, dans Elektor 7/2014 p. 126 : www.elektormagazine.fr/magazine/elektor-201407/26964
- [5] article « PET fête ses 40 ans, bon anniversaire ! » par Karl-Ludwig Butte, dans Elektor 5/2017 p. 90 : www.elektormagazine.fr/magazine/elektor-201705/40364
- [6] HP-IL sur Wikipedia : https://de.wikipedia.org/wiki/Hewlett-Packard_Interface_Loop
- [7] page HP-IL de J.-F. Garnier : <http://www.jeffcalc.hp41.eu/hpil/>

apprentissage automatique : un bel avenir

Entretien avec Daniel Situnayake

C. J. Abate

Daniel Situnayake est ingénieur et ingénieur. Selon ses mots, l'apprentissage automatique (*machine learning* en anglais, ML) est une "technique unique en son genre". Il nous parle ici du potentiel du ML et présente TinyML, ainsi que quelques applications idéales. Nous évoquons aussi son expérience de développeur chez Google.



Figure 1. Daniel Situnayake (Ingénieur, fondateur de TinyML, Edge Impulse).

Quitter l'université

Abate : Commençons par tes antécédents. Quand t'es-tu intéressé pour la première fois au ML intégré ? Était-ce à l'université de Birmingham City (BCU), où tu as obtenu ta licence en réseaux informatiques et sécurité, ou après ?

Situnayake : Je n'ai pas vraiment cherché à travailler sur l'apprentissage automatique. Ce fut un itinéraire long et sinueux, avec des expériences apparemment disparates qui m'ont poussé ! Après mon diplôme de la BCU, on m'a proposé de rester comme enseignant et consultant. Je me suis concentré sur l'identification automatique et la capture de données, c'est-à-dire toutes les façons dont un ordinateur peut reconnaître et traiter des objets physiques. Ça va de la biométrie (reconnaissance faciale) à la RFID, en passant par les codes à barres et les cartes à puce. Je découvrais les algorithmes de vision par ordinateur, qui à l'époque n'étaient pas basés sur l'apprentissage automatique, et les systèmes embarqués. Quelques emplois plus tard, suite à une acquisition, je travaillais dans l'équipe naissante de *sciences des données* d'une société de banque généraliste. C'est là que je suis entré en contact avec les outils et les procédés de la science des données, et que j'ai appris à travailler avec de grands ensembles de données, sujet nouveau et fascinant pour moi. Avec la création de *Tiny Farms*, j'ai eu l'occasion de réunir ces

domaines, en utilisant des outils de science des données pour interpréter et réagir aux données issues de capteurs intégrés dans un contexte de technologie agricole. Mais c'est chez Google, en rencontrant Pete Warden (**fig. 2**) et Andy Selle de l'équipe *TensorFlow Lite*, que j'ai vraiment saisi le potentiel d'un apprentissage approfondi sur de tout petits appareils. C'est alors devenu mon objectif principal.

Abate : Quand as-tu déménagé en Californie ? Était-ce par opportunisme professionnel ?

Situnayake : En 2009, un an après l'implosion de l'économie mondiale. Au fil des dernières années, j'avais passé une batterie de tests de sélection comme élève officier de l'armée britannique. Je voulais piloter des hélicoptères Apache ! Avant la formation d'officier prévue pour mai 2009, je suis parti en voyage d'adieu à travers les États-Unis.

En Californie, j'ai rencontré une fille. Quelques mois plus tard, et quelques semaines après ma prise de service, j'ai compris que j'étais moins fait pour la vie de gentleman pilote que pour une vie décontractée en Californie. J'ai réservé un vol, j'ai emménagé avec la fille que je venais de rencontrer, et nous nous sommes mariés six mois plus tard !

Elle a repris ses valises, je suis resté. Après avoir quitté l'université, j'ai rejoint une jeune pousse spécialisée dans l'IA à Los Angeles, puis je suis arrivé dans la région de la Baie juste à point pour le boom technologique des années 2010. Un trip de dingue.

Élevage d'insectes

Abate : Tu es cofondateur de Tiny Farms, que tu décris sur LinkedIn comme "la première entreprise américaine de technologie d'élevage d'insectes". Comment a-t-elle vu le jour ? Où en est-elle ?

Situnayake : Comme toutes mes meilleures histoires, celle-ci commence encore par un voyage à travers les États-Unis. Deux de mes amis proches s'apprêtaient à lancer un projet de sécurité alimentaire. Au cours d'un voyage, ils ont séjourné dans une cabane au bord d'un lac, entourés de grosses sauterelles bruyantes dont ils se sont demandé, par jeu, si elles étaient comestibles. Quelques recherches leur ont appris l'importance cruciale des insectes comestibles pour la sécurité alimentaire mondiale - mais que les techniques d'élevage en captivité étaient mal connues.

Ils m'ont proposé de me joindre à eux pour créer une entreprise. J'ai vu le potentiel de l'application à l'automatisation de l'élevage d'insectes des techniques que j'avais apprises autour de l'identification automatique et de la capture de données. Mes connaissances en science des données me guideraient... J'ai fait le saut ! D'abord à temps partiel, puis nous avons réuni des fonds pour construire une ferme d'élevage d'insectes à l'échelle industrielle à San Leandro, en Californie. C'était intense, excitant et plus que stressant d'essayer de



Figure 2. Daniel Situnayake et Pete Warden signent leur livre TinyML.

Google a une étonnante culture de la liberté : tu travailles sur pratiquement tout ce que tu veux, à condition que cela ait un impact sur ton produit. Si tu as une idée géniale, tu es encouragé à la poursuivre et à faire appel à d'autres pour t'aider. Ainsi naissent de nombreux projets passionnants, mais cela peut devenir écrasant : devant un choix aussi vaste, sur quoi faut-il se concentrer ?



Des modèles TinyML pour détecter les éléphants sauvages, afin de prévenir les villageois que des pachydermes se dirigent vers eux.



maintenir en vie des millions de criquets tout en traitant avec les clients, les investisseurs et la recherche. Cinq ans, j'ai jeté l'éponge. Mes cofondateurs ont continué pendant quelques années, pour renoncer finalement devant la difficulté d'obtenir un produit viable. Tiny Farms continue de fonctionner comme dépôt libre des études et des données accumulées pendant sept ans d'activité [1].

Travailler pour Google

Abate : Tu as passé presque trois ans avec Google. Avant d'entrer dans les détails de tes fonctions, peux-tu nous dire ce que ça t'a fait de rejoindre une entreprise aussi impressionnante ?

Situnayake : Après cinq ans dans une entreprise agricole – où j'ai passé plus de temps dans la construction que dans le codage – je voulais me remettre dans le bain. Or, il n'y avait pas d'entreprise que j'admirais plus, techniquement en tout cas, que Google. Dès les premiers jours, j'ai été époustouflé. Je n'avais jamais travaillé dans une grande entreprise de technologie auparavant. Avec ton badge Google, tu accèdes aux installations de n'importe quel bureau Google, partout dans le monde. J'étais devenu citoyen d'un pays inconnu, avec ses propres services publics, ses infrastructures et son gouvernement.

Là ça se corse, parce qu'il se passe tant de choses, il y a des réunions potentiellement interminables, des distractions et tant d'idées stimulantes. Une bonne partie de ta première année chez Google tu la passes en fait à développer ton propre filtre intérieur, pour ne pas te laisser submerger. Les génies qui travaillent chez Google, c'est un peu un mythe. Il y a grosso modo la même gamme de compétences chez Google qu'ailleurs dans l'industrie technologique. Je ne suis pas porté sur le culte des héros. C'est agréable de rencontrer les meilleurs dans ton domaine et de constater que ce sont des gens comme toi.

Abate : Tu as travaillé chez Google en tant qu'avocat des développeurs pour TensorFlow Lite. Qu'est-ce que TensorFlow Lite ? Et comment avez-vous formé les programmeurs, par des rencontres, des cours en ligne ?

Situnayake : TensorFlow est le nom de l'écosystème d'outils libres de Google pour la formation, l'évaluation et le déploiement de modèles d'apprentissage profond. Il comprend tout, du code Python de haut niveau utilisé pour définir des architectures de modèles jusqu'au code de bas niveau utilisé pour exécuter ces modèles sur différents types de processeurs.

TensorFlow Lite est le sous-ensemble des outils *TensorFlow* qui traitent du déploiement de modèles vers des “périphériques”, c'est-à-dire tout ce qui est plus petit qu'un ordinateur personnel : téléphones, microcontrôleurs, etc. Les outils *TensorFlow Lite* peuvent prendre un modèle d'apprentissage approfondi et l'optimiser pour ces types d'appareils. Il comprend également le code hautement optimisé qui s'exécute sur les appareils eux-mêmes.

Ce que je préfère dans *TensorFlow Lite*, c'est la variante dépouillée et super rapide conçue pour faire tourner des modèles d'apprentissage profond sur de minuscules μ C bon marché et de faible puissance. Avant le lancement de *TensorFlow Lite* pour μ C, les programmeurs devaient écrire leur propre code de bas niveau pour exécuter des modèles d'apprentissage sur des appareils embarqués. Le barrage d'entrée pour l'apprentissage machine sur les appareils était donc difficile à franchir. L'entrée de *TensorFlow* dans ce domaine a fait tomber beaucoup de murs, si bien que soudain, avec un peu d'expérience de l'embarqué, n'importe qui pouvait exécuter des modèles. C'est excitant de voir le domaine du *TinyML* prendre vie.

Mon rôle était d'aider l'équipe de *TensorFlow Lite* à comprendre l'écosystème des développeurs et à s'y connecter. Tous les programmeurs sont différents, il importe donc de créer de multiples façons pour eux de se lancer. Nous avons fait tout ce qui est imaginable, depuis les rencontres en personne jusqu'aux conférences financées par Google, en passant par la création d'exemples de code faciles à comprendre et la rédaction du livre *TinyML* !

Abate : Pourquoi as-tu quitté Google ? Fut-ce une décision difficile ?

Situnayake : En travaillant avec les concepteurs de *TensorFlow*, j'ai pu constater qu'il fallait un énorme effort d'éducation pour que même les meilleurs outils et bibliothèques d'apprentissage sur machine soient utilisés. Malgré l'excellence de certains cours et de leur contenu, le temps qu'il faut y consacrer met l'apprentissage automatique hors de portée d'un grand nombre d'ingénieurs trop occupés. Les modèles de formation sont autant un art qu'une science, et pour acquérir les réflexes et les bonnes pratiques, il faut des années. C'est encore plus difficile avec le *TinyML*, un domaine tellement nouveau et aux contraintes mal connues. Quand j'ai entendu parler d'*Edge Impulse*, la société pour laquelle je quitterai Google, j'ai été époustoufflé. Avant même le lancement officiel, les fondateurs avaient déjà un produit qui permettait à n'importe quel programmeur de former des modèles d'apprentissage profond sur du matériel embarqué. J'ai pu constater, grâce à mon travail avec la communauté des développeurs, qu'ils avaient résolu certains des problèmes les plus difficiles en éliminant les pierres d'achoppement qui rendent les outils d'apprentissage automatique si difficiles à utiliser. Je pense que l'apprentissage automatique est une technique unique en son genre. Elle va transformer le monde. *Edge Impulse* a pour mission de mettre cette technologie entre les mains de tout ingénieur, afin qu'il puisse utiliser ses diverses perspectives et sa connaissance approfondie du domaine pour résoudre des problèmes complexes où que ce soit dans le monde.

Conscient de ce potentiel, j'ai décidé de me lancer. J'étais triste de quitter mes amis de Google, mais le monde (de *TinyML*) est petit et je reste en collaboration étroite avec beaucoup d'entre eux !

Tiny Machine Learning (TinyML)

Abate : Que doivent savoir de *TinyML* les ingénieurs et les électroniciens sérieux ? Pourrais-tu donner des exemples d'application d'apprentissage automatique à très faible consommation ?

Situnayake : Notre monde est truffé de capteurs, dans toutes sortes d'endroits, des appareils ménagers aux usines intelligentes en passant par nos véhicules. Les données de milliards de capteurs représentent une occasion précieuse pour les concepteurs de créer des appareils utiles, qui comprennent le monde autour d'eux et prennent des décisions intelligentes.

Auparavant, la seule façon d'utiliser ces données était de les envoyer par l'internet à un ordinateur puissant capable de calculer, d'interpréter puis de déclencher des actions en fonction des résultats. Cela fonctionne pour certaines applications. Pas de problème p. ex, si les données d'une station météo mettent quelques secondes à arriver sur un serveur, mais il y a de nombreuses applications où cela ne fonctionne pas.

L'un de mes exemples favoris est celui des caméras animalières. Les chercheurs les utilisent pour observer les animaux, souvent dans des endroits éloignés dépourvus d'accès à l'internet. Un animal qui passe déclenche le capteur de mouvement, et une photo est prise. Les chercheurs passent périodiquement pour vider la carte mémoire remplie de photos.

Or, le détecteur de mouvement a pu être déclenché par des feuilles poussées par le vent, ou des espèces qui n'intéressent pas le chercheur. Il faut beaucoup de temps pour passer au crible toutes ces photos. La carte mémoire se remplit rapidement, il faut donc la vider fréquemment. Même avec une connexion internet, la forte consommation déchargerait les piles de l'appareil qu'il faudrait remplacer.

Grâce à un modèle *TinyML* entraîné à reconnaître le bon type d'animal, la caméra ne prendra que les bons sujets. Les caméras peuvent donc être laissées en place plus longtemps, tandis que les chercheurs font des économies de temps et d'argent.

Il y a quatre contraintes principales pour qu'une application soit idéale pour *TinyML*. Si une application a une connectivité limitée, une faible tolérance à la latence, une source d'alimentation limitée ou une exigence de confidentialité élevée, *TinyML* vous évitera l'envoi de données à partir de l'appareil.

Seuls certains problèmes peuvent être résolus par l'apprentissage automatique. Ce sont des problèmes pour lesquels un peu de flou est acceptable, car l'apprentissage automatique donne rarement des réponses absolues. Il est parfait pour prendre des photos d'animaux, où l'erreur occasionnelle est acceptable, mais il n'est pas assez fiable dans certaines applications critiques pour la sécurité.

Abate : Tu es co-auteur du livre, *TinyML : Machine Learning with TensorFlow Lite on Arduino and Ultra-Low-Power Microcontrollers*. Comme il couvre *Arduino*, cela me porte à croire que la technologie *TinyML* est accessible aux non-experts comme aux professionnels. Outre la lecture de ton livre, comment les ingénieurs et les innovateurs qui ne sont pas familiers avec *TinyML* peuvent-ils se lancer ?

Situnayake : Comme toujours, la meilleure façon d'apprendre est de construire soi-même ! Un bon point de départ est de suivre certains des tutoriels fournis dans la documentation d'*Edge Impulse* : la reconnaissance de mouvement avec un accéléromètre [2] ; la reconnaissance de sons à partir de l'audio [3] ; et l'ajout de la vue à vos capteurs [4].

Nous avons construit *Edge Impulse* pour que tu puisses utiliser votre téléphone portable pour collecter des données et tester des modèles, afin de commencer sans matériel spécial. Une fois familia-

risé avec le flux d'apprentissage de la machine, tu peux prendre ta carte de développement préférée et commencer à programmer ! Je recommande l'Arduino Nano 33 BLE Sense comme tremplin pour le TinyML. Cette carte possède un processeur Cortex M-4F rapide, mais de faible puissance, et suffisamment de RAM pour certains modèles intéressants. Elle est entièrement prise en charge dans Edge Impulse.

Impulse Edge

Abate : Tu as rejoint l'organisation Edge Impulse, basée à San José, en janvier 2020. Comment cela s'est-il fait ?

Situnayake : J'ai rencontré le PDG, Zach Shelby quand j'étais chez Google. Il était auparavant vice-président chez ARM, qui travaille en étroite collaboration avec l'équipe de TensorFlow Lite. Quand Zach a annoncé qu'il quittait ARM pour fonder une société TinyML, cela a attiré mon attention, et quand ils ont publié leur première offre d'emploi, j'ai postulé !

Abate : Comme se déroule ta journée de travail ? Codes-tu toute la journée ? Enseignes-tu ? Programmes-tu pour des entreprises ?

Situnayake : Ce que j'apprécie le plus à Edge Impulse c'est d'être redevenu principalement un ingénieur. Je passe la plupart de mon temps à programmer. J'ai toujours l'occasion de rencontrer des clients, ce qui est assez gratifiant et crucial pour savoir comment concentrer mon temps d'ingénieur. Je suis également très actif dans la communauté TinyML, c'est pourquoi je donne régulièrement des conférences et j'aide à organiser des rencontres TinyML. Cela dit, j'ai généralement deux ou trois réunions le matin, ce qui me permet de me concentrer et coder le reste de la journée. C'est génial !

Abate : Je vois que l'équipe d'Edge Impulse a commencé le projet COVID-19 en mai - "Détection de toux avec TinyML sur Arduino" (fig. 3). Le projet a été visionné plus de 5 400 fois. Quelle a été la réaction de la communauté ? Quel est le statut du projet ?

Situnayake : Même si Edge Impulse n'existe que depuis quelques mois, nous avons déjà une communauté fantastique venue d'horizons divers. Lorsque la COVID-19 a frappé, l'un des membres de notre communauté, Kartik Thakore, chercheur et ingénieur biomédical, était intéressé par la construction de quelques projets pour nous aider. Persuadés que ce serait une bonne occasion de montrer le potentiel de TinyML, nous lui avons donc envoyé du matériel et il a monté un détecteur de toux avec un modèle TinyML. Le résultat a été ce projet [5], qui, nous l'espérons, inspirera d'autres innovateurs à relever des défis similaires !

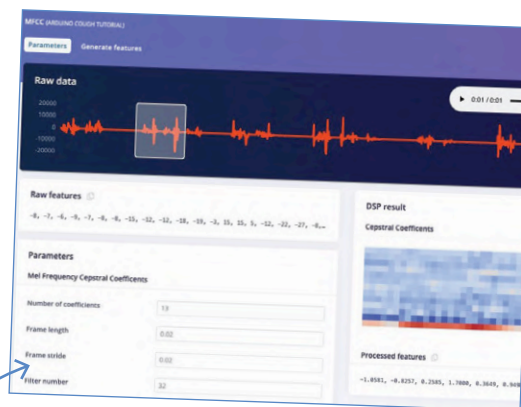


Figure 3. Détecteur de toux avec TinyML sur Arduino (Source : Edge Impulse).

Projets TinyML

Abate : Y a-t-il d'autres projets TinyML que tu souhaites mettre en avant pour Elektor ?

Situnayake : Chez Edge Impulse, nous croyons à la technique comme **force du bien**, et certains de mes projets préférés relèvent de cette catégorie. Les Smart Parks forment p. ex. des modèles TinyML pour détecter les éléphants sauvages, afin de prévenir les villageois que des pachydermes se dirigent vers eux.

Une autre de mes catégories favorites est celle des personnes qui font faire aux ordinateurs des choses inouïes. Comme le projet étonnant de Benjamin Cabé : un capteur de gaz reconnaît différentes marques de spiritueux ! S'il vous intéresse, il existe un tutoriel génial de Seed Studio. ❏

200423-03

Votre avis, s'il vous plaît...

Vous pouvez adresser vos questions et vos commentaires (en anglais) à la rédaction : editor@elektor.com

Ont contribué à cet article :

Entretien : **C. J. Abate**

Maquette : **Giel Dols**

Traduction : **Henry Philippe**

LIENS

- [1] **Open Tiny Farms**: <https://www.opentinyfarms.com/>
- [2] **Edge Impulse, "Continuous Motion Recognition"**: <https://docs.edgeimpulse.com/docs/continuous-motion-recognition>
- [3] **Edge Impulse, "Recognize Sounds from Audio"**: <https://docs.edgeimpulse.com/docs/audio-classification>
- [4] **Edge Impulse, "Adding Sight to Your Sensors"**: <https://docs.edgeimpulse.com/docs/image-classification>
- [5] **Edge Impulse, "Cough Detection with TinyML on Arduino"**: <http://bit.ly/edge-cough-detection-project>

applications mobiles

pour Android
et iOS

à partir
d'un moule unique



Veikko Krypczyk (Allemagne)

De plus en plus d'applications (p. ex. autour de l'internet des objets) requièrent un accès par les appareils mobiles, *Android* et *iOS*. Diverses approches se disputent les faveurs du programmeur qui peut difficilement se soustraire à la demande multi-plate-forme et devra offrir une interface intuitive. L'environnement de développement *RAD Studio* – plus connu sous le nom de *Delphi* – semble bien approprié à ces exigences.

De plus en plus d'applications pour l'électronique reposent sur une interaction raffinée entre matériel et logiciel. Les applications pour appareils mobiles tels que tablettes et téléphones tactiles avec les systèmes d'exploitation *Android* et *iOS* sont courantes. Si la conception d'applications mobiles n'est pas votre pain quotidien, la création d'une application propre aux deux systèmes, adaptée parfaitement aux exigences du projet risque d'être un croûton sur lequel vous vous casserez les dents. Le démarrage ne sera pas sans surprises. Les approches en concurrence ont en commun des courbes d'apprentissage abruptes. Ainsi la programmation des petits appareils se change en grosse affaire, au point de devenir une pierre d'achoppement pour

l'ensemble du projet. Un remède possible est une approche multi-plate-forme, résolument intuitive, de la conception de l'interface utilisateur.

Dans cet article, nous donnons un aperçu des possibilités de conception d'applications mobiles, avant d'examiner de plus près une approche intéressante. Avec *RAD Studio* – également connu par de nombreux praticiens sous le nom de *Delphi*, un outil efficace pour le développement d'applications de bureau – les applications pour *Android* et *iOS* peuvent désormais être créées à partir d'une base commune de code source, et d'un effort considérablement réduit. La programmation se fait avec *Delphi (Object Pascal)* et

le support d'un outil graphique approprié. Un exemple montrera comment cela fonctionne.

Applications pour les appareils mobiles

Du point de vue de la technique, on peut distinguer les types applications, natives, web et hybrides.

Les applications natives sont créées exclusivement pour *Android* ou *iOS*. Vous utilisez directement l'API des systèmes. L'interface utilisateur s'intègre au mieux dans la plate-forme. L'absence de restriction dans l'accès à d'éventuels matériels spécifiques du téléphone ou de la tablette est un avantage : il est permis de s'adresser directement à tous les capteurs

de ces appareils. L'application est disponible dans les **app stores**. Si une application native est installée, elle peut, si cela a un sens, être exécutée sans accès à l'internet (hors ligne). Une synchronisation des données nécessaire peut avoir lieu automatiquement lors de la prochaine connexion en ligne. Les applications pour iOS (*Apple*) sont créées avec *Xcode* (environnement de développement) et *Swift* (langage de programmation). Pour *Android*, on utilise *Android Studio* et *Kotlin* ou *Java*.

Les applications web, en revanche, sont destinées aux appareils mobiles. Cela concerne principalement la conception de l'interface utilisateur. L'accès au matériel du système est limité. Certaines fonctions de base, comme le positionnement par GPS, sont toutefois possibles. Les API *JavaScript* correspondantes sont utilisées à cet effet. Les applications ne peuvent pas être proposées dans les **stores**. Elles fonctionnent sur un serveur et imposent donc la connexion permanente à l'internet. Une icône peut être installée sur l'écran d'accueil afin d'accélérer le démarrage de l'application. Le manque de capacité hors ligne peut être partiellement éliminé grâce à la technologie des applications web progressives (PWA). Une PWA met en symbiose un site web et une application. Par le biais d'un travailleur de service, une fonction de mise en cache peut être mise en œuvre. Ce travailleur de service est connecté entre serveur web et application sur l'appareil mobile. Pour créer une application web pour systèmes mobiles, toute la gamme des technologies web est disponible. En particulier, les bibliothèques et les cadres les plus courants peuvent être utilisés. En fin de compte, ces applications reposent sur les éléments HTML (structure), CSS (conception) et *JavaScript* (interaction, logique), puisque le navigateur ne peut qu'interpréter ces langages.

Les applications hybrides sont basées en interne sur les technologies du web. L'application web est exécutée dans un navigateur intégré, c'est-à-dire qu'elle fonctionne dans une sorte de «bac à sable» qui la rend indépendante de la plate-forme. Le système croit donc avoir une application native (le navigateur) devant lui. Il existe plusieurs approches pour les applications hybrides, p. ex. Cordova de l'*Apache Software Foundation*. Le principe est toujours similaire. L'application ouvre une fenêtre de navigateur en mode plein écran au démarrage, de sorte que le navigateur n'est pas identifiable en tant que tel et que l'adresse web ne peut

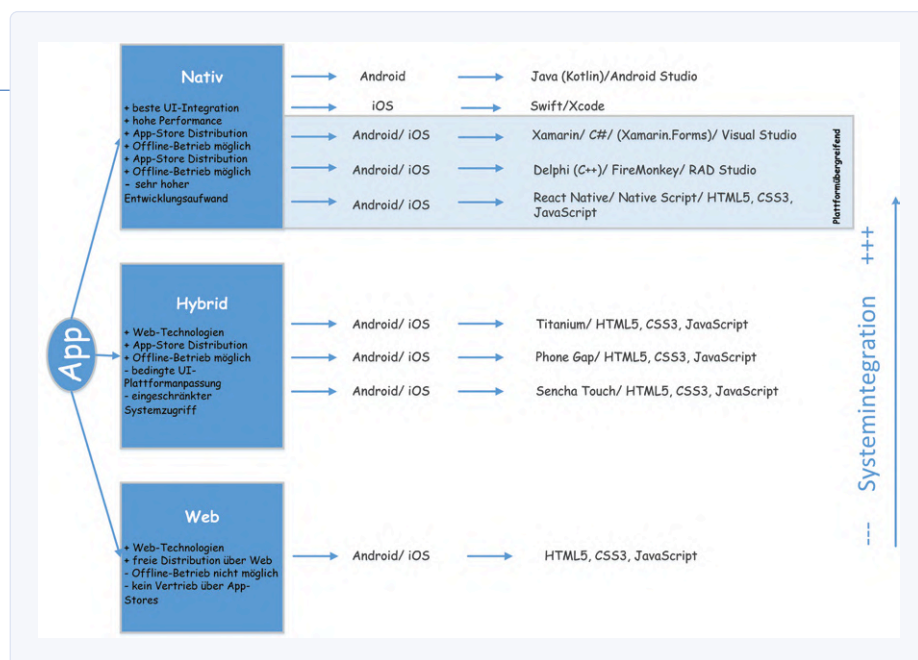


Figure 1. Types d'applications et moyens d'accès à l'application.

pas être modifiée. L'application créée avec HTML, CSS et *JavaScript* est affichée dans le navigateur. Le cadre donne à l'application l'accès aux fonctions du système telles que caméra ou carnet d'adresses. Cela se fait à l'aide de greffons (*plug-ins*).

Les applications web et les applications hybrides ciblent *Android* et iOS simultanément. Cependant, elles présentent des inconvénients tels que des performances médiocres et un accès matériel limité par rapport aux applications natives. C'est pourquoi on essaie de relier les deux types de produits entre eux par des **approches multi-plates-formes**. L'objectif est de se rapprocher le plus possible du modèle natif (également pour l'aspect et pour la convivialité de l'interface utilisateur) et de créer l'application pour *Android* et iOS à partir d'une base de code source commune. Il existe un grand nombre d'approches multi-plates-formes, très différentes les unes des autres, notamment *Xamarin*, *RAD Studio*, *NativeScript* et *React Native* (fig. 1).

Dans cet article, nous allons examiner de plus près la procédure avec *RAD Studio* (*Delphi*), bien adaptée, en particulier pour les projets électroniques et IoT. La procédure est familière à de nombreux praticiens. Avec l'aide d'un concepteur graphique, l'interface utilisateur peut être créée - comme pour le bureau de *Windows*. La logique est écrite dans le langage *Delphi* (*Object Pascal*), également intuitif et simple. Certains électroniciens connaîtront déjà *Pascal*, pour avoir p. ex. programmé des logiciels embarqués pour microcontrôleurs.

Installation et configuration du système

Nous installons maintenant *Delphi* sur *Windows* et le configurons pour le développement mobile. Téléchargez l'édition communautaire [1] après l'enregistrement et lancez l'installateur. Pendant l'installation, vous verrez une boîte de dialogue avec une option de sélection de la plate-forme (fig. 2).

RAD STUDIO, DELPHI ET C++-BUILDER

Delphi et *C++ Builder* sont les environnements intégrés d'*Embarcadero* pour la création d'applications pour différents systèmes d'exploitation. *Delphi* utilise le langage de programmation *Delphi* du même nom (une évolution d'Object Pascal), tandis que *C++-Builder* utilise le C++. Dans *RAD Studio*, les deux environnements sont combinés. Plusieurs éditions sont disponibles. Pour l'usage professionnel, il existe les produits commerciaux *Professionnel*, *Entreprise* et *Architecte*. Pour un usage commercial limité, pour tester, apprendre, pour des projets privés et open source, il existe l'édition communautaire gratuite. Nous l'utilisons également ici. La version actuelle est la 10.4 de *RAD Studio*.

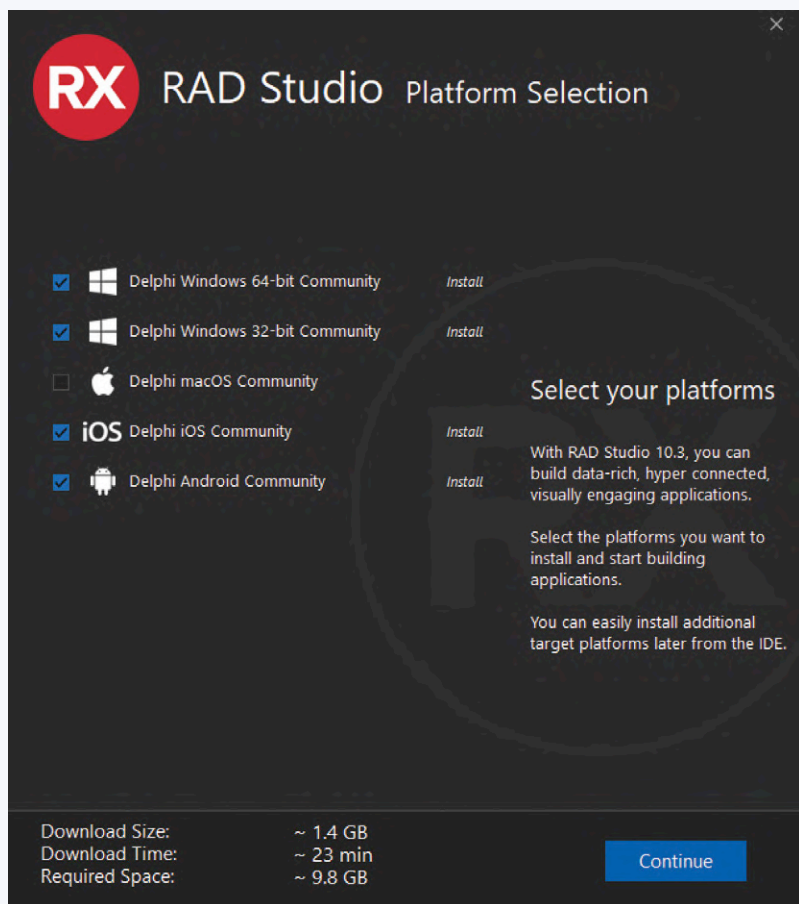


Figure 2. Sélection de la plate-forme.

En plus de *Windows*, sélectionnez *Android* et *iOS*. Ensuite, il vous sera demandé de sélectionner des options supplémentaires. Vous devriez faire cela pour le *SDK/NDK Android*. Après l'installation, ouvrez l'environnement de développement (fig. 3).

Pour le développement pour l'appareil mobile, des retouches sont encore nécessaires. Si vous voulez tester l'application sur iOS, vous devez avoir accès à un ordinateur *Apple* avec *Xcode* sur le réseau. Un simulateur pour iOS ne peut être exécuté que sous macOS. Même

un *iPhone* physique est directement couplé avec l'ordinateur Mac. Celui-ci est ensuite accessible sur le réseau à partir de *Windows* avec *RAD Studio* (fig. 4).

Pour ce faire, le logiciel *PAServer* doit être installé sur le Mac pour un accès à distance. Si vous travaillez sur un ordinateur Mac, vous pouvez exécuter *Windows* avec une machine virtuelle (*Parallels*, *Virtual Box*, etc.) et installer *RAD Studio* dans cette machine virtuelle à partir de laquelle on accède ensuite à l'ordinateur Mac, c'est-à-dire que tous les composants pour le développement sont installés sur un seul ordinateur. Le fait de devoir disposer d'un ordinateur Mac pour créer des applications pour iOS n'est pas propre à *RAD Studio*, mais s'applique à toutes les procédures.

Vous pouvez créer des applications pour *Android* directement à partir de *Windows*. Pour les tests, vous pouvez connecter un téléphone/tablette à votre ordinateur par l'USB ou (alternativement) installer un émulateur. Cependant, un «vrai» appareil est beaucoup plus rapide. Connectez votre téléphone à l'ordinateur avec le câble USB et configurez-le correctement, c'est-à-dire activez dans les paramètres les autorisations du développeur, y compris le débogage USB sur le téléphone. Vérifiez maintenant dans le gestionnaire de périphériques si votre appareil mobile a été correctement reconnu et configuré. La documentation officielle pour la mise en place de *Delphi* pour le développement mobile se trouve sous [3] et [4]. La configuration est terminée. Un premier test («*Hello World*») illustre la procédure de développement d'une application mobile.

« Hello Elektor »

Cette section concerne la création d'un premier projet. Dans l'environnement de développement *Delphi*, sélectionnez *File | New | Multi-Device Application - Delphi* dans le menu principal et cliquez sur *Blank Application* (fig. 5) dans la fenêtre, puis sur *OK*. Sauvegardez le projet. Grâce à la palette d'outils, nous pouvons créer l'interface utilisateur à l'aide de contrôles visuels. Pour ce faire, faites glisser un composant de type *TLabel* dans la boîte de dialogue pour un test. Vous pouvez changer l'aperçu pour les différents systèmes (*Windows*, *Android*, *iOS*).

Dans l'*inspecteur d'objets*, les propriétés des composants peuvent être configurées. Nous avons réglé la *propriété Text* sur la valeur

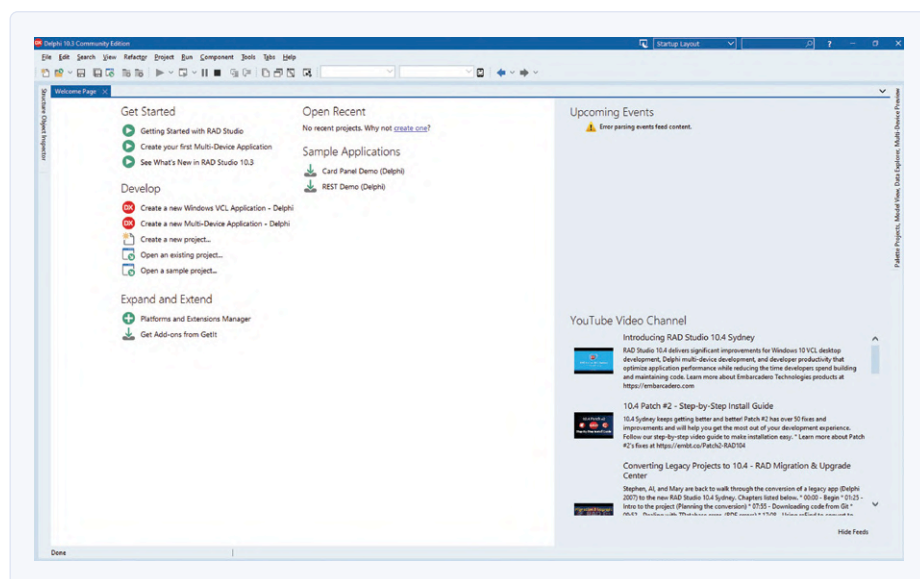


Figure 3. Écran de démarrage de l'environnement de développement intégré de Delphi.

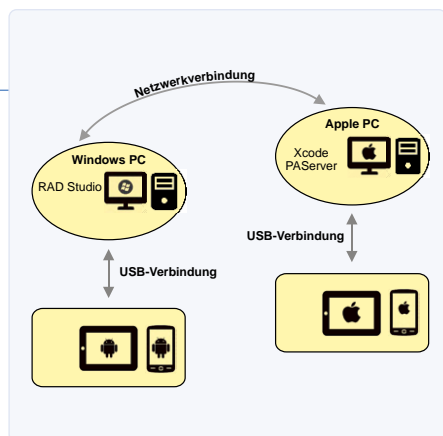


Figure 4. Configuration du système pour le développement d'applications pour iOS.

PAS D'IOS SANS MACOS ET XCODE

iOS est un système fermé, c'est-à-dire que pour créer les paquets d'applications et les distribuer aux appareils mobiles il faut impérativement Xcode (environnement de développement intégré) et un ordinateur Mac avec macOS. De même, le simulateur iOS ne peut être utilisé que sous macOS. Vous pouvez également utiliser un service de *cloud computing*, c'est-à-dire «louer» un ordinateur Mac hébergé avec tous les outils de développement nécessaires et le commander à travers une connexion à distance, comme le service MacInCloud [2]. L'écran est ensuite transféré sur l'ordinateur de développement propre. Cela résout le problème de sorte que vous pouvez créer le paquet d'applications et utiliser le simulateur iOS pendant la programmation. Dans le nuage, il n'est pas possible d'utiliser un appareil réel. L'auteur a fait de bonnes expériences avec le service mentionné. Le logiciel PAServer pour la commande à distance du Mac est préinstallé, vous pouvez donc démarrer immédiatement après l'enregistrement et la connexion. Pour commencer, il existe la formule flexible *Pay-as-You-Go* : vous ne payez que le temps d'utilisation du Mac dans le nuage.

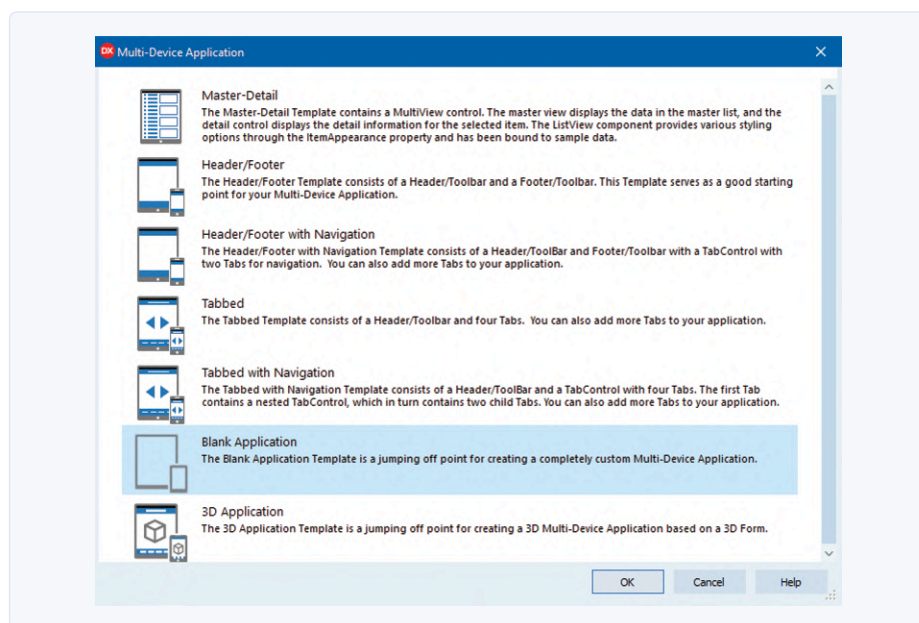


Figure 5. Création d'une nouvelle application multi-appareils.

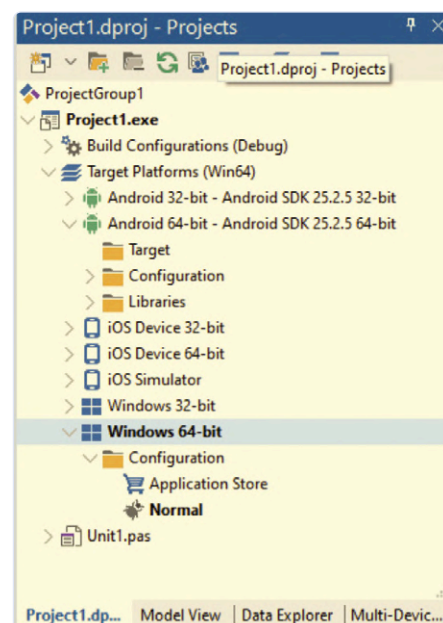


Figure 6. Les plates-formes cibles sont sélectionnées directement dans Delphi

«Hello Elektor». Enregistrez le projet et les fichiers associés à l'aide du menu *Fichier*. Nous avons déjà un premier test sur les différentes plateformes. Ouvrez le nœud *Plates-formes cibles* dans le *gestionnaire de projet* et sélectionnez la plate-forme *Windows 32 bits* en double-cliquant dessus (fig 6). Démarrez l'application (flèche verte). Il fonctionnera comme une application *Windows* «normale» (fig 7).

Fermez maintenant l'application et retournez à l'environnement de développement. Sélectionnez *Android 64 bits* comme plate-forme cible. Si le téléphone/tablette a été correctement configuré ci-dessus, l'appareil connecté apparaît maintenant ici. Activez-le et redémarrez l'application. L'application est installée et exécutée sur l'appareil mobile. Vous pouvez également prévisualiser l'application pour un

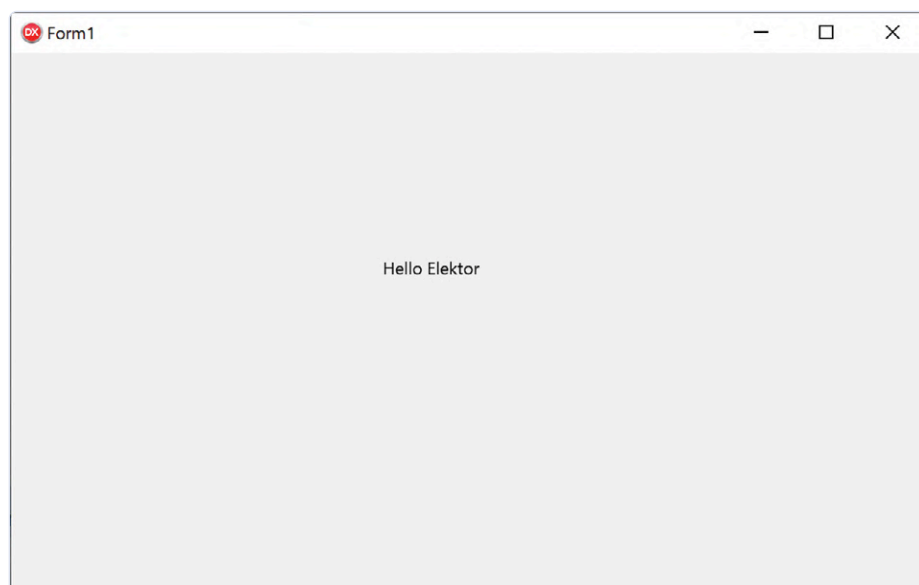


Figure 7. L'application de test en tant qu'application Windows.

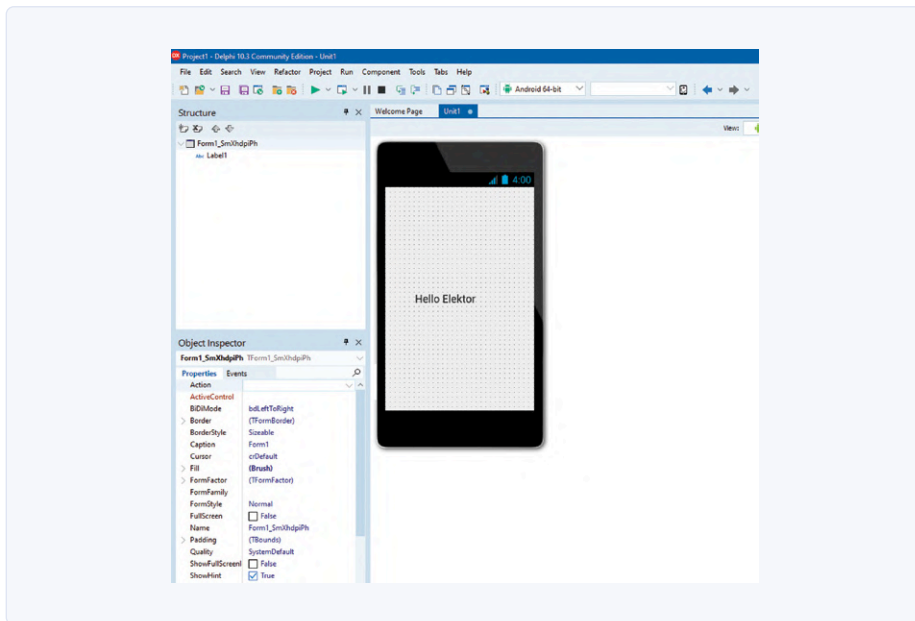


Figure 8. L'aperçu de l'application pour Android dans le RAD Studio Designer.

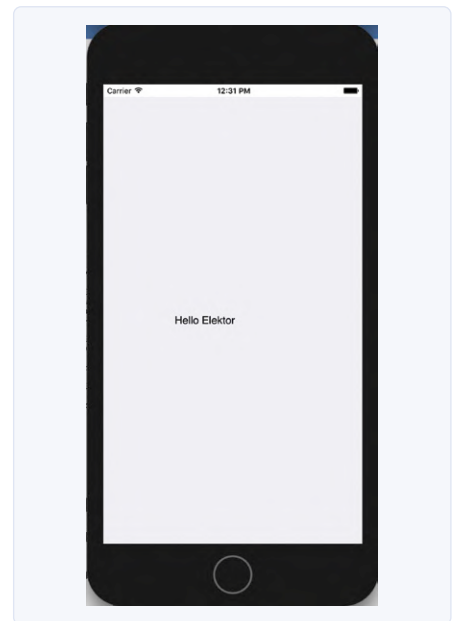


Figure 9. «Hello Elektor» sur le simulateur de l'iPhone.

appareil *Android* dans le *Designer* de l'environnement de développement (fig. 8).

Si vous avez également configuré le système pour iOS, vous pouvez vous connecter au Mac et, via le serveur PAS (voir ci-dessus), à *Xcode*, au simulateur et à un *iPhone* et y exécuter l'application (fig. 9). Après cette introduction succincte, nous voulons maintenant créer une application plus complète et nous familiariser davantage avec son fonctionnement.

Une application pour électroniciens

Dans cette section, nous décrivons la conception d'une application permettant de déterminer la résistance à partir des anneaux de couleur. Nous concevons d'abord l'interface dans l'outil graphique. Encore une fois, choisissez *Fichier | Nouveau | Application Multi-Dispositifs - Delphi* comme modèle et maintenant la sélection *Onglet*. On obtient une vue avec

des onglets (dans *Android* en haut, dans iOS en bas). Nous devons maintenant remplir le contenu de ces onglets en conséquence. Le modèle contient quatre onglets (type *TTabItem*), que nous réduisons à deux, c'est-à-dire que nous supprimons deux onglets en utilisant la vue *Structure*. Dans l'onglet 1, nous sélectionnons les éléments du code couleur et dans l'onglet 2, nous plaçons les informations générales sur l'application. Il est judicieux de faire un croquis avant de mettre en place l'interface utilisateur et de préciser ainsi où les contrôles doivent être placés (du papier et un crayon suffisent). La disposition des éléments de contrôle ne se fait généralement pas avec des informations de position absolue, car il faut tenir compte du fait que nous lancerons l'application sur des appareils mobiles avec des tailles d'écran et des résolutions différentes. Une structure tabulaire peut être un bon point de départ pour une répartition approximative de l'espace. Les différents éléments sont ensuite disposés en lignes et en colonnes. Pour chaque ligne et colonne, nous pouvons à nouveau spécifier la taille (absolue, ou, mieux, relative). Pour notre application, nous avons choisi une grille de cinq lignes et quatre colonnes. Utilisez une commande du type *TGridPanelLayout*. Sélectionnez-le dans la palette et faites-le glisser jusqu'au premier onglet. Pour qu'il occupe tout l'espace, réglez le paramètre *Align* sur la valeur *Client*. En utilisant la vue *Structure*, ajoutez un total de cinq lignes et quatre colonnes à ce *TGridPanelLayout*. Vous pouvez diviser la taille des

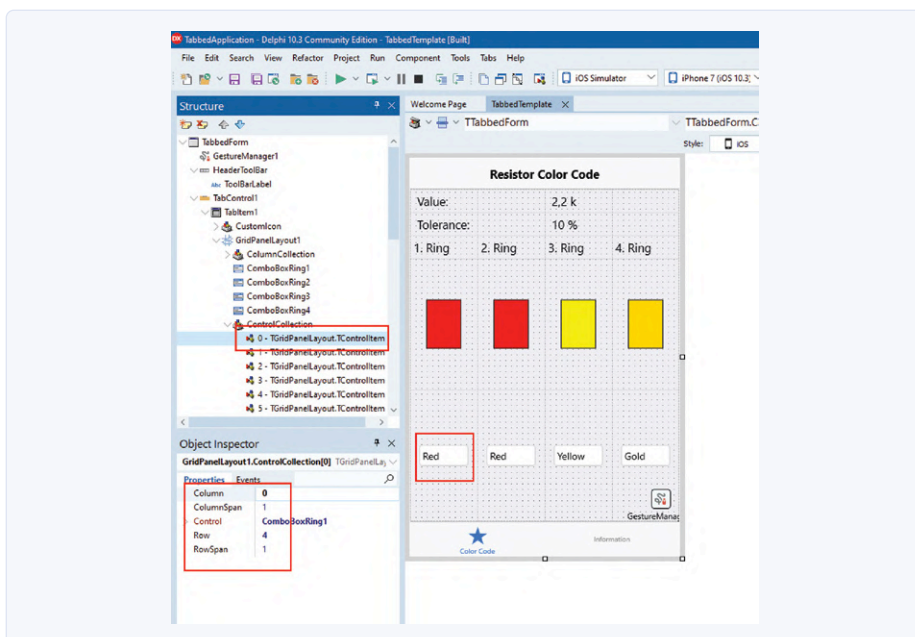


Figure 10. Le positionnement des éléments de contrôle se fait ici dans une grille de tableau.

couleur	anneau 1 position 1	index dans TComboBox	anneau 2 position 2	index dans TComboBox	anneau 3 multiplicateur	index dans TComboBox	anneau 4 tolérance	index dans TComboBox
noir	0	0	0	0	1	0	-	-
marron	1	1	1	1	10	1	1%	0
rouge	2	2	2	2	100	2	2%	1
orange	3	3	3	3	1 000	3	-	-
jaune	4	4	4	4	10 000	4	-	-
vert	5	5	5	5	100 000	5	0,5 %	2
bleu	6	6	6	6	1 000 000	6	0,25 %	3
violet	7	7	7	7	10 000 000	7	0,1 %	4
gros	8	8	8	8	-	-	0,05 %	5
blanc	9	9	9	9	-	-	-	-
or					0,1	8	5 %	6
argent					0,01	9	10 %	7

Tableau 1 Signification des anneaux de couleur et position (index) dans la boîte de sélection respective (TComboBox).

lignes en pourcentage, par exemple, comme suit : 7 %, 7 %, 7 %, 40 % et 39 % (total = 100 %). Vous divisez les quatre colonnes en parts égales, c'est-à-dire avec une taille de 25 % chacune.

Il s'agit maintenant de placer les contrôles dans ce tableau. Sélectionnez un élément souhaité dans la palette et faites-le glisser vers l'onglet. Pour la sélection des couleurs, nous utilisons des contrôles de type *TComboBox*. Dans la vue *Structure*, nous pouvons alors définir la position exacte (ligne et colonne) de chaque élément sous le nœud *TabItem1 | Control Collection*. La **figure 10** illustre cette situation pour la sélection de la couleur du premier anneau.

Les positions *Column : 0* et *Row : 4* sont sélectionnées, puisque la case de sélection doit également être placée à cette position. Pour l'affichage du texte, nous utilisons des contrôles de type *TLabel*, par exemple pour les étiquettes ou les notes. Nous utilisons également *TLabel* pour les valeurs que nous voulons afficher plus tard, par exemple pour la valeur de résistance calculée. Nous simulons les anneaux colorés avec des rectangles colorés, c'est-à-dire des contrôles de type *TRectangle*. Nous voulons ajuster leur couleur plus tard à partir du code source. Dans les champs de sélection des anneaux de couleur (*TComboBox*), nous ajoutons les couleurs disponibles selon le modèle connu du **tableau 1**.

La mise en page de l'application dans l'outil graphique est illustrée à la **figure 11**. Après avoir lancé l'application (simulateur ou dispositif physique), nous voyons la disposition de l'onglet (**fig. 12**).

Nous avons initialement rempli l'onglet *Information* uniquement avec un logo (contrôle de type *TImage*) et un texte (**fig. 13**).

L'étape suivante consiste à mettre en place la logique de détermination de la valeur de résis-

tance. Un nouveau calcul et une mise à jour de l'affichage doivent être effectués chaque fois que nous sélectionnons une nouvelle valeur de couleur. Pour faciliter l'identification des contrôles de l'interface dans le code source, nous donnons aux éléments des noms uniques. À cette fin, la propriété *Name de l'élément* est ajustée dans chaque cas, par exemple *LabelValue*, *RectangleRing1*, *ComboBoxRing1* etc.

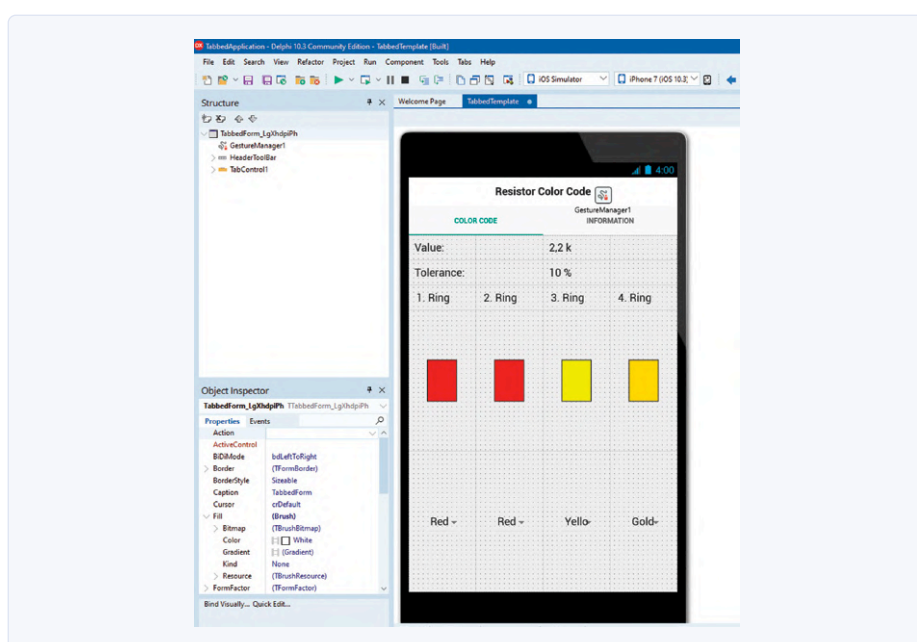


Figure 11. La présentation de l'application dans la vue Delphi - Android.

Liste 1. Affichage des couleurs selon la sélection.

```
case digit1 of
  0:
  begin
    RectangleRing1.Fill.Color := TAlphaColors.Black;
  end;
  1:
  begin
    RectangleRing1.Fill.Color := TAlphaColors.Brown;
  end;
  2:
  begin
    RectangleRing1.Fill.Color := TAlphaColors.Red;
  end;
  ...
  ...
  9:
  begin
    RectangleRing1.Fill.Color := TAlphaColors.White;
  end;
end;
```

Liste 2. Calcul de la valeur de résistance et formatage de l'affichage.

```
if digit3 < 8 then
  value := Power10((digit1 * 10 + digit2), digit3)
else if digit3 = 8 then
  value := (digit1 * 10 + digit2) * 0.1
else if digit3 = 9 then
  value := (digit1 * 10 + digit2) * 0.01;

result := FloatToStr(value) + ' Ohm';

if value > 1000 then
  result := FloatToStr((value / 1000)) + ' k Ohm';

if value > 1000000 then
  result := FloatToStr((value / 1000000)) + ' M Ohm';

LabelValue.Text := result;
```

Nous sélectionnons les cases de sélection pour les couleurs des anneaux tous ensemble et choisissons l'événement *OnChange*. Par un double-clic dans l'inspecteur d'objets, nous faisons en sorte que l'environnement de développement crée une procédure pour cet événement. Ce code source est donc appelé chaque fois que nous changeons la sélection de couleur.

Nous commençons par ajuster la couleur des rectangles en fonction de la sélection

de l'anneau. Cela se fait, par exemple, par l'utilisation :

Anneau rectangulaire 1, couleur de remplissage : = Couleurs TAlpha, rouge

Le rectangle sera alors dessiné en rouge. Pour la première case de sélection, elle ressemblera à la **liste 1**.

Pour les autres anneaux de couleur, la sélection est ajustée en fonction de l'indice ci-des-

sus (voir **tableau 1**). Le calcul de la valeur de résistance suit, c'est-à-dire
(Valeur du chiffre 1 * 10 + valeur du chiffre 2) * 10^{multiplificateur}.

Le calcul est effectué en ohms. Pour les valeurs supérieures à 1.000, une conversion est effectuée en kΩ et pour les valeurs > 1.000.000 en MΩ. Pour les couleurs de bague or et argent, une multiplication avec les valeurs 0,1 et 0,01 est effectuée. Il ne manque que la valeur de la tolérance dans le tableau ci-dessus (**liste 2**).

Notre application est maintenant fonctionnelle : vous pouvez choisir n'importe quelle combinaison de couleurs des anneaux et la valeur correcte de la résistance est affichée. Vous pouvez maintenant créer les paquets d'application pour *Android* et/ou *iOS* et déployer l'application. Il est également possible de rendre l'application disponible dans *Google Play* ou l'*Apple Store* à partir de *Delphi*. Vous pouvez télécharger le code source de l'exemple de programme sur le site d'Elektor [5].

Autres possibilités

Dans l'exemple, nous avons déjà montré quelques possibilités de développement d'applications avec *Delphi*. Mais il y a bien plus que cela. Voici quelques exemples :

- > **Composants visuels** : Pour créer des interfaces attrayantes, il existe divers composants (boutons, champs de texte, cases à cocher, onglets, etc.). Vous pouvez placer ces contrôles à l'aide du concepteur graphique et les configurer via les propriétés. Le système crée automatiquement les éléments appropriés pour la plate-forme cible (*Android*, *iOS*, *Windows*) lorsque vous compilez l'application.
- > **Composants non visuels** : Ces derniers constituent des éléments de base importants pour la programmation, qui sont nécessaires à maintes reprises. Il s'agit p. ex. d'un composant de temporisation (*timer*) ou de composants destinés à faciliter l'accès aux bases de données.
- > **Spécificités des applications mobiles** : des composants visuels et non visuels sont proposés, ce qui permet de résoudre plus rapidement les tâches typiques de la programmation d'applications mobiles, par exemple l'accès aux fonctions de suivi ou à la caméra du téléphone ou la gestion de l'échange de données avec les serveurs (*backends*) dans le nuage.

Ces possibilités de développement sont nécessaires successivement, si vous voulez créer une application mobile jusqu'à ce qu'elle soit prête pour une utilisation productive. Le développeur est également aidé à générer les paquets d'applications directement à partir de l'environnement de développement, qui sont nécessaires pour la distribution des applications par *Google* ou dans *l'Apple Store*.

Conclusion et perspectives

De telles applications « de poche » requièrent donc un travail qu'on aurait tort de sous-estimer. Grâce à un outil approprié, vous vous épargnerez bien des tracas et développer simultanément pour *Android* et *iOS* à partir d'un code source commun. *Delphi* offre une approche intuitive, qui devrait séduire de nombreux électroniciens familiers de *Visual Basic*, *Windows Forms*, etc.. L'interface utilisateur est conçue directement et assez rapidement dans l'éditeur graphique. Cette approche est également utile si vous devez écrire une application pour *Windows* ou *Linux*. De cette façon, vous pouvez réutiliser le code source plusieurs fois. En un clic, vous pouvez passer d'une plate-forme à l'autre. L'environnement de développement produit automatiquement les paquets d'applications nécessaires. Dans l'idéal, vous n'aurez rien d'autre à faire. 

200265-02

Votre avis, s'il vous plaît...

Vous pouvez adresser vos commentaires ou vos questions à l'auteur (en anglais ou en allemand) : v.krypczyk@larinet.com.



DANS L'E-CHOPPE D'ELEKTOR

- > **Livre Android App Development for Electronics Designers**
www.elektor.fr/android-app-development-for-electronics-designers
- > **Livre C# Programming for Windows and Android**
www.elektor.fr/c-programming-for-windows-and-android

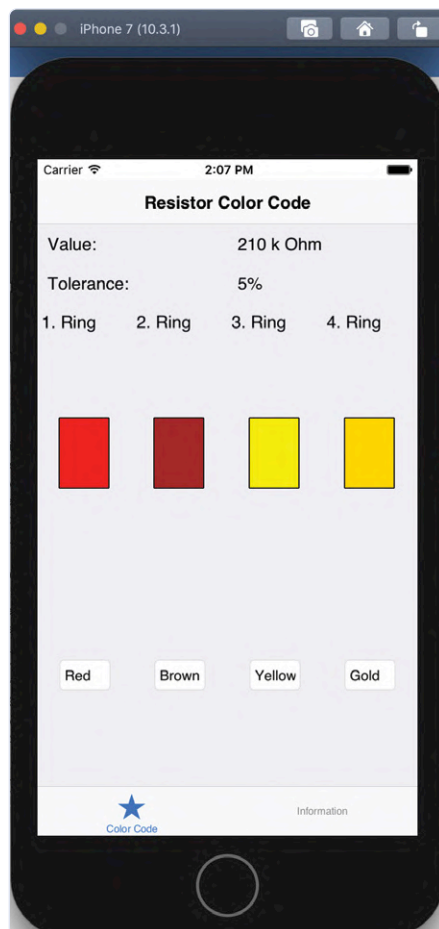


Figure 12. L'application lancée - toujours sans fonction - dans le simulateur (iOS)

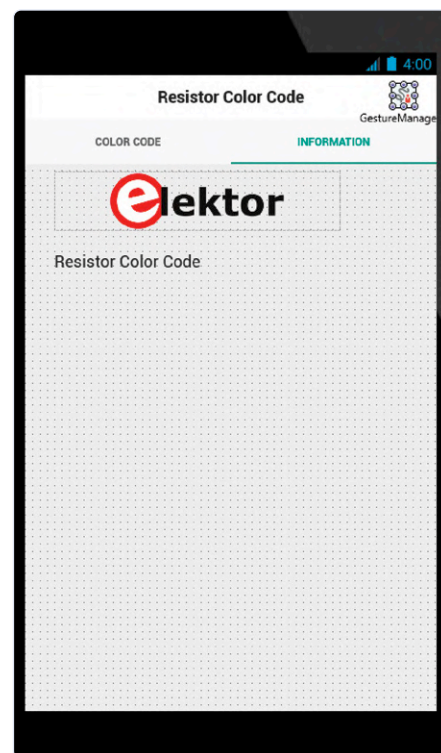


Figure 13. L'onglet Information.

Ont contribué à cet article :

Auteur : **Veikko Krypczyk**

Rédacteur : **Jens Nickel**

Traduction : **C. Zig**

Maquette : **Giel Dols**

LIENS

- [1] <https://www.embarcadero.com/products/delphi/starter/free-download>
- [2] <http://www.macincloud.com/>
- [3] http://docwiki.embarcadero.com/RADStudio/Rio/en/Android_Mobile_Application_Development
- [4] http://docwiki.embarcadero.com/RADStudio/Rio/en/IOS_Mobile_Application_Development
- [5] **La page de cet article sur le site d'Elektor** : www.elektormagazine.fr/200265-02

sur le vif

Professions en déficit de main d'oeuvre

Un jour, au lycée technique que je fréquentais à Bruxelles, un prof de français passionné de musées et amateur d'opéra, nous a parlé d'une exposition des œuvres du peintre surréaliste belge René Magritte. L'un de ses tableaux les plus célèbres, nous dit-il, représente une pipe avec la légende «Ceci n'est pas une pipe».

Notre prof s'est contenté d'affirmer sèchement «Vous autres ne comprenez pas ça» puis il a continué sa leçon. Tout le monde avait perçu son mépris pour les stupides étudiants en sciences que nous étions à ses yeux, incultes et imperméables à tant de beauté.

Je crains qu'il ne s'agisse pas d'une exception.

Malgré la demande constante de personnel qualifié techniquement et la prétendue pénurie sur le marché du travail, resterons-nous (comme) des parias ?

Déglingue en cascade

En dépit des nombreuses réformes de l'enseignement et malgré l'accent mis et remis sur les matières techniques et scientifiques, l'enseignement secondaire belge semble être le théâtre de la course permanente au gaspillage. Les meilleurs élèves à l'école primaire, quand j'avais douze ans, étaient tout naturellement orientés vers l'enseignement secondaire général. C'est la direction suivie par la majorité de mes camarades tandis que les élèves les moins « bons » comme on disait alors, les tocards si vous préférez, s'engageaient à reculons dans l'enseignement secondaire technique et professionnel.

Quel gâchis ! C'était déjà une épreuve de devoir troquer le latin et les langues modernes contre l'économie, mais ça devenait



franchement désastreux lorsque vous étiez banni dans l'enseignement technique.

Ce destin cruel m'a heureusement été épargné. *Veni, vidi, vici !* Une telle déglingue en cascade alimente forcément un cercle vicieux : l'afflux d'élèves moins bons abaisse le niveau de l'enseignement technique, ce qui déprécie la technologie aux yeux des parents. Ceux-ci ne rêvent pour leurs enfants que de l'enseignement général, même pour ceux qui s'intéressent à la technologie et présentent des aptitudes évidentes. Par ailleurs, les naufragés de l'enseignement technique migrent vers la formation professionnelle. On croit savoir que les étudiants d'une discipline technique ne sont « pas faits » pour l'enseignement supérieur, d'où il résulte que ceux qui accéderont aux cours techniques supérieurs dans les universités sont justement les étudiants bénéficiant d'une formation générale totalement privée de connaissances techniques pratiques.

Là on les gave de théorie, d'intégrales doubles, triples et circulaires, d'équations différentielles et autres inconnues, on leur apprend peut-être à calculer un câble de distribution, mais jamais on ne leur montrera cet animal dans son habitat naturel.

Fermez-le ban, la boucle est bouclée !

J'adore la technique, mais ces années-là furent les plus ennuyeuses de ma vie.

La technique réduite à l'art de crocheter les serrures

Dans une vie d'étudiant, le jour de la remise des diplômes, avec ou sans mention, est un grand moment. Le lendemain de la proclamation solennelle, on est ramené à la très dure réalité, et s'il n'a pas un maximum de chance, l'étudiant qui vient de réussir devient



DANS L'E-CHOPPE D'ELEKTOR :

- > Livre «Réparez vous-même vos appareils électroniques»
www.elektor.fr/19284
- > Livre «Boîte à idées pour le Raspberry Pi»
www.elektor.fr/19416



en, moins de temps qu'il n'en faut pour le dire, un perdant au chômage. Le monde des affaires est régi par des comptables, des économistes, des opportunistes, des consultants et des criminels en col blanc, un biotope peu favorable aux sensibilités techniques et scientifiques. Vous trouvez peut-être que j'exagère, mais écoutez bien comment ces gens-là parlent des techniciens qui « trouveront bien un moyen de nous bricoler ça ». Pour eux les problèmes se réduisent à des serrures dont ils ont perdu la clé, et qu'il suffit de crocheter. Ils ne méritent pas le moindre respect [1].

Cette sous-estimation généralisée des techniciens est traitée avec humour dans la sitcom britannique «*The IT Crowd*» (= *les gens de l'informatique*) du réalisateur irlandais Graham Linehan. En tant qu'ancien administrateur de système dans une certaine société commerciale, je confirme qu'il y a beaucoup de vécu dans les expériences folles de Roy Trenneman et Maurice Moss. Je recommande !

La pénurie de personnel technique qualifié : un conte de fées

On nous rebat les oreilles avec la litanie du manque de profils techniques et d'ingénieurs. En Belgique, ces postes figurent depuis des années sur la liste des professions pour lesquelles le recrutement est sous tension extrême. Si une telle pénurie était réelle, la rémunération et les conditions de travail des techniciens devraient être proportionnellement élevées. Rien n'est moins vrai : de nombreux techniciens professionnels qualifiés sont sous-payés en Belgique. Avec un emploi de bureau standard de neuf à cinq, dont l'essentiel consiste à faire des photocopies et du café et à arroser les plantes, vous gagnerez mieux votre vie que dans un emploi de technicien. Et vous serez probablement logé dans un bureau agréable, bien éclairé et aéré, et pas dans un réduit malodorant, sous un tube néon blafard et grésillant.

Non, il n'y pas de marché pour les profils techniques supérieurs. Une de mes connaissances, ingénieur en électronique de premier plan – ils appellent ça *Senior Electronics Hardware Engineer* – dans une société de renom dont justement je tairai le nom, s'est vu remercié, lors d'une « fête du personnel » après des années de loyaux services en remerciement desquels on lui a décerné un piètre certificat encadré et un bouquet de fleurs.

Merci et bon vent ! Pas de quoi pavoiser surtout quand on sait combien dans la même boîte de primes et d'augmentations de salaire sont accordées aux cadres et aux commerciaux. Connaissez-vous la devise de ces gens-là ? C'est « À quoi bon me fouler à devenir ingénieur (ou scientifique) quand je peux être son patron ? » ◀

200455-03

Votre avis, s'il vous plaît...

Vous pouvez adresser vos questions ou commentaires en latin seulement à la rédaction d'Elektor : redaction@elektor.fr

Ont contribué à cet article :

Texte : Ilse Joostens (Belgique)
Rédacteur : Eric Bogers

Traduction : Alice Coper
Conception : Giel Dols

LIEN

[1] Pour se défouler un peu : www.topito.com/top-generateurs-bons-mots-internet-replique

5G qui façonne qui... les infrastructures ou la société ?

La question n'est pas anodine,
les conséquences non plus

Tessel Renzenbrink

« Une société de la connaissance qui privatise le savoir est bien sûr une absurdité », déclare Niels ten Oever, associé à l'université d'Amsterdam et à l'université A&M du Texas. Ce chercheur, qui étudie l'infrastructure des réseaux d'information et leur impact sur la société, parle dans cet entretien de l'introduction du réseau mobile de cinquième génération dit 5G.

« Une société de la connaissance qui privatise le savoir est bien sûr une absurdité », déclare Niels ten Oever, associé à l'université d'Amsterdam et à l'université A&M du Texas. Ce chercheur, qui étudie l'infrastructure des réseaux d'information et leur impact sur la société, parle dans cet entretien de l'introduction du réseau mobile de cinquième génération dit 5G.

« Habituellement, nous ne pensons pas aux infrastructures, dit Ten Oever. Sauf quand elle s'effondre. Courant coupé, route fermée... et déjà tout le monde prend (brièvement) conscience de leur importance. En général, cependant, nous sous-estimons l'importance de la conception des infrastructures. Prenons le réseau routier et son impact sur votre vie quotidienne. Une route à quatre voies flanquée de trottoirs étroits encourage une utilisation de l'espace public différente de celle des larges voies piétonnes et de pistes cyclables. C'est pareil pour les infrastructures d'information telles que la 5G. L'architecture du réseau porte ses possibilités et ses limites. Il faut donc se demander à quoi doit servir tel ou tel réseau. »

Virtualisation du réseau

« L'arrivée de la 5G va créer une nouvelle infrastructure, poursuit Ten Oever. Les fréquences ne sont pas celles de la 4G. Il y aura donc de nouvelles antennes, d'autres routeurs et d'autres commutateurs. Cela implique de nouveaux *paradigmes informatiques*. Comme ces routeurs et commutateurs sont des *ordinateurs* plus *universels*, ils servent donc à des fins beaucoup plus générales : des appareils qui peuvent simplement fonctionner sous Linux. Cela contraste avec la génération actuelle du matériel orienté vers les applications et des micrologiciels spéci-

fiques. Dès lors que le logiciel joue un rôle beaucoup plus important, les fonctions de réseau peuvent être virtualisées ».

La virtualisation des réseaux rend le réseau de plus en plus intelligent, ce qui implique qu'il faut faire son deuil des principes fondamentaux de l'internet : le principe du *End-to-End* ou *de bout en bout*, ce qui signifie que chaque ordinateur du réseau peut se connecter à tout autre ordinateur connecté. L'idée des fondateurs était que le réseau lui-même serait le plus transparent possible, qu'il ne serait donc constitué que de tuyaux par où passent « bêtement » des données. L'intelligence – c'est-à-dire la puissance de calcul et de stockage des données – se situe aux extrémités du réseau : les ordinateurs des utilisateurs. Cependant, la tendance à déporter l'intelligence vers le réseau lui-même au détriment des appareils finaux n'est pas neuve. C'est le principe du nuage : le stockage des données est effectué sur des serveurs tiers où s'exécutent aussi de plus en plus d'applications logicielles. L'introduction de la 5G permettra d'accentuer cette tendance vers plus d'intelligence dans le réseau, par exemple dans les *réseaux centrés sur l'information*, où les données sont stockées sur le réseau lui-même. Ten Oever donne un exemple de ce à quoi cela pourrait ressembler.

Game of Thrones dans la baignoire

« Il arrive déjà que des applications vous fassent des recommandations inspirées par votre propre comportement. Nitfex prévoit où vous utiliserez Nitfex : vous regardez toujours *Game of Thrones* dans la baignoire. Vous écoutez toujours vos podcasts sur le vélo. Un podcast pourra donc être mis en cache dans le lampadaire "intelligent" à côté

duquel vous passez souvent. Lorsque vous arrivez au travail, votre ordinateur est déjà allumé et vos programmes habituels sont déjà ouverts. C'est pratique, mais comment garder le contrôle de ce confort ? Faut-il d'ailleurs un tel contrôle ? Comment savoir qui contrôle quoi et en fonction de quels critères ?

« Nous n'en sommes pas encore là, mais avec l'infrastructure polyvalente créée avec la 5G, c'est devenu possible, poursuit Ten Oever. Cela permettra d'optimiser encore davantage un tel réseau, comme en rêvent tous les opérateurs, qui pourront en tirer un bien meilleur parti. Le rendement d'un réseau est un paramètre optimisable, au prix, hélas ! de la disparition de toute possibilité de contrôle par le consommateur lui-même. Nous saurons de moins en moins quel appareil est connecté à quoi et quelles informations il envoie. En revanche, les possibilités de filtrage augmentent fortement, ainsi que l'extraction de données encore plus critiques pour la vie privée. Il n'est pas exagéré d'imaginer des profils de *censure personnalisée*, qui permettraient d'interdire à une certaine personne l'accès à certains contenus, ou encore de l'empêcher d'entrer en contact avec telle autre personne. »

Pléthore de normes et brevets 5G

Ten Oever étudie le fonctionnement des processus de décision dans la normalisation de la 5G. La 5G est un ensemble de technologies différentes normalisées en trois phases dont la deuxième devrait s'achever en 2020. Les spécifications de la 5G sont définies par 3GPP, association de six organisations de normalisation, dont l'ETSI européen. Les normes définitives seront adoptées au sein de l'Union internationale des télécommunications (UIT), où les États membres ont droit de vote. Ten Oever : « Il est assez difficile d'évaluer pourquoi certaines décisions sont prises. L'une des raisons du développement de certaines technologies est le brevet. Celui-ci joue un rôle moindre dans le monde de l'internet. Les entreprises y détiennent des brevets, qu'elles utilisent cependant de manière défensive. C'est une sorte de trêve : tu as des brevets, j'ai des brevets, mais nous ne les revendiquons pas. Si tu commences, j'en ferai autant.

« Dans le secteur de la téléphonie mobile, au sein du 3GPP, le rôle du matériel et des logiciels propriétaires est beaucoup plus grand. C'est aussi ce qui rapporte le plus d'argent. Pour une entreprise comme *Qualcomm*, les brevets rapportent bien plus que la vente de matériel. Les entreprises s'efforcent donc de placer leur technologie brevetée dans l'élaboration de normes. Je m'attends à une explosion du nombre de nouveaux appareils dans les années à venir. Supposons que pour ses licences une entreprise reçoive 10 centimes par appareil. Non seulement ce sont des montants élevés, mais, pour recueillir cette manne, l'entreprise n'aura plus jamais à faire quoi que ce soit. »

"Nous ignorons tout du fonctionnement de ces réseaux"

Les brevets rendent la 5G plus opaque que l'internet actuel. Ten Oever : « Vous et moi pouvons apprendre comment fonctionne l'internet, car ces connaissances sont publiques. Vous pouvez construire un internet



miniature chez vous, mais nul ne pourrait implanter un mini réseau 5G équivalent, car une grande partie des connaissances sont verrouillées par des brevets et des licences. Le manque de transparence fait également obstacle à la recherche et à l'innovation, déclare Ten Oever. Maintenant, nous suivons des protocoles. Ce sont des règles relativement simples, mais bientôt tout sera optimisé par des algorithmes. Cela soulève beaucoup de questions complexes auxquelles il serait bon de réfléchir ensemble, universitaires et chercheurs inclus. Le marché devrait être ouvert aux nouveaux venus avec de nouvelles idées. C'est impossible maintenant, car nous ignorons tout du fonctionnement de ces réseaux. Les brevets sont une barrière à l'échange de connaissances.

« Au lieu de protéger les intérêts des entreprises établies, nous devrions promouvoir le matériel et les logiciels ouverts, déclare Ten Oever. Ainsi nous créerons un écosystème numérique ouvert dans lequel l'innovation sera possible. Ici les gouvernements ont un rôle à jouer en soutenant fermement de telles demandes. Il existe des directives strictes pour l'approvisionnement en eau et en électricité, des procédures, des contrôles et des exigences. Pourquoi pas pour les infrastructures numériques ? La création d'une infrastructure cruciale implique de grandes responsabilités. Cette prise de conscience est insuffisante pour l'internet et les télécommunications. Les gouvernements semblent insuffisamment informés et ignorent la nécessité de telles exigences. « Le droit à la connaissance et l'accès au savoir devraient englober notre infrastructure d'information », conclut Ten Oever. Il doit primer la protection des intérêts des entreprises. Si nous sommes une société de l'information et de la connaissance, celles-ci doivent être accessibles. Une société de la connaissance qui privatise le savoir est bien sûr une absurdité. »

200468-04

Votre avis, s'il vous plaît...

Vous pouvez poser des questions sur cet article ou le commenter en vous adressant à la rédaction d'Elektor redaction@elektor.fr

On contribué à cet article :

Texte : Tessel Renzenbrink

Traduction : Catherine Rappin

Rédaction : Eric Bogers

Maquette : Giel Dols

multitâche en pratique avec l'ESP32 (5)

Notification d'événements de tâches

Warren Gay (Canada)

FreeRTOS offre des fonctions de file d'attente, de sémaphore et de *mutex* qui, pour la synchronisation, peuvent parfois sembler trop compliquées lorsque les besoins sont eux-mêmes simples. La version V8.2.0 de FreeRTOS a introduit le concept de *notification directe des événements* aux tâches, ce qui offre au programmeur une méthode légère de synchronisation avec les tâches.

Chaque tâche comprend une valeur de notification d'événement intégrée de 32 bits, initialisée lors de la création de la tâche. Comme cette valeur est intégrée à chaque tâche, aucune mémoire vive supplémentaire n'est requise, soit une économie considérable par rapport à l'utilisation p. ex. d'objets sémaphores.

Limitations

Il existe toutefois certaines limites à l'utilisation des notifications de tâches directes :

- Vous ne pouvez pas envoyer de notification à un ISR, car un ISR n'est pas une tâche. Cependant, un ISR *peut* notifier une tâche.
- Une seule tâche peut être notifiée par un appel de notification de tâche (les groupes d'événements doivent être utilisés pour notifier plusieurs tâches).
- Les événements de notification ne peuvent pas être mis en mémoire tampon comme les files d'attente.
- La tâche de *notification* (ou ISR) ne bloque pas son exécution pour attendre que la tâche de réception reçoive l'événement. L'événement est simplement affiché et l'appel revient immédiatement, sans entrave.

Si l'une de ces restrictions est en conflit avec vos exigences, vous devrez choisir un autre mécanisme FreeRTOS à la place.

En attente d'un événement

La première étape de l'utilisation des notifications d'événements de tâches consiste à informer la tâche destinataire d'attendre une notification. En d'autres termes, la tâche doit être configurée pour s'interrompre jusqu'à ce qu'un événement soit déclenché. Ceci est mis en œuvre en utilisant l'une des fonctions suivantes :

- `ulTaskNotifyTake()`
- `xTaskNotifyWait()`

La fonction `ulTaskNotifyTake()` est la plus simple des deux, c'est celle que nous examinons dans cet article. La fonction `xTaskNotifyWait()` est plus avancée et est examinée en détail dans mon livre *FreeRTOS for ESP32-Arduino* publié par Elektor.

`ulTaskNotifyTake()`

La tâche qui reçoit une notification bloque son exécution en appelant `ulTaskNotifyTake()`. L'appel nécessite deux arguments :

```
uint32_t ulTaskNotifyTake(
    BaseType_t xClearCountOnExit, // pdFALSE or pdTRUE
    TickType_t xTicksToWait
);
```

Le premier argument opère sur le mot de notification de tâche de 32 bits, également appelé mot d'événement de tâche (tableau 1). Le deuxième paramètre est la valeur familière de la *temporisation* dans les tics (utilisez la macro `pdMAX_DELAY` si vous ne voulez pas de temporisation). Dans tous les appels à `ulTaskNotifyTake()`, l'exécution de la tâche appelante est bloquée tant que le mot d'événement de la tâche reste à zéro. Dès que la valeur devient non nulle, l'argument `xClearCountOnExit` détermine comment le mot d'événement de la tâche est mis à jour avant de retourner.

Tableau 1. La signification des arguments `xClearCountOnExit` pour `ulTaskNotifyTake()`.

<code>xClearCountOnExit</code>	Effet sur le mot de notification des tâches de 32 bits
<code>pdFALSE</code>	Décrémenter, blocage de l'exécution si la valeur était nulle avant la décrémentation, avant de renvoyer la valeur antérieure.
<code>pdTRUE</code>	Effacement de la valeur à 0, blocage de l'exécution si la valeur était déjà nulle, avant de renvoyer la valeur antérieure.

La valeur renvoyée par l'appel de fonction est la valeur du mot de notification de l'événement *avant* qu'il ne soit effacé ou décrémenté. En cas de dépassement de *délai*, la valeur renvoyée sera également égale à zéro.

Notification binaire

Lorsque la valeur de l'argument `xClearCountOnExit` est `pdTRUE`, l'appel à `ulTaskNotifyTake()` fonctionne comme l'opération de

prise de sémaphore binaire. Si le mot d'événement de notification de la tâche est *déjà non nul*, l'appel renvoie immédiatement et efface le mot d'événement (sa valeur antérieure non nulle est cependant renvoyée). Si le mot de notification de la tâche était à zéro au moment de l'appel, l'exécution de la tâche est *bloquée* jusqu'à ce que la tâche soit notifiée (sous réserve d'un délai d'attente). Si un délai d'attente se produit, la valeur de retour sera toujours zéro (reflétant la valeur du mot d'événement au moment du retour). Dans ce mode de fonctionnement, l'appel fonctionne comme un sémaphore binaire. Si la tâche devait être notifiée deux fois avant que la tâche réceptrice n'appelle `ulTaskNotifyTake()`, alors la valeur renvoyée sera 2. Cependant, le mot d'événement de la tâche est effacé lors du retour. Cela ne fournit effectivement qu'un seul événement de notification dans ce cas particulier, mais le nombre réel de notifications est facilement déterminé en notant la valeur de retour.

Notification de comptage

Si l'exigence est que les notifications de tâches multiples *doivent* provoquer des réveils de tâches multiples, alors l'argument `xClearCountOnExit` doit être fourni avec la valeur `pdFALSE`. Dans un tel cas, la tâche réceptrice se bloque alors que la valeur de l'événement de la tâche est à zéro (comme auparavant). Chaque notification de la tâche incrémente la valeur de l'événement de tâche, mais elle n'est décrémentée que d'une unité pour chaque appel à `ulTaskNotifyTake()`. Par exemple, si quatre notifications se produisent avant que la tâche réceptrice n'appelle `ulTaskNotifyTake()`, alors cet appel revient immédiatement sans blocage jusqu'à ce que la valeur de notification de la tâche ait été décrémentée à nouveau à zéro. En d'autres termes, la tâche réceptrice doit effectuer quatre appels à `ulTaskNotifyTake()` avant que le mot d'événement ne soit décrémenté jusqu'à zéro.

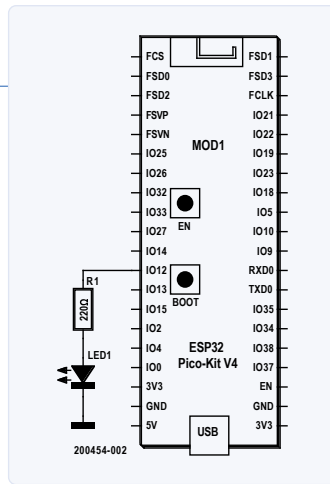


Figure 1. Schéma pour le code `tasknfy1.ino`.

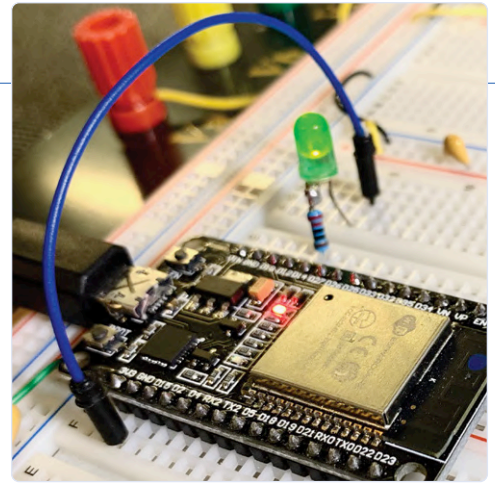


Figure 2. Mise en place du circuit de démonstration final à l'aide de GPIO 12.

Notifier

Alors que la tâche réceptrice utilise la fonction `ulTaskNotifyTake()`, la notification est fournie en utilisant `xTaskNotifyGive()` :

```
BaseType_t xTaskNotifyGive(TaskHandle_t xTaskToNotify);
```

Cette fonction ne nécessite que le *handle* (ou poignée) de la tâche à notifier et renvoie toujours `pdTRUE` (le manuel de FreeRTOS indique que `xTaskNotifyGive()` est défini comme une macro).

Démonstration

Le programme de démonstration très simple (**liste 1**) [1] utilise le moniteur sériel afin que vous puissiez voir la valeur du mot d'événement rapportée. La **figure 1** donne le câblage d'une LED optionnelle, tandis que la **figure 2** illustre la configuration du tableau de bord. Presque toutes les cartes de développement ESP32 peuvent être utilisées si GPIO 12 est disponible.

Le programme `tasknfy1.ino` utilise le `loopTask()` fourni par Arduino

Liste 1. Programme de démonstration `tasknfy1.ino` avec `xTaskNotifyGive()` et `ulTaskNotifyTake()`.

```
0001: // tasknfy1.ino
0002:
0003: #define GPIO_LED      12
0004:
0005: static TaskHandle_t htask1;
0006:
0007: static void task1(void *arg) {
0008:     uint32_t rv;
0009:
0010:     for (;;) {
0011:         rv = ulTaskNotifyTake(pdTRUE, portMAX_DELAY);
0012:         digitalWrite(GPIO_LED, digitalRead(GPIO_LED)^HIGH);
0013:         printf("Task notified: rv=%u\n", unsigned(rv));
0014:     }
0015: }
0016:
0017: void setup() {
0018:     int app_cpu = 0;
0019:     BaseType_t rc;
0020:
0021:     app_cpu = xPortGetCoreID();
0022:     pinMode(GPIO_LED, OUTPUT);
0023:     digitalWrite(GPIO_LED, LOW);
0024:
0025:     delay(2000); // Allow USB to connect
0026:     printf("tasknfy1.ino:\n");
0027:
0028:     rc = xTaskCreatePinnedToCore(
0029:         task1, // Task function
0030:         "task1", // Name
0031:         3000, // Stack size
0032:         nullptr, // Parameters
0033:         1, // Priority
0034:         &htask1, // handle
0035:         app_cpu // CPU
0036:     );
0037:     assert(rc == pdPASS);
0038: }
0039:
0040: void loop() {
0041:     delay(1000);
0042:     xTaskNotifyGive(htask1);
0043: }
```

pour notifier de manière répétée la `task1()` (ligne 42) à intervalles d'une seconde. La fonction `task1()` se bloque alors (ligne 11) jusqu'à ce qu'elle soit notifiée. Une fois notifiée, elle bascule ensuite l'état de la LED (ligne 12). La fonction `task1()` est notifiée par l'utilisation du handle `htask1` (ligne 42). Ce *handle* a été créé dans la fonction `setup()` à la ligne 34.

```
0040: void loop() {
0041:   delay(1000);
0042:   xTaskNotifyGive(htask1);
0043: }
```

Le code de `task1` est presque aussi banal, c'est une boucle sans fin. Il bloque son exécution lors de l'appel à `ulTaskNotifyTake()` dans la ligne 11 pour attendre un événement de notification :

```
0007: static void task1(void *arg) {
0008:   uint32_t rv;
0009:
0010:   for (;;) {
0011:     rv = ulTaskNotifyTake(pdTRUE, portMAX_DELAY);
0012:     digitalWrite(GPIO_LED, digitalRead(GPIO_LED)^HIGH);
0013:     printf("Task notified: rv=%u\n", unsigned(rv));
0014:   }
0015: }
```

La poursuite de l'exécution de `task1` reste bloquée jusqu'à l'arrivée de l'événement de notification de la tâche. La valeur attribuée à `rv` (ligne 11) est la valeur du mot de notification d'événement *avant* qu'il ne soit effacé (en raison de l'argument `pdTRUE`). C'est propre et simple - il n'y a pas d'autres objets sémaphores impliqués et, donc, pas de *handle* supplémentaires.

La session de surveillance en série qui en résulte devrait ressembler à ceci :

```
tasknfy1.ino:
Task notified: rv=1
Task notified: rv=1
...
```

Si vous souhaitez notifier une tâche à partir d'un ISR, utilisez la fonction `xTaskNotifyGiveFromISR()`. La convention du suffixe FreeRTOS «FromISR» permet de travailler en toute sécurité à partir d'un ISR, qui implique souvent une manipulation particulière.

```
void vTaskNotifyGiveFromISR(
    TaskHandle_t xTaskToNotify,
```

```
    BaseType_t *pxHigherPriorityTaskWoken
);
```

Le deuxième argument est l'adresse du drapeau de réveil de l'ISR, indiquant si l'ordonnanceur doit être invoqué ou non (fournir `NULL/NULLptr` lorsque ce n'est pas nécessaire). Cette fonction de notification de tâche constitue un moyen très pratique pour une routine ISR de notifier une tâche.

Notification préalable


Dans cet article, nous n'avons abordé que la paire de fonctions la plus simple, `xTaskNotifyGive()` (ou `xTaskNotifyGiveFive fromISR()`) et `ulTaskNotifyTake()`.

Des fonctions supplémentaires et avancées sont disponibles, notamment les suivantes :

- `xTaskNotify()` (or `xTaskNotifyFromISR()`)
- `xTaskNotifyAndQuery()` (or `xTaskNotifyAndQueryFromISR()`)
- `xTaskNotifyWait()`
- `xTaskNotifyStateClear()`
- `xTaskNotifyValueClear()`

Ces fonctions sont similaires, mais supportent le traitement des événements au niveau du bit. Elles fournissent également des formes plus sélectives de test et de compensation des événements. Vous pouvez vous informer à ce sujet dans le manuel de référence de FreeRTOS [2] ou en ligne [3].

Conclusion

L'API de notification de tâches FreeRTOS fournit au développeur d'applications un mécanisme léger de notification d'événements. Ce mécanisme est particulièrement utile pour mettre en œuvre la notification de tâches à partir d'ISR, car il permet de réduire le temps de traitement des interruptions, tout en permettant d'effectuer le travail difficile dans la tâche notifiée. Même sans utiliser les interruptions, elle économise de la mémoire vive et simplifie l'application en ne nécessitant pas d'objets de synchronisation supplémentaires tels que des sémaphores. 

200454-03

Votre avis, s'il vous plaît...

Vous pouvez adresser vos questions ou vos commentaires à l'auteur (en anglais) à ve3wwg@gmail.com ou (en français) à redaction@elektor.fr

On participé à cet article

Idée et texte : **Warren Gay**

Rédaction : **Stuart Cording**

Maquette : **Giel Dols**

Traduction : **Frédéric Handle**

LIENS

- [1] **Code de `tasknfy.ino`** : https://github.com/ve3wwg/FreeRTOS_for_ESP32/blob/master/tasknfy1/tasknfy1.ino
- [2] **Documentation de FreeRTOS** : www.freertos.org/Documentation/RTOS_book.html
- [3] **Documentation pour la notification de tâches FreeRTOS** : www.freertos.org/RTOS-task-notification-API.html

oscillateur HP 10811

Un drôle de composant drôlement précis

Neil Gruending

Pour mesurer le temps avec précision, il faut, comme pour mesurer une tension, une solide référence de comparaison. Le plus souvent un oscillateur à quartz est un bon point de départ, mais il faut le soigner aux petits oignons pour en faire une référence vraiment précise. Examinons l'une des solutions de Hewlett-Packard pour cela, l'oscillateur 10811.

Au fil du temps, la fréquence d'un oscillateur à quartz peut fluctuer ou changer pour de bon. Les fluctuations à court terme sont généralement dues à la modulation de fréquence ou de phase. Une dérive à plus long terme peut résulter de changements de température et du vieillissement des composants. Si vous utilisez un quartz pour l'horloge d'un processeur, ces effets-là sont bénins, sauf si vous voulez faire des mesures de fréquence répétées avec une précision de 0,001 Hz. Dans ce cas, rien ne doit être négligé.

Le comportement de la fréquence de résonance du quartz en fonction de la température dépend de l'angle de coupe du cristal. Cet angle de taille est tel que la fréquence de résonance soit la plus stable possible en température. La dérive de fréquence d'un quartz est la plus faible à sa température nominale. Pour un quartz de taille AT classique, c'est $\sim 25^\circ\text{C}$. La dérive est exponentielle par rapport à la température. Pour une référence de fréquence, ça ne passe pas. C'est pourquoi le HP 10811 utilise une taille dite SC, une spécialité qui réduit l'erreur de fréquence. La stabilisation de la température du quartz près de sa valeur de consigne permet d'augmenter la précision. C'est ainsi que le quartz du HP 10811 est dans un four (Oven Controlled Crystal Oscillator) thermostaté (fig. 1). Ces caractéristiques, combinées à la conception d'un circuit oscillateur dont le bruit et les erreurs de phase sont réduites au minimum, confèrent au 10811 son excellente précision.

Ce composant est intéressant aussi comme exemplaire précoce de circuit imprimé flexible. Celui-ci est plié pour tenir dans son

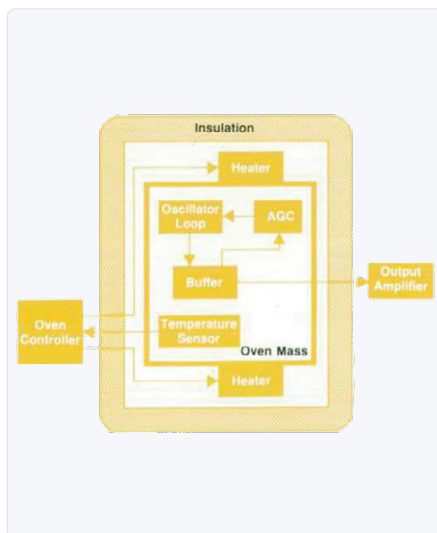


Figure 1 : Schéma de l'oscillateur HP 10811 [1].

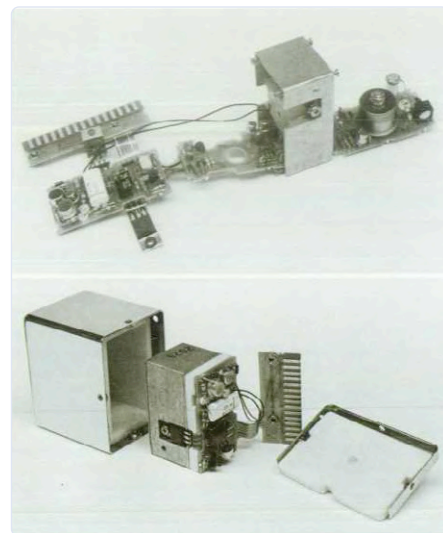


Figure 2 : Photos de l'oscillateur HP 10811 démonté [1].

boîtier (fig. 2). Sur la photo, on voit l'horloge et le circuit de commande automatique de gain pliés et fixés de chaque côté de la structure métallique. Le circuit de commande du four est replié sur un isolant thermique tandis que les éléments chauffants TO220 sont fixés à la structure (photo du bas). Une fois l'opération terminée, tout est placé dans le boîtier isolé thermiquement et scellé.

Le HP 10811 existe en plusieurs variantes

qu'on trouve sur des appareils de mesure nécessitant une référence de temps précise, comme le fréquencesmètre HP 5334A et des références de fréquence aujourd'hui reléguées par le GPS, comme le HP Z3801A. Le 10811 n'augmente pas seulement leur précision, mais sa stabilité signifie que des mesures à plus long terme, comme la variance d'Allan [2], deviennent possibles, même dans votre labo ! ◀

190383-F-04

Contributeurs

Idee, texte et images : Neil Gruending

Maquette : Giel Dols

Rédaction : Stuart Cording

Traduction : Lucien Javelle

LIENS

[1] Hewlett-Packard Journal - mars 1981 :

<https://www.hpl.hp.com/hpjournals/pdfs/IssuePDFs/1981-03.pdf>

[2] Écart-type d'Allan : https://fr.wikipedia.org/wiki/Variance_d%27Allan

[3] Taille de quartz : [https://fr.wikipedia.org/wiki/Quartz_\(%C3%A9lectronique\)](https://fr.wikipedia.org/wiki/Quartz_(%C3%A9lectronique))

balise GPS LoRa

Matériel et logiciel libres et ouverts



Quel bel instrument qu'une balise GPS ! Elle donne sa position à intervalles réguliers dont vous réglez la durée à votre guise ! Avec ce projet proposé par Elektor, la position est communiquée par Node-RED, en passant par le réseau LoRa, à un PC, un RPi, un téléphone portable, une tablette ou un autre appareil. Celui-ci affiche la position instantanée de l'instrument sur une carte ouverte dans un navigateur. Magellan aurait apprécié.

INFOS SUR LE PROJET

Mots clés

LoRa, RFM95, STM32, Raspberry Pi, Arduino, GPS, enseignement, OpenHardware, OpenStreetMap

Niveau

débutant – connaisseur – expert

Temps

1 h environ pour l'assemblage complet (coffret et antenne compris)

Outils

fer à souder, tournevis, perceuse

Coût

90 à 100 € environ

Beaucoup d'entre nous aiment savoir si un objet se trouve toujours à l'endroit où on l'a laissé : la voiture par exemple, des engins de chantier, ou un animal domestique. On s'est donc mis à équiper ces objets (pardon pour les animaux...) de balises GSM-GPS telles que ElekTrack [1]. Leur propriétaire peut ainsi s'assurer de leur position ou être prévenu si celle-ci changeait. Pour cela il fallait un module de téléphone mobile et une couverture réseau pas forcément disponible. Un tel suivi n'est pas gratuit. Les redevances dépassent quelquefois le raisonnable, comme certains chercheurs russes en ont fait l'amère expérience [2]. Comme alternative, il y a le LoRaWAN, un réseau dans la bande ISM des 868 MHz, sans licence, et dont le point fort est une grande portée pour une faible puissance d'émission.

L'un de ces réseaux est *The Things Network*. Les passerelles, c'est-à-dire les points d'accès, sont gérées par une communauté à laquelle chacun peut se joindre avec sa propre passerelle et contribuer ainsi à étendre le réseau. Le transport de données dans ce réseau est gratuit, l'idéal pour entrer dans le monde des capteurs en réseau. Si, dans votre environnement, il ne se trouve pas de passerelle, vous pouvez installer la vôtre à peu de frais.

Nœud LoRa

La liste des ingrédients pour une balise est assez courte. Comme pour la bonne cuisine, tout est dans l'art de bien les accommoder. Dans ce cas, il aura fallu quelques semaines de sang et de larmes. Au fil de la conception, nous sommes passés de la simple

balise LoRa-GPS initiale au nœud universel LoRa-Elektor (**fig. 1**) aux utilisations multiples, dont la balise n'en est qu'une parmi d'autres. Le nœud LoRa-Elektor et tous ses avatars passés, présents et futurs sont libres et ouverts aussi bien pour le matériel que pour le logiciel.

Comme le nœud LoRa-Elektor a été décrit en détail dans le numéro de mars/avril [3], de même que l'accès à *The Things Network* [4], il vaudrait mieux avoir ce numéro à portée de main. Dans la description du matériel (**fig. 2**), nous pourrions ainsi nous concentrer sur l'essentiel et utiliser la place disponible pour nous occuper du module GPS, du logiciel GPS, de l'intégration dans *The Things Network*, du transport de données à travers

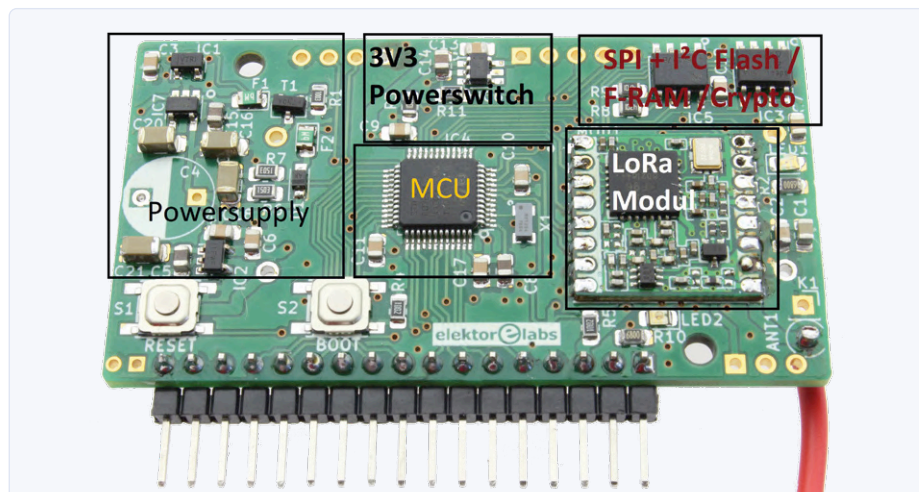


Figure 1. Le circuit équipé du nœud LoRa Elektor

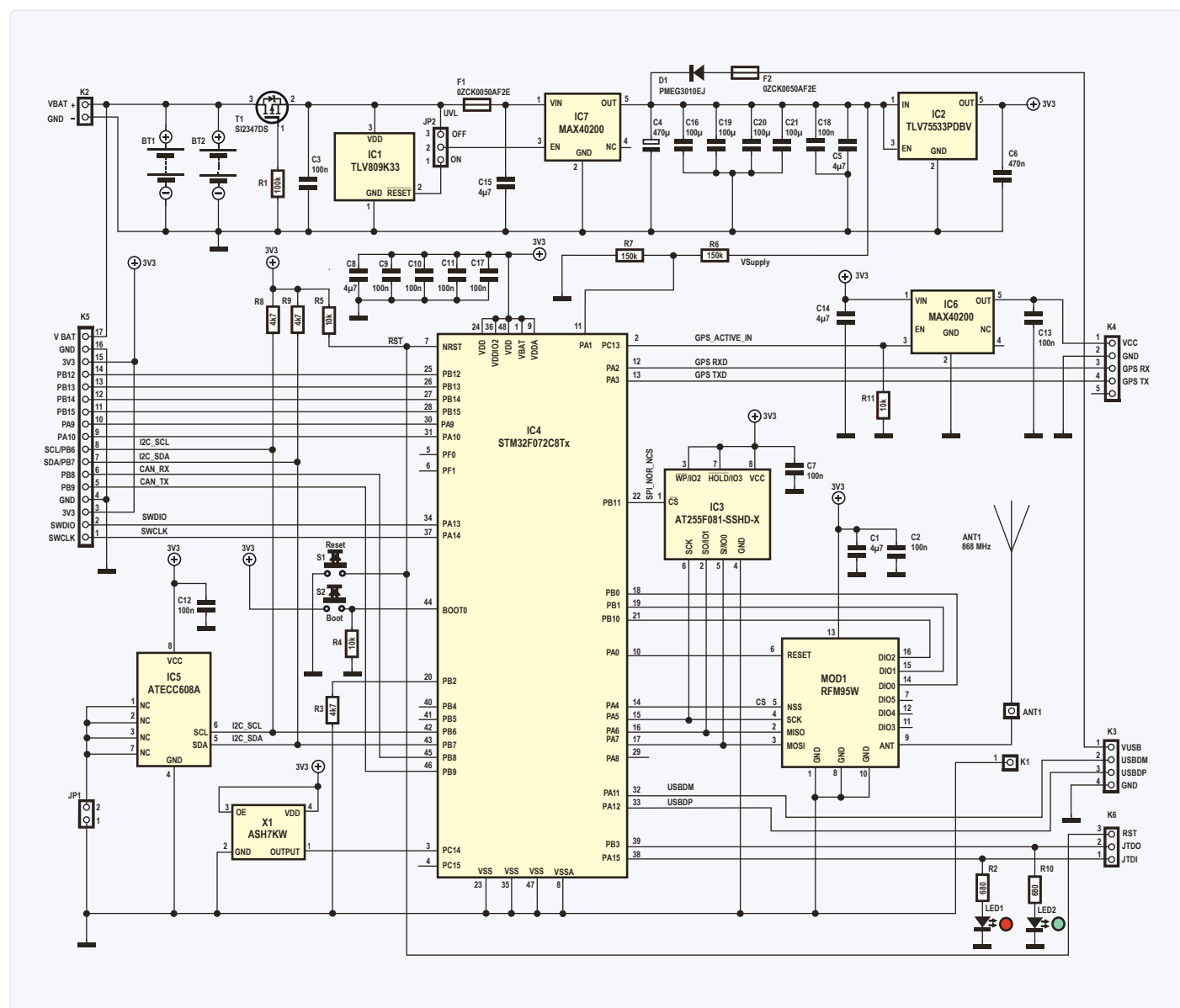


Figure 2. Schéma du nœud LoRa.

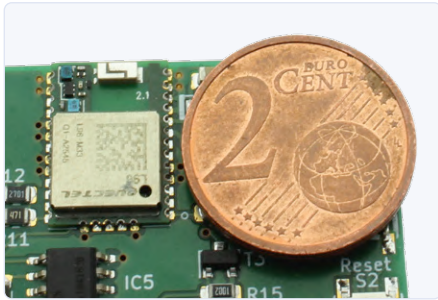


Figure. 3. Le minuscule module GPS L96.

ce réseau, ainsi que de MQTT, de Node-RED et de l'affichage sur une carte géographique.

Coffret

Pour le coffret, qui doit contenir la batterie, la carte mère et un module GPS, nous avons retenu un modèle de la série 1551 de Hammond, avec l'indice de protection IP54 (étanche à la poussière et aux projections d'eau), propre à l'usage en extérieur.

Microcontrôleur

Le microcontrôleur utilisé est un STM32F072C8T avec 64 Ko de mémoire flash et 16 Ko de mémoire RAM. La famille des STM32 comporte beaucoup de puces en boîtier LQFP48 aux broches compatibles, permettant d'utiliser avec la même carte un microcontrôleur avec davantage de mémoire flash ou à plus faible consommation. Le développement du micrologiciel peut se faire classiquement en C/C++ ou avec l'EDI Arduino (grâce au projet STM32duino). Grâce au chargeur (non

réinscriptible) de la puce, il suffit d'un simple convertisseur USB-série pour télécharger le micrologiciel.

Le transmetteur LoRa

Sur la carte se trouve le transmetteur LoRa, un RFM95W, qui travaille dans la bande ISM 868 MHz.

Alimentation

L'alimentation est fournie par une batterie Li-ion bon marché, rechargeable, utilisée sur les drones. Des cellules AAA au lithium pourraient convenir mais exigeraient un coffret plus grand.

Nous avons délibérément renoncé à un circuit de charge, la batterie doit donc être rechargée par un chargeur externe approprié. Nous n'avons pas renoncé à une détection de tension basse pour protéger la batterie d'une décharge profonde ni à une protection contre une inversion de courant intempestive, sous la forme d'une «diode idéale» IC7, pour que la batterie ne risque pas d'être surchargée, de fuir, d'exploser ou de prendre feu. Le circuit de protection est protégé directement au niveau de la batterie par un fusible de 500 mA pour éviter une destruction des pistes du PCB en cas de court-circuit. Ce circuit a l'avantage de permettre, moyennant une diode de plus, l'alimentation directe sous 5 V, avec la batterie en secours.

Mémoire flash et module de cryptage I²C/SPI

La carte peut recevoir en option une mémoire flash (IC3) et un module de cryptage (IC5) connectés par I²C/SPI au contrôleur. Ces deux

puces ne sont pas nécessaires pour la balise GPS, on peut donc les ignorer.

Régulateur 3,3 V

Le régulateur 3,3 V IC2 est assez puissant pour alimenter à la fois le nœud LoRa et le module GPS.

Module GPS

Il nous a fallu un peu de temps pour choisir et mettre en œuvre le module GPS. Au départ, nous avions prévu le Quectel L96, un module minuscule (**fig. 3**), à monter directement sur la carte mère. Dans une note d'application du L96, il était question d'un plan de masse approprié sur lequel le module devait être correctement positionné. Dans la première version du circuit imprimé, cette note avait été «oubliée», mais même dans la deuxième version, où elle avait été scrupuleusement respectée, la réception GPS par la minuscule antenne du module (le composant au-dessus de l'inscription 2.1) était si médiocre qu'il fallait un temps considérable pour obtenir la localisation, avec un effet néfaste sur la consommation d'énergie du module.

C'est ainsi que nous en sommes arrivés à la version définitive du circuit, dans laquelle le récepteur GPS n'est plus intégré à la carte mère, mais est un module séparé placé au-dessus. Pour ce module GPS, nous avons finalement choisi le GT-7U 1728 d'Open-Smart (**fig. 4**), l'un de ces composants chinois étonnamment bon marché et disponible dans notre boutique en ligne [3]. Ce mode de montage facilite l'emploi du module. Pour améliorer la réception, il est possible d'y connecter une antenne externe.

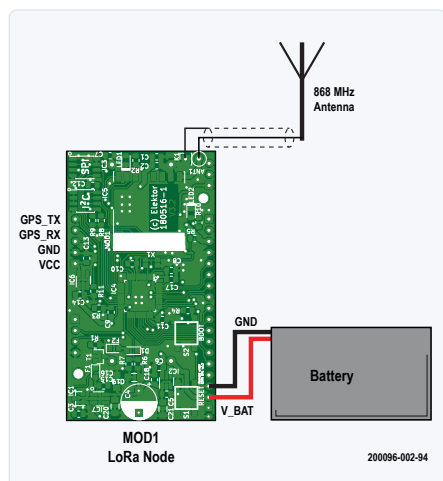


Figure. 4. Le module GPS installé sur le nœud LoRa.

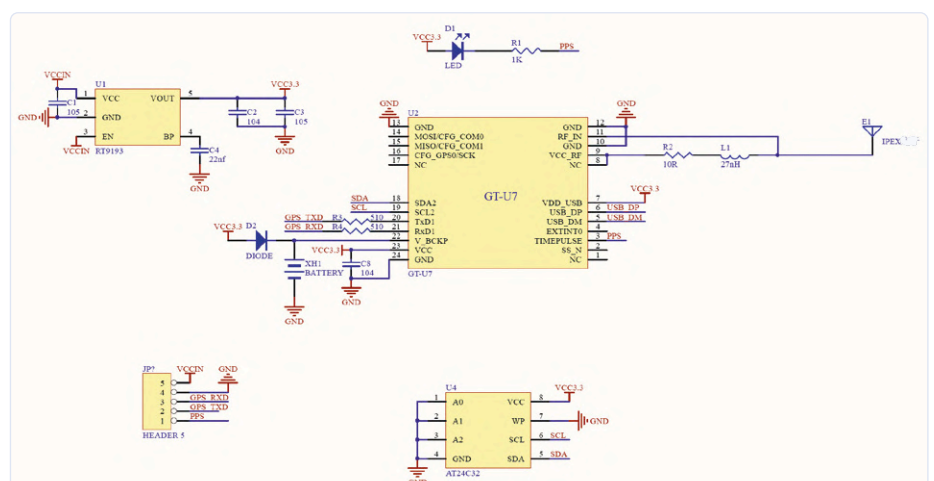


Figure. 5. Schéma interne du module GPS.

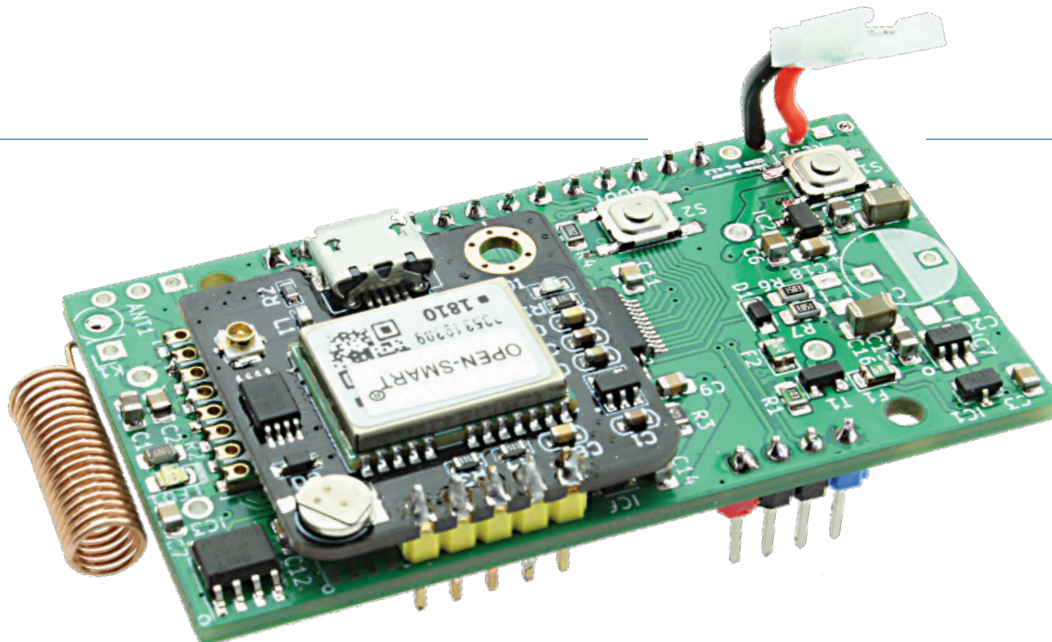


Figure. 6. Le module GPS survole la carte mère de quelques millimètres.

De plus, la carte mère peut ainsi se limiter aux fonctions nécessaires à l'ensemble des applications prévues.

Dans notre boutique en ligne, vous trouverez quelques données de base du module GPS et, caché sous *For more information click here*, un lien vers une archive avec des pilotes, un manuel utilisateur, des programmes pour le µC et l'Arduino, des outils de test et un schéma du circuit (fig. 5), qui montre que quand le module fonctionne avec le port USB, son alimentation passe aussi par ce port. La ligne VCC de l'USB est connectée directement à la broche VCC du module (qui peut supporter jusqu'à 5 V). Sans précaution spéciale, on pourrait donc avoir une connexion directe du 5 V de l'USB avec l'alimentation 3,3 V de la carte mère, ce qui endommagerait la balise en cas de connexion directe à un port USB 5 V. Même si l'USB respectait la plage de tension permise (3,0 à 3,6 V), il y aurait un conflit si le module GPS était alimenté par le nœud LoRa. Une seconde «diode idéale» IC6 permet de l'éviter. Si la tension à la sortie du MAX40200 dépasse la tension à son entrée, la «diode idéale» bloque, mais si le module est alimenté avec les 3,3 V de la carte mère, la faible tension de chute directe de cette diode (85 mV seulement) assurera au module une tension d'alimentation correcte. De plus, le µC a la possibilité, par la broche *Enable*, de couper l'alimentation du module. Dans notre application, le mieux serait d'oublier complètement ce port USB du module GPS !

Pour faire entrer le module GPS dans le coffret, il a été embroché par-dessus la carte mère (fig. 6) et survole de quelques millimètres les composants du nœud LoRa. C'est ainsi que la batterie, la carte mère et le module GPS se partagent au mieux la place dans le

coffret. Outre l'antenne GPS, couchée dans le coffret, il faut aussi tenir compte de l'antenne LoRa. Une antenne hélicoïdale miniature serait assez petite pour entrer dans le coffret, mais cela réduirait la puissance émise de 30 dBm, ce qui rendrait l'émission d'un signal vers une passerelle LoRaWAN presque impossible. C'est pourquoi une antenne extérieure est préférable, qu'il s'agisse d'un simple fil ou d'un dispositif esthétique et étanche [2].

Micrologiciel

Le micrologiciel se compose essentiellement de quatre parties (fig. 7), un décodeur pour les données GPS, la bibliothèque Arduino LMIC comme pile LoRaWAN, une bibliothèque pour la basse consommation et une interface utilisateur pour le paramétrage. Le décodage des données GPS est effectué par la bibliothèque TinyGPS++, que nos lecteurs connaissent depuis son usage dans le serveur de temps ESP32-NTP [5]. Pour que la bibliothèque puisse effectuer son travail, les données entrantes doivent être lues sur l'interface série et passées à la bibliothèque pour décodage. Pour éviter une collision entre le module GPS et l'interface série utilisée pour le téléchargement du micrologiciel, on utilise le deuxième UART qu'on initialise dans le croquis avec **HardwareSerial Serial X(USART2)** ; Dans la description de la carte, on a déjà défini les broches utilisées pour la commande du module GPS. Le µC doit alimenter la bibliothèque avec les données entrantes qu'il prélève dans le tampon de l'UART. Pour s'apercevoir que le module GPS a relevé une nouvelle position, la bibliothèque lit périodiquement la position courante et traite la réponse.

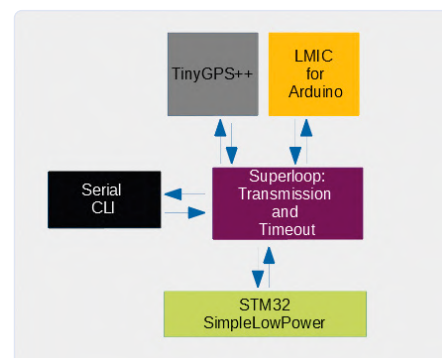


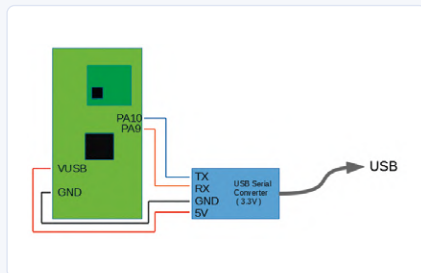
Figure. 7. Structure du micrologiciel.

Pour la communication LoRaWAN, on utilise la même bibliothèque LMIC qui a servi de pile LoRaWAN dans le LoRaSwitch et un grand nombre d'autres projets Arduino-LoRaWAN. En utilisant cette bibliothèque, on doit veiller à ce que la fonction **os_runloop_once** soit appelée aussi souvent que possible et qu'il se produise le moins possible d'interruptions de longue durée. C'est aussi l'une des raisons pour lesquelles il est si difficile de faire tourner la LMIC sur un ESP32 et de respecter les contraintes de temps.

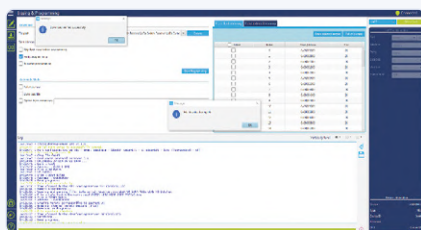
Pour le transport de données, on passe une charge utile à la bibliothèque et on attend la confirmation que les données ont bien été envoyées. Dans la plupart des croquis, on définit la configuration de la bibliothèque LoRaWAN et la plupart de ses données d'accès à la compilation ; ici elle peut être modifiée en cours d'exécution. Cette configuration est effectuée au moyen d'une ligne de commande sérielle qui peut être saisie dans la fenêtre terminal de l'EDI Arduino (fig. 10). Les éléments de base de cette ligne

TÉLÉCHARGEMENT DU MICROLOGICIEL

Tout ce matériel ne fait pas une balise LoRa. Pour cela, il faut encore «un peu» de logiciel, disponible sur la page web du projet [9] sous forme de fichier hexadécimal. Pour télécharger le logiciel, il suffit d'un convertisseur USB-série [voir l'encadré produits] et quatre fils Dupont (femelle-femelle).

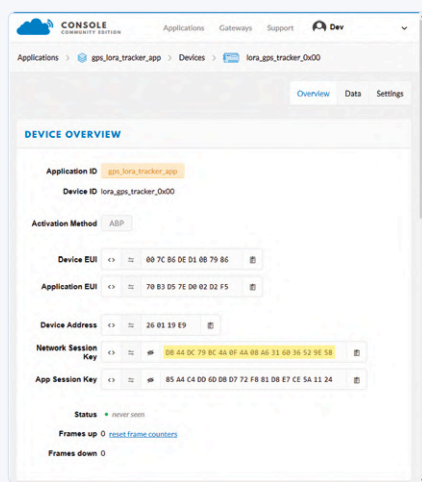


Le fichier hexadécimal peut être programmé avec le programmeur STM32-Cube. Pour cela, il faut appuyer simultanément pendant un petit moment sur les boutons *Boot* et *Reset* pour que la puce passe dans le mode *bootloader*. On peut alors choisir la bonne interface série et établir la connexion en appuyant sur *Connect*. Si aucun message d'erreur n'apparaît, la programmation du micrologiciel peut commencer.



Une fois la programmation réussie, on sort la puce du mode *bootloader* en appuyant sur *Reset*. Elle commence alors immédiatement à exécuter le code.

Si l'on ne désire pas programmer la puce avec le micrologiciel tel quel, on peut l'adapter à ses propres besoins en utilisant l'EDI Arduino, le recompiler et le télécharger. L'EDI Arduino supporte la carte LoRa Elektor avec la version 1.8.0 du projet STM32duino, sans nécessiter de modification d'un fichier quelconque. Mais attention : la version courante 1.8.0 installable au moyen du gestionnaire de cartes Arduino comporte quelques bogues qui peuvent provoquer le plantage du micrologiciel. Dans la version courante disponible sur Github du projet STM32duino, ces bogues sont corrigés.



de commande ont été repris du projet de la station de soudage à température contrôlée [6], où elle a été utilisée pour le réglage de quelques paramètres. Pour la balise LoRa-GPS, on a ajouté quelques fonctions telles que l'analyse de chaînes hexadécimales et quelques petites modifications pour faciliter l'addition de nouvelles commandes. Le code est utilisable non seulement sur le STM32, mais aussi, moyennant quelques modifications, sur les μ C ESP32 et AVR.

L'interface elle-même utilise le premier UART et on peut y accéder par les broches PA9 (RX MCU) et PA10 (TX MCU). Elle est paramétrée pour une vitesse de 115200 bauds,

8 bits de données, sans parité et un bit d'arrêt (115200Baud 8N1). Pour la fenêtre terminal de l'EDI Arduino, il suffit de régler la vitesse à 115200 bauds.

La quatrième partie est une bibliothèque minimale de gestion de la basse consommation, qui fait en sorte que le STM32 passe en sommeil au bout d'un temps prédéfini et puis se réveille. Dans ce mode, le contenu de la RAM est maintenu et l'horloge en temps réel continue de fonctionner. Le réveil est provoqué par une alarme gérée par l'horloge en temps réel. Il existe une bibliothèque bien plus riche, mais qui consomme 12 Ko de mémoire flash.

IEEE 754 float (4Byte)	IEEE 754 float (4Byte)	4 Byte Status
---------------------------	---------------------------	---------------

Figure 8. Paquet de données pour une position GPS.

Reserved Bit 31 - 16	VBat/2 Bit 15 - 8	Time until GPS-Fix Bit 7 - 1	GPS Invalid Bit 0
-------------------------	----------------------	---------------------------------	----------------------

Figure 9. Signification de l'octet d'état

Dans la description des diverses composantes, il ne faut pas oublier que pour émettre une information avec la bibliothèque LMIC, on ne dispose que de 51 octets. Plus l'information est courte, plus on dispose de temps d'activité et moins le LoRaWAN et la bande de fréquence 868 MHz sont encombrés. Dans la version courante du micrologiciel on émet 12 octets pour une position (fig. 8), ce qui n'est pas optimal et pourrait être réduit à 7 octets. La longitude est contenue dans la première valeur flottante au format IEEE754, la latitude dans la seconde. Les 4 octets suivants (fig. 9) contiennent diverses informations de statut (les données de position sont-elles valides, au bout de combien de temps la position GPS a-t-elle été déterminée, état de la batterie). On peut utiliser ces données comme informations complémentaires. Le code étant ouvert, on peut y ajouter ses propres valeurs.

Optimisation de la consommation poussée à ses limites

L'énergie fournie par la batterie étant finie, on doit porter une grande attention à l'optimisation de son utilisation, et donc couper ou mettre en sommeil tous les consommateurs inutiles. La gestion du module GPS est simple : une broche GPIO du contrôleur fait changer d'état la broche de validation de la diode idéale IC6, ce qui commute l'alimentation du module. Le module LoRa se met de lui-même en sommeil après l'émission. La feuille de caractéristiques du STM32 laisse supposer que le courant consommé serait de l'ordre de 68 μ A, mais si l'on mesure le courant du circuit sans précaution spéciale, on trouve 0,5 à 0,7 mA, dont une grande partie est consommée par le microcontrôleur. Où passe ce courant ? La réponse se trouve chez les broches GPIO non utilisées. Chaque broche non explicitement définie dans le code travaille en mode entrée flottante par défaut (voir l'encadré *Consommation électrique des broches GPIO non utilisées*). Comme mesure

correctrice, on devrait toujours activer leur résistance de rappel interne pour que ces entrées soient à un potentiel déterminé. On peut encore économiser un peu d'énergie en mettant les broches de l'UART communiquant avec le module GPS en mode entrée quand ce module est coupé, sinon la broche TX de l'ESP32 pourrait appliquer 3,3 V à la broche E/S du module GPS.

Outre le STM32, il y a d'autres composants consommateurs d'énergie, par ex. le régulateur de tension, les diodes idéales, le module LoRa et les condensateurs. Pour ne pas être plus royalistes que le roi, nous laisserons les choses en l'état. La consommation de la balise LoRa-GPS au repos s'établit alors à 150 µA à une température ambiante de 25 °C.

Quand la balise se réveille, la consommation augmente sensiblement. La plus grande part en revient au module GPS, qui a besoin de 60 mA et qui détermine l'autonomie de la balise. On comprend pourquoi il est important qu'il évalue la position de la balise le plus rapidement possible ! Le module LoRa requiert 100 mA en pointe pendant l'émission, mais celle-ci dure moins d'une seconde. À une fréquence d'horloge de 48 MHz, le µC consomme 2,5 mA et dispose d'un temps suffisant pour le traitement des données de position et leur envoi vers le LoRaWAN. Si la puissance de calcul maximum n'est pas nécessaire, on devrait réduire autant que possible la fréquence de l'horloge. L'autonomie maximale de la batterie dépend donc essentiellement de l'intervalle entre deux relevés de position et la durée (*timeout*) au bout de laquelle la tentative de relevé de la position par le module GPS est abandonnée en cas de mauvaise réception.

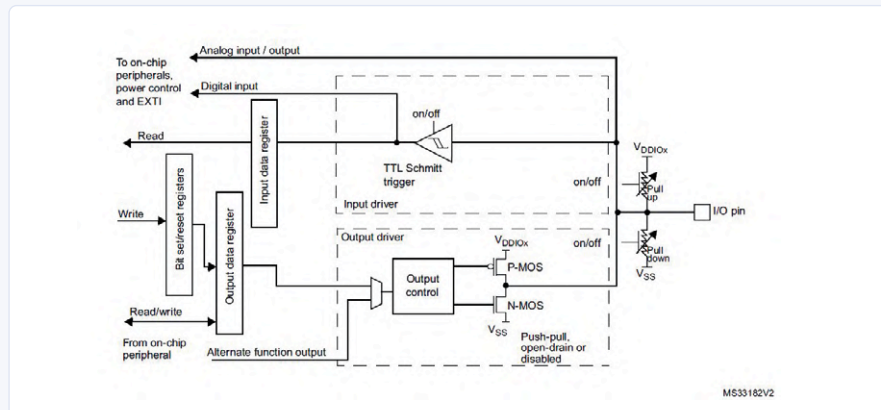
Réglage des paramètres

Pour qu'il ne soit pas nécessaire de télécharger un nouveau micrologiciel à chaque modification d'un paramètre, l'interface série utilisateur (à utiliser par ex. avec la fenêtre terminal de l'EDI Arduino) permet quelques réglages de base :

- intervalle entre deux émissions de la position GPS
- temps maximum d'une tentative de relevé de la position
- vitesse d'émission LoRa

Ces trois paramètres ont une influence sur la consommation d'énergie de la balise. L'intervalle entre deux émissions est réglé avec **set INTERVALL** et peut être lu avec

QUE CONSOMMENT LES BROCHES E/S INUTILISÉES ?



Dans le *Reference Manual RM0091* on trouve une description de la logique d'une broche GPIO. La bascule de Schmitt intégrée assure que lorsque qu'elle fonctionne comme entrée logique, la broche fournit toujours un niveau déterminé, haut ou bas selon le résultat de la comparaison de la tension lue avec des seuils fixés. Si la broche est laissée flottante, elle se comporte comme une antenne, si bien qu'il arrive une tension fluctuante sur la bascule de Schmitt, ce qui la fait fréquemment commuter entre haut et bas et entraîne une (faible) consommation de courant. Les broches non utilisées sur la carte totalisent ainsi une consommation de courant évaluée à 0,5 à 0,7 mA.

get INTERVALL. Il en est de même avec la position GPS, réglée par **set GPSTIMEOUT** et lue par **get GPSTIMEOUT**. Les valeurs initiales sont 15 mn pour l'intervalle d'émission et 10 mn avant l'abandon en cas de réception GPS mauvaise ou absente.

La vitesse de transfert de données LoRa peut être réglée dans le domaine SF7...SF10, où SF7 est la vitesse maximum et donc le temps d'émission le plus court, et SF10 une vitesse lente où la modulation a besoin de plus d'énergie par unité de temps. SF7 est pré-réglée pour une transmission rapide, mais aussi une portée limitée. Lorsqu'on augmente la valeur de SF7, cela accroît la portée, mais peut conduire à des perturbations de la transmission si la balise est en mouvement : une lame à deux tranchants ! Cela réduit aussi le nombre de positions transmissibles par jour. *The Things Network* prévoit un temps d'émission quotidien de 30 secondes ; avec SF7, il faut compter une durée d'émission de 62 ms et 412 ms avec SF10, ce qui représente par jour 450 envois pour SF7 et 70 envois pour SF10.

Pour pouvoir transmettre des données par LoRaWAN, il faut inclure les clés du destinataire final. La balise LoRa-GPS utilise le mode ABP. Nous en arrivons ainsi à la configuration dans *The Things Network*.

Configuration dans The Things Network

L'accès au réseau *The Things Network* a déjà été traité en détail dans le numéro de mars/avril [4], nous pouvons donc nous contenter de l'effleurer ici. Après avoir créé un compte utilisateur et s'être connecté, il faut commencer par cliquer sur *Applications* puis sur *Add application* pour créer sur la page suivante une application à associer à la balise. Dans le dialogue suivant, il faut saisir un nom unique (*gps_lora_tracker_app*) et une description (*Test Sensor Nodes for Developement*) de l'application.

Les appareils LoRa sont toujours associés à une application pour que les données incidentes puissent être immédiatement décodées et traitées de manière appropriée dans *The Things Network*. Pour la balise LoRa-GPS ceci ne joue tout d'abord aucun rôle, le traitement des données étant effectué par un serveur (un RPi) avec Node-RED. L'application créée doit encore être associée à un appareil, en l'occurrence, notre balise LoRa-GPS. Avec *Register device*, on accède à un dialogue qui permet de saisir les données de la balise. Pour chaque balise, il faut indiquer un nom unique et terminer le dialogue avec *Register*. Dans l'appareil ainsi créé, sous *Settings*, il faut changer la méthode d'acti-

BROCHES À FONCTIONS VARIABLES : QUE DU PLAISIR !

Les broches à fonctions variables (*Alternate Pin Functions*) ne sont apparues que tout récemment dans le monde des AVR, alors que chez le STM32, on les connaît depuis pas mal de temps. Si pour l'Atmega128, on devait encore décider si une broche était une GPIO ou remplissait une fonction particulière telle que SPI, chez le STM32 une broche peut changer de fonction. On peut ainsi utiliser pour l'UART0 les broches PA8 et PA9 ou bien les broches PB6 et PB7. Pour toutes les broches à fonctions variables, on devrait décider très tôt, même si elles ne sont pas utilisées tout de suite, de les rendre accessibles en les routant vers l'extérieur. Il serait bien fâcheux, une fois le projet terminé, de constater au vu de la feuille de caractéristiques, qu'une broche non routée gère le signal d'horloge du deuxième port SPI qui serait bien utile. C'est pourquoi, sur entente entre les concepteurs du matériel et du logiciel, l'étude du circuit imprimé devrait rendre disponibles pour des utilisations ultérieures certaines des *Alternate Pin Functions*.

tion de OTAA à ABP pour pouvoir accéder aux données qui doivent être intégrées au micro-logiciel de la balise. Après *Save*, on peut prélever dans la vue d'ensemble les informations de réglage (*Network Session Key, Application*

Session Key et Device Address) à envoyer à la balise par l'interface série.

Pour éviter la saisie manuelle des trois clés, utilisez le presse-papiers pour les

copier depuis le *TheThingsNetwork (Device Overview)* dans le moniteur sériel. Lors de la transmission, les clés sont converties en interne et stockées dans l'EEPROM. Une fois que tous les paramètres ont été transmis, la balise est prête pour sa première utilisation.

De la balise à la carte géographique

Après sa configuration, la balise envoie vers le LoRaWAN des données qui doivent être traitées pour qu'une position puisse être affichée sur une carte géographique, par exemple. Ce traitement peut être confié à Node-RED tournant sur un RPi (ou sur une machine virtuelle, ou sur un ordinateur sur lequel il est installé). L'OS Raspberry Pi (ci-devant Raspbian) ne contient pas Node-RED. L'installation sous Raspbian (version Jessie ou supérieure) est décrite en détail sur la page web du projet [7]. Un article [8] est consacré à une introduction à Node-RED.

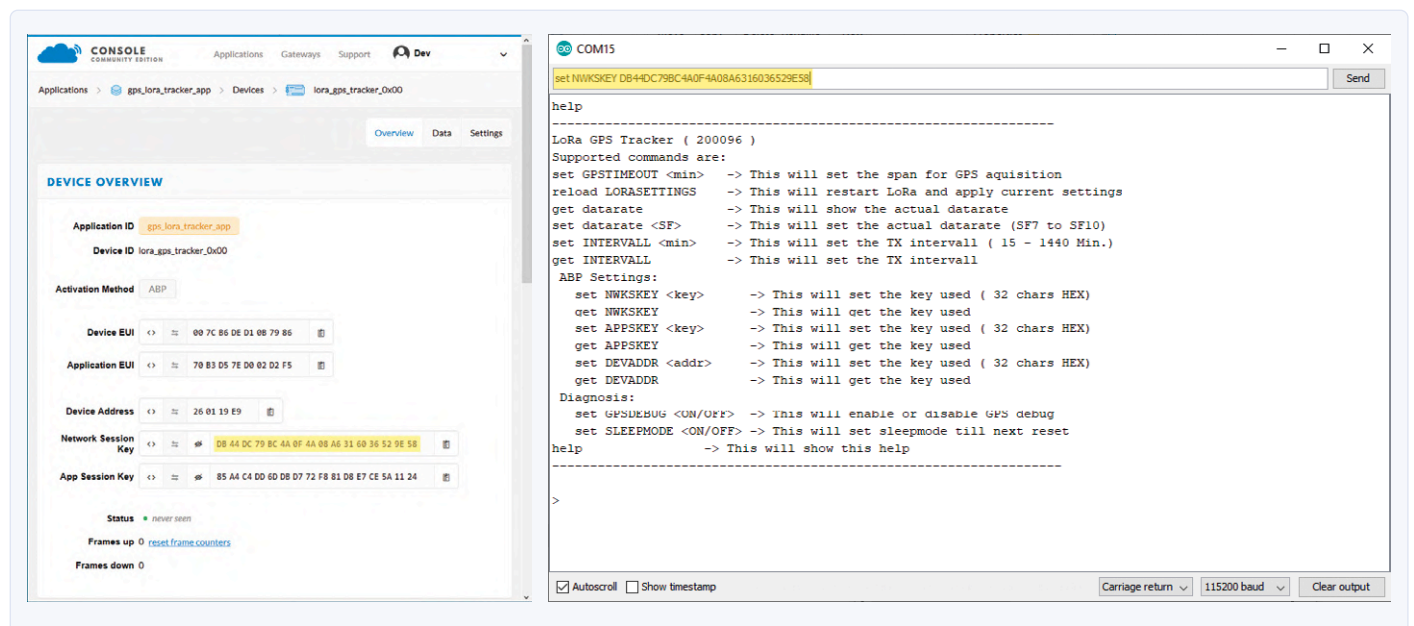


Figure 10. Copie des clés depuis la fenêtre *Device Overview* dans le moniteur sériel.

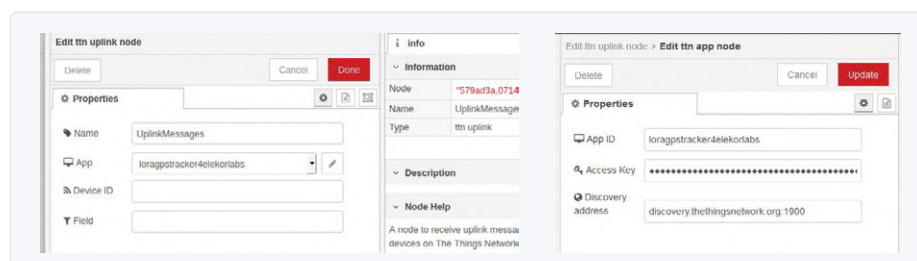


Figure 11. Transfert des données TTN vers NodeRED (*Node_Red_TTN_node_Info* et *TTN_Access*).

Votre avis, s'il vous plaît ?

Veuillez adresser vos questions et vos commentaires à l'auteur :
mathias.claussen@elektor.com

Les données sont présentement envoyées par la balise au *TheThingsNetwork*, qui ne les stocke pas, mais les envoie à un système prêt à les recevoir. Comme interface de récupération des données, on utilise le protocole MQTT qui a déjà servi dans d'autres projets Elektor. On profite ainsi de la présence dans Node-RED d'éléments déjà prévus pour le *TheThingsNetwork*, qu'il suffit de configurer au moyen de quelques paramètres, ce qui simplifie considérablement le travail. Le nœud *UplinkMessages* requiert l'établissement préalable d'une liaison avec l'application. Pour cela, on a besoin de l'*Application ID* et de l'*Access Key* de la console de *The Things Network*, ce qui permet de s'identifier comme utilisateur légitime auprès des serveurs et de recevoir les données. Ces données doivent être copiées depuis la console TTN et insérées aux emplacements correspondants du nœud *UplinkMessages* (fig. 11).

Dès que le nœud est choisi, on ouvre un nouveau dialogue en cliquant sur l'icône crayon à côté de *App*. On y dépose les données de la console TTN par copier-coller, on valide avec *Update* et on termine avec *Done*. Comme pour toutes les modifications, un *Deploy* est nécessaire. Le nœud devrait maintenant afficher un *Connected* dans sa vue d'ensemble (fig. 12).

On reçoit ainsi du nœud de nouvelles données à traiter. Les informations ne contiennent pas seulement les données utiles envoyées par la balise, mais d'autres données comme les passerelles qui ont reçu l'information. À la configuration, l'utilisateur peut préciser l'emplacement des passerelles, ce qui va bientôt se montrer très utile.

Si la balise n'a pas été en mesure de relever une position exacte, elle envoie quand même des données toutefois sans aucune position GPS valide. On peut donc essayer de déterminer au moins la position de la passerelle qui a reçu les données. On ne peut pas en déduire la position exacte de la balise, mais avec les réglages en cours on obtient une zone circulaire d'un rayon de 2 km environ, dans laquelle la balise doit se trouver (fig. 13). L'affichage lui-même est effectué par une extension de World-Map, une carte interactive qu'on peut afficher dans un navigateur. Dans l'exemple Node-RED joint, on peut y accéder par `!8883/worldmap` (fig. 14).

Les données cartographiques proviennent du projet OpenStreetMap. Cette carte accepte des données de position d'un objet qu'elle affiche sous la forme d'une icône, d'un rayon

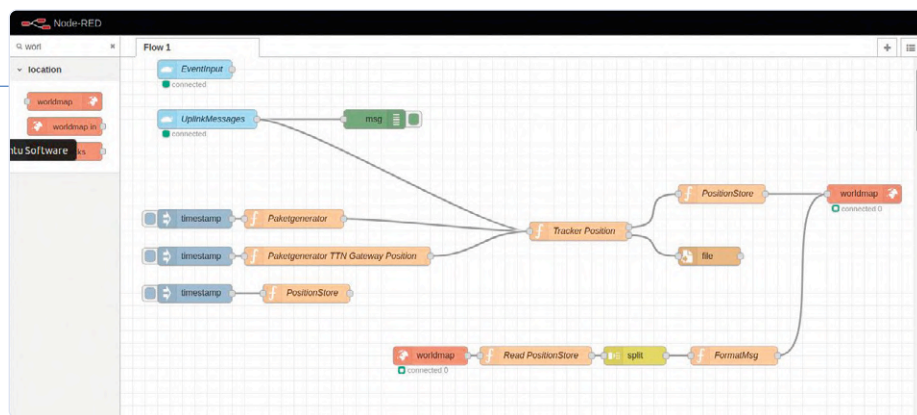


Figure 12. Organigramme de Node-RED.

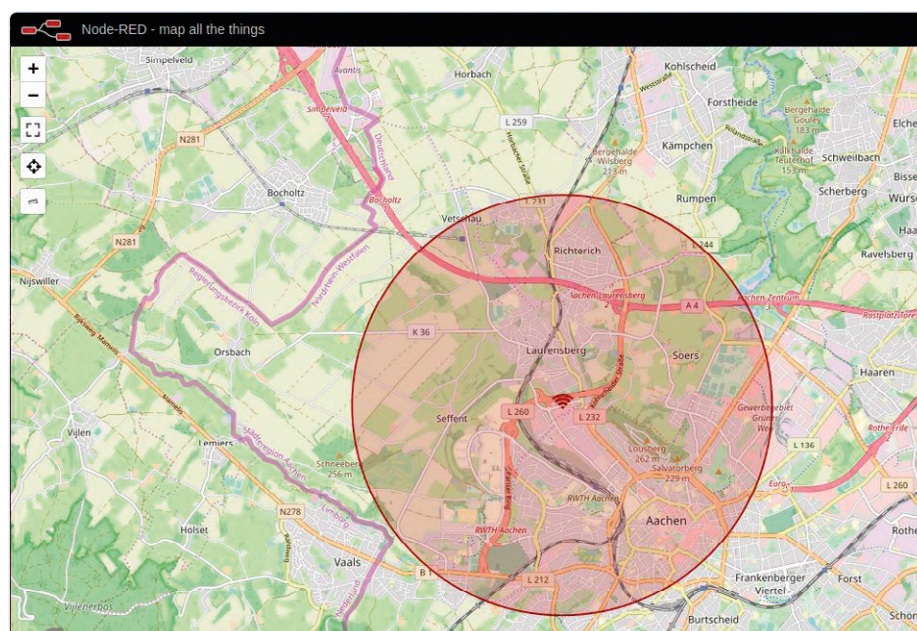


Figure 13. Il est possible de localiser la balise même sans réception GPS (© OpenStreetMap contributors).

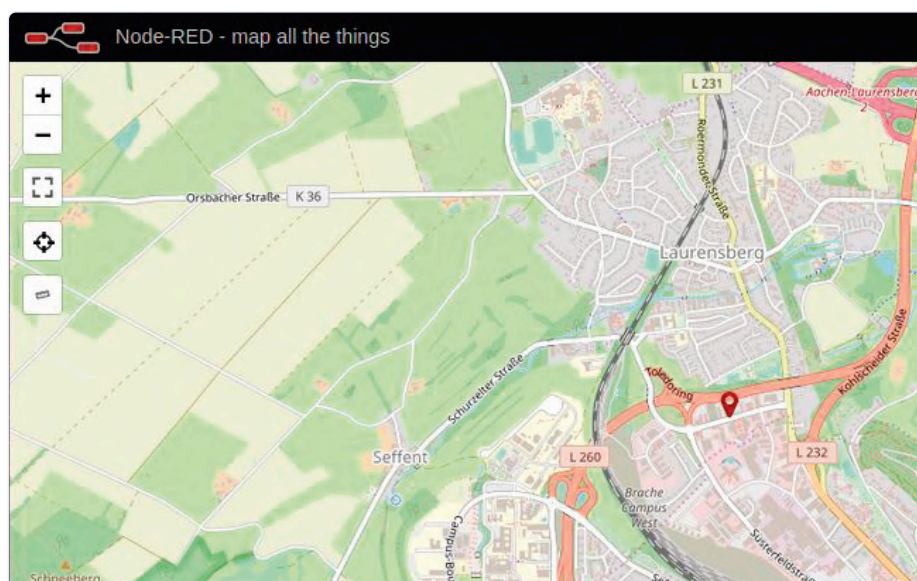


Figure 14. La balise dévoile la position exacte du quartier général d'Elektor. Vous savez maintenant où nous trouver ! (© OSM contributors).



MODULES REQUIS

Elektor LoRa Node (circuit imprimé disponible chez Elektor*, liste des composants et fichiers KiCAD, micrologiciel sous [10])
RFM95 Ultra LoRa Transceiver Module (868/915 MHz) *

Antenne SMA 868-MHz
Coffret Hammond 1551K
Câble batterie avec connecteur 2 mm RM (Molex 51005)
Batterie (voir texte)

OPEN-SMART GPS – Serial GPS Module *
Câble SMA avec Pigtail et picots

* cf. encadré ci-dessous


ou de quelque autre manière. Il y a un inconvénient : ces données ne survivent pas à la fermeture du navigateur. C'est pourquoi les derniers 2048 points de données soumis à la

carte sont stockés dans la RAM du serveur Node-RED. Lorsque la carte est affichée à nouveau, on y retrouve ces points. Mais pour que les itinéraires et les changements

de position pendant un temps assez long restent intelligibles, ils sont stockés, pour chaque balise, dans un fichier CSV compatible Excel. L'affichage sur la World-Map n'est donc pas une solution complète et définitive pour la représentation des données de position, elle est plutôt un moyen de visualiser nettement le flux de données de la balise vers l'utilisateur et de vous servir de base pour le développement d'idées personnelles.

Conclusion

LoRaWAN offre une plateforme intéressante pour suivre la position d'objets. Avec la solution proposée ici, on peut déterminer la position d'objets, la transmettre par LoRaWAN et l'afficher sur son propre ordinateur. Le caractère ouvert du matériel et du logiciel de cette solution permet à chacun de jeter un regard dans les coulisses et de s'en servir pour ses propres besoins.

Les données complètes de la balise LoRa-GPS sont disponibles sur GitHub [10] en tant que projet KiCad et peuvent être utilisées sous la licence OSHL V1.2. On trouvera aussi le logiciel sur GitHub, ainsi que le projet Node-RED d'affichage de la position. Après son utilisation comme balise, nous présenterons dans le futur (hélas indéterminé pour cause de Covid-19) d'autres applications du nœud LoRa Elektor. 

200096-02



Dans l'e-shoppe d'Elektor

- > **Elektor LoRa Node circuit imprimé nu 180516-1**
www.elektor.fr/180516-1
- > **RFM95 Ultra LoRa Transceiver Module (868/915 MHz)**
www.elektor.fr/18715
- > **OPEN-SMART GPS – Serial GPS Module**
www.elektor.fr/18733
- > **USB to TTL Converter UART Module CH340G (3.3 V/5.5 V)**
www.elektor.fr/19151
- > **Livre «Programming with Node-RED»**
www.elektor.fr/19224
www.elektor.fr/19225

Ont contribué à cet article :

Conception, texte et figures :
Mathias Claussen

Rédaction : **Rolf Gerstendorf**
Traduction : **Helmut Müller**

Maquette : **Giel Dols**

LIENS

- [1] **ElekTrack** : <http://www.elektormagazine.fr/magazine/elektor-200710/10840>
- [2] **Itinérance ruineuse** : <https://www.welt.de/vermischtes/article202525470/Roaming-Gebuehren-Als-die-Adler-in-den-Iran-flogen-wurde-es-teuer-fuer-die-Forscher.html>
- [3] **Noeud LoRa (Elektor mars-avril 2020)** : <https://www.elektormagazine.fr/magazine/elektor-142/57184>
- [4] **LoRaWAN – décollage facile (Elektor mars-avril/2020)** : <https://www.elektormagazine.fr/magazine/elektor-142/57188>
- [5] **ESP32 comme serveur de temps** : <https://www.elektormagazine.fr/magazine/elektor-101/50947>
- [6] **Poste de soudage de CMS compact** : <http://www.elektormagazine.fr/magazine/elektor-71/42314>
- [7] **Node-RED Getting Started** : <https://nodered.org/docs/getting-started/raspberrypi>
- [8] **Prise en main de Node-RED** : <http://www.elektormagazine.fr/magazine/elektor-152/58792>
- [9] **Page du projet sur Elektor Labs** : <https://www.elektormagazine.fr/200096-01>
- [10] **Fichiers KiCad sur GitHub** : https://github.com/ElektorLabs/180516-Elektor_LoRa_Node

je programme avec des automates finis

en assembleur et en C sur des PIC à 8 bits

Andrew Pratt (Royaume-Uni)

Saviez-vous qu'il existait une autre façon de programmer les PIC ? C'est ce qu'explique le livre *Programming the Finite State Machine — with 8-Bit PICs in Assembly and C* en adoptant une approche pratique. Les outils de programmation, dont l'assembleur gpasm pour PIC, peuvent être installés sur Windows ou sur Linux. Les extraits reproduits ici sont tirés du deuxième chapitre intitulé *The Assembly Program as a Finite State Machine*.

Un clignotant à LED plus complexe

L'assembleur a la réputation d'être difficile à lire. Il est vrai que même le code que l'on vient juste d'écrire semble parfois abscons. Outre la mise en forme elle-même de ce code, le diagramme d'état favorise aussi la lisibilité d'un programme en fournissant une vue d'ensemble de son déroulement. États et transitions entre états y sont représentés au moyen de blocs et de flèches. Un tel diagramme facilite la lecture du code assembleur.

Pour vous en convaincre, étudiez la **figure 2-5**. Le diagramme est à nouveau celui d'une LED, mais cette fois-ci sa séquence de clignotement est plus complexe [que celle de la section précédente du livre] : la LED clignote lentement 3 fois, clignote ensuite rapidement 10 fois, puis revient au cycle lent. Notez que le diagramme a quatre états et six transitions. Les quatre états sont les deux états **on** et **off** pour le cycle rapide, et les deux états **on** et **off** pour le cycle lent. Le diagramme peut être divisé en quatre parties au moyen de deux lignes : une séparant les états **on** et **off**, l'autre séparant les cycles lent et rapide.

La **figure 2-6** montre l'oscillogramme de la tension aux bornes de la LED.

De la compréhension du diagramme d'état devrait maintenant découler celle du programme (**listage 1**). Nous n'avons pas encore rencontré le signe **\$**. Il représente l'adresse de la ligne actuelle, ce qui permet de brancher le programme sur une certaine ligne sans utiliser d'étiquette. Par exemple **GOTO \$+3** effectue un saut en avant de 3 lignes.

Exécuter plus de deux automates dans un programme

Nous allons voir ici une façon d'exécuter simultanément plus d'un automate fini dans un même programme. La méthode ressemblera à celle que nous adoptons lorsque nous prétendons faire deux choses à la fois, alors qu'en réalité nous procédons par multiplexage temporel : nous effectuons une petite partie de la première tâche, passons à la seconde, revenons à la première, et ainsi de suite. Certaines unités distinctes du PIC telles que les temporisateurs (*timers*) peuvent effectivement fonctionner simultanément, mais le processeur du microcontrôleur ne peut traiter qu'une seule instruction à la fois. Si un programme requiert plus d'une tâche, celles-ci devront donc être exécutées dans autant de fils,

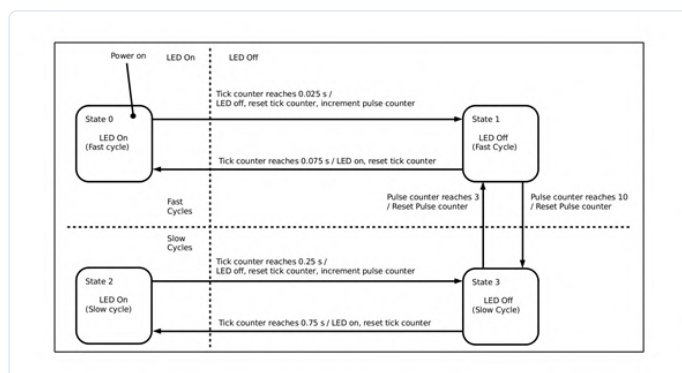


Figure 2-5. Diagramme d'état du programme de clignotement de LED de cycle.

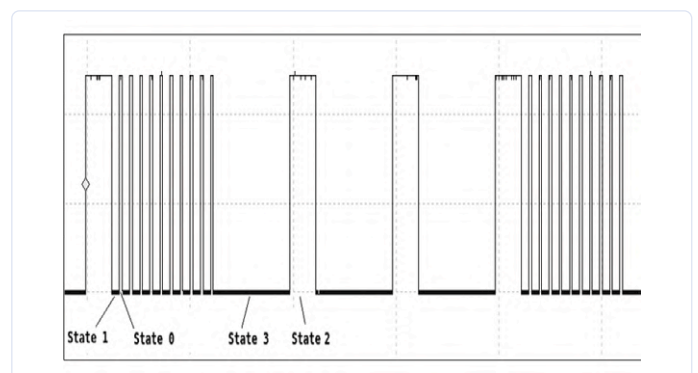


Figure 2-6. Oscillogramme du programme de cycle.

Listage 1. Programme Cycle de Cycle (prog_02_02.asm).

```
; prog_02_02.asm
; Page numbers refer to the data sheet DS40001413E.
; Tables refer to the book.

LIST      P=12f1822
#include <p12f1822.inc>

RADIX DEC                ; Default numbers are to base 10.

__CONFIG 0X8007, (_FOSC_INTOSC & _WDTE_OFF & _PWRTE_OFF & _CP_OFF & _BOREN_OFF & _IESO_OFF & _FCMEN_
OFF )
__CONFIG 0X8008, (_LVP_ON)

CBLOCK 0x70              ; In common RAM, accessible in all banks.
TICK_COUNTER
PULSE_COUNTER
ENDC

ORG 0X00
GOTO START

ORG 0X04                 ; Interrupt vector.
BCF INTCON, TMR0IF       ; Clear the tmr0 overflow interrupt flag. INTCON is in all banks.
INCF TICK_COUNTER, F     ; Increment by one counter_off.
RETFIE                  ; Return from interrupt

START
MOVLB 1                 ; To access TRISA, OSCCON, OPTION_REG.
BCF TRISA, 2            ; Configure PORTA bit 2 (chip pin 5) as an output
MOVLW 0x70
MOVWF OSCCON            ; Page 65 32MHz. See Table 2-1 in the book.
MOVLW 0xD7
MOVWF OPTION_REG       ; Page 164 in the data sheet and Table 2-2 in the book.
MOVLW 0xE0
MOVWF INTCON           ; Page 86 in the data sheet and Table 2-3 in the book.
MOVLB 0                ; To access PORTA for the rest of the program.
BSF PORTA, 2           ; Initiaise LED on;
;-----Machine States.
S0                      ; LED on. Fast cycle.
MOVLW 3
SUBWF TICK_COUNTER, W   ; C in STATUS is set when TICK_COUNTER >= to 3.
BTFSS STATUS, C         ; Skip the next instruction if C in STATUS is found set.
GOTO S0                 ; C in STATUS found not set. Stay in this state.
BCF PORTA, 2           ; Turn LED off
CLRF TICK_COUNTER      ; Reset the tick counter.
INCF PULSE_COUNTER, F   ; Add 1 to PULSE_COUNTER and put the result in PULSE_COUNTER.
                        ; Continue to S1

S1                      ; LED off. Fast cycle.
MOVLW 10
SUBWF PULSE_COUNTER, W   ; C in STATUS is set when PULSE_COUNTER >= to 10.
BTFSS STATUS, C         ; Skip the next instruction if C in STATUS is found set.
GOTO $+3               ; Have not reached 10 pulses jump to ticks check.
CLRF PULSE_COUNTER      ; 10 Pulses have been counted, reset PULSE_COUNTER.
GOTO S3                 ; Transition to S3.
MOVLW 9
SUBWF TICK_COUNTER, W   ; C in STATUS is set when TICK_COUNTER >= to 9.
BTFSS STATUS, C         ; Skip the next instruction if C in STATUS is found set.
GOTO S1                 ; C in STATUS found set, 9 ticks counted.
CLRF TICK_COUNTER
BSF PORTA, 2           ; Turn LED on.
GOTO S0

S2
MOVLW 31
SUBWF TICK_COUNTER, W   ; C in STATUS is set when TICK_COUNTER >= to 3.
BTFSS STATUS, C         ; Skip the next instruction if C in STATUS is found set.
GOTO S2                 ; C in STATUS found not set.
BCF PORTA, 2           ; Turn LED off
```



```

CLRf TICK_COUNTER          ; Reset the tick counter.
INCF PULSE_COUNTER, F      ; Add 1 to PULSE_COUNTER and put the result in PULSE_COUNTER.
                           ; Continue to S3

S3                           ; LED on. Slow cycle.
MOVLW 3
SUBWF PULSE_COUNTER, W      ; C in STATUS is set when PULSE_COUNTER >= to 3.
BTFSS STATUS, C             ; Skip the next instruction if C in STATUS is found set.
GOTO $+3                    ; ave not reached 3 pulses jump to ticks check.
CLRf PULSE_COUNTER          ; Reset PULSE_COUNTER.
GOTO S1                     ; Transition to S1.
MOVLW 91
SUBWF TICK_COUNTER, W       ; C in STATUS is set when TICK_COUNTER >= to 9.
BTFSS STATUS, C             ; Skip the next instruction if C in STATUS is found set.
GOTO S3                     ; C in STATUS found set.
CLRf TICK_COUNTER
BSF PORTA, 2                ; Turn LED on.
GOTO S2

END

```

chaque fil d'exécution partageant le « temps » du processeur. Pour obtenir ce *multi-threading*, le premier automate exécutera le bloc de code associé à son état en cours, puis ce sera au tour du second, et le programme reviendra au premier. Ces automates sont indépendants, ils ne font qu'utiliser le même processeur en temps partagé. Ils peuvent aussi partager la mémoire pour accéder aux données de l'autre.

Jusqu'à présent, l'état d'un automate était défini par le bloc de code dans lequel il se trouvait. Cet état sera maintenant défini par la

valeur stockée dans un registre. Appelons **STATE_M0** le registre contenant l'état de l'automate 0, et **STATE_M1** celui de l'automate 1 (**M** pour « machine »). La **figure 2-7** montre le flux du programme. Nous utiliserons l'instruction **BRW** pour contrôler le passage d'un état à l'autre. **BRW** ajoute au compteur de programme la valeur contenue dans le registre W, autrement dit permet d'effectuer un saut déterminé. D'où son nom : **BR**anch with **W**.

Le programme exécute le bloc de code d'état d'un des automates, puis se branche sur l'instruction de changement d'automate

Listage 2. Programme fictif ; l'automate M0 a deux états, M1 en a trois.

```

SWITCH_M0
    MOVFW STATE_M0          ; Moves the value of STATE_M0 to W.
    BRW                     ; The program will jump from here depending on W.
    GOTO M0_S0              ; The jump will be to this line if W = 0.
    GOTO M0_S1              ; The jump will be to this line if W = 1.

SWITCH_M1
    MOVFW STATE_M0          ; Moves the value of STATE_M0 to W.
    BRW                     ; The program will jump from here depending on W.
    GOTO M1_S0              ; The jump will be to this line if W = 0.
    GOTO M1_S1              ; The jump will be to this line if W = 1.
    GOTO M1_S2              ; The jump will be to this line if W = 1.

M0_S0
    -----                ; code to decide if a change of state is needed if so change STATE_M0. GOTO
    SWITCH_M1              ; Go to the other machine's switch.

M0_S1
    -----                ; code to decide if a change of state is needed if so change STATE_M0.
    GOTO SWITCH_M1         ; Go to the other machine's switch.

M1_S0
    -----                ; code to decide if a change of state is needed if so change STATE_M1.
    GOTO SWITCH_M0         ; Go to the other machine's switch.

M1_S1
    -----                ; code to decide if a change of state is needed if so change STATE_M1.
    GOTO SWITCH_M0         ; Go to the other machine's switch.

M1_S2
    -----                ; code to decide if a change of state is needed if so change STATE_M1.
    GOTO SWITCH_M0         ; Go to the other machine's switch.

```

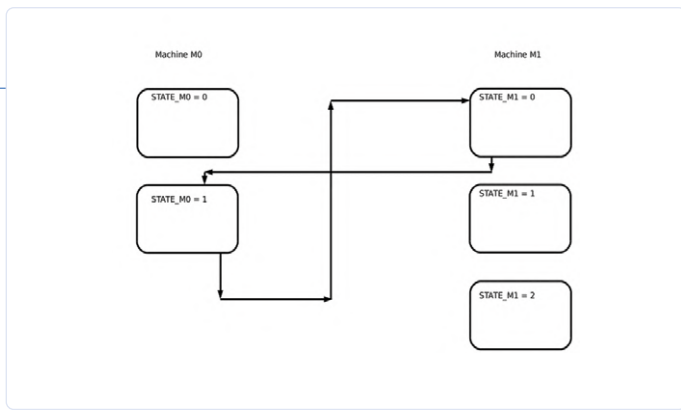


Figure 2-7. Exemple de flux de programme entre deux automates d'un même programme.

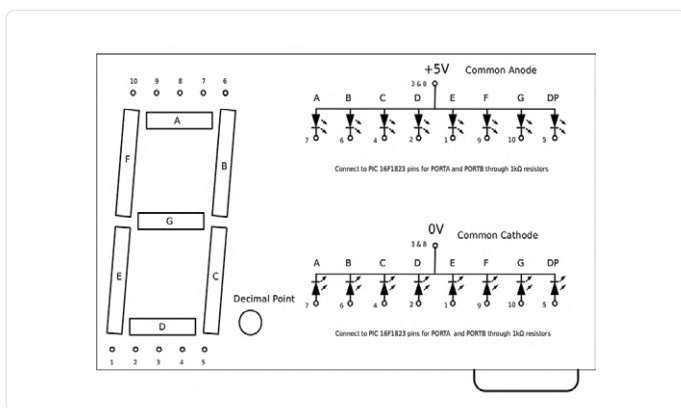


Figure 2-8. Connexions d'un afficheur à 7 segments standard.

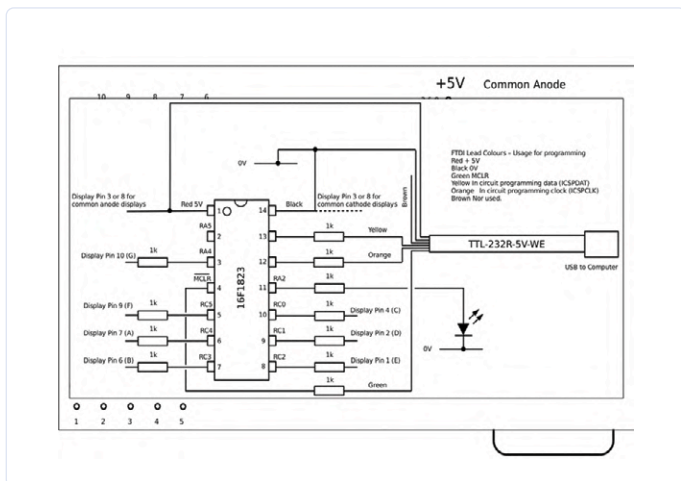


Figure 2-9. Câblage du PIC 16F1823 et de l'afficheur à LED.

(SWITCH_Mx) ; de là, l'instruction **BRW** passe à la ligne suivante si la valeur d'état est 0, saute une ligne si elle vaut 1, ou en saute deux si elle vaut 2.

Les « commutateurs » SWITCH_Mx assurent le flux continu du programme entre les blocs d'état. Pour faire tourner deux automates sur deux fils d'exécution, nous avons donc eu besoin de deux commutateurs et de deux variables d'état, STATE_M0 et STATE_M1. Référez-vous au code simplifié du **listage 2** pour comprendre le flux de ce programme fictif ; notez la façon dont il alterne entre

les automates lorsqu'il quitte un bloc de code d'état. L'automate M0 a deux états, M1 en a trois.

Pilotage d'un afficheur LED à 7 segments

Illustrons l'exploitation simultanée de deux automates finis avec l'exemple d'un programme comptant le nombre de clignotements d'une LED et affichant ce nombre sur un afficheur à 7 segments. Nous avons le choix entre un afficheur à cristaux liquides et un afficheur à LED. Les écrans à LED consomment plusieurs milliam-pères par segment mais sont plus faciles à piloter car ils peuvent l'être avec un courant continu. Les LCD requièrent quant à eux une tension alternative. J'ai donc opté pour un modèle à LED. La **figure 2-8** montre les connexions d'une unité typique. Un afficheur peut être à cathode commune ou à anode commune. Dans le premier cas la broche commune doit être reliée au 0 V, dans le second elle se connecte au +5 V. La logique de commande des LED, et par suite l'écriture du code, dépendent de cette polarité. J'ai utilisé un modèle basique à anode commune. Si vous n'en disposez pas, le LA-601VB de Rohm ou le SA52-11SRWA de Kingbright conviendront.

Les LED à piloter étant au nombre de huit, nous utilisons ici le PIC 16F1823 et son port d'E/S supplémentaire (PORTC). La **figure 2-9** montre comment relier les broches de l'afficheur à celles du PIC. Le câblage prêt, nous pouvons tracer le diagramme d'état du programme (**fig. 2-10**). **M1** est l'automate pilotant l'afficheur. La LED est initialisée à **on** à la mise sous tension, puis l'automate **M0** démarre dans l'état 0. Après 0,25 s le *timer* (temporisateur) expire et **M0** passe à l'état 1. Le *timer* est alors initialisé, la LED mise hors tension, le compteur d'impulsion incrémenté, et un signal appelé UPDATE est mis au niveau haut. L'automate **M1** utilise ce signal. Lorsque le *timer* expire au bout de 0,75 s, **M0** passe de l'état 1 à l'état 0, le *timer* est initialisé, et la LED mise sous tension. Ce cycle se poursuit indéfiniment. À l'état 1, une transition vers ce même état se produit lorsque le compteur d'impulsion atteint 10, ce qui a pour effet de le remettre à zéro. Simultanément et indépendamment tourne l'automate **M1**, mais en temps partagé (du processeur). **M1** démarre à l'état 0, segments éteints. Lorsque le signal UPDATE est mis au niveau haut, **M1** passe à l'état 2 et remet UPDATE au niveau bas. **M1** revient à l'état 1 et commande les segments de l'afficheur en fonction de la valeur de la variable **N**. L'automate reste à l'état 0 dans l'attente d'un nouveau signal UPDATE. La logique pour **M1** sera inversée dans le cas d'un afficheur à cathode commune. Pour adapter le programme à un tel afficheur, référez-vous aux commentaires du code **prog_02_03.asm** que, faute de place, nous ne reproduisons pas ici. Vous pouvez le télécharger depuis la page [1] associée à ce livre.

Ce que j'ai décrit du déroulement du programme devrait suffire à sa compréhension, mais j'aimerais attirer votre attention sur les problèmes que peuvent occasionner les instructions **BSF** et **BCF** lors d'une procédure READ MODIFY WRITE modifiant les bits du registre d'un port. Quelque chose d'étrange peut en effet survenir lorsqu'on modifie la sortie d'une broche **juste après** avoir modifié une autre broche. Ce phénomène vient de la nature capacitive des charges liées aux broches d'E/S et de la façon dont un PIC change la valeur d'une broche. Lorsque le PIC écrit dans le registre d'un port pour modifier une sortie, il doit d'abord lire l'état des broches du port car il ne sait qu'écrire un octet complet. Il lit donc ces états, modifie le bit concerné, et réécrit les autres. Supposons qu'une instruction demande de mettre à 1 le bit 5 du

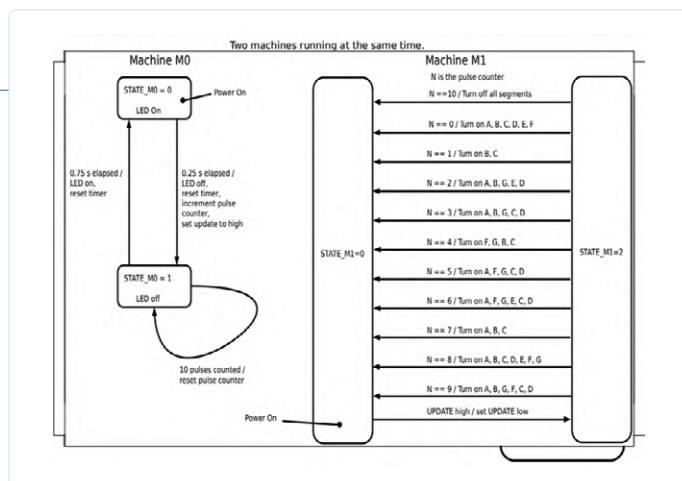


Figure 2-10. Diagramme d'état du programme 02-03.asm (non reproduit ici, cf. [1]).

PORTA (broche physique 2), et que l'instruction suivante demande de modifier la valeur du bit 4 du PORTA (broche physique 3). Le PIC lit les valeurs des broches du PORTA, et l'on pourrait s'attendre à ce que la broche 2 soit au niveau haut puisque son bit 5 vient d'être mis à 1. Pourtant rien n'est moins sûr. Le circuit pourrait être capacitif, et dans ce cas il faudra un certain temps avant que la tension atteigne le niveau haut. Si ce temps est trop long, le PIC considérera que la broche 2 est au niveau bas et remettra le bit 5 à 0 lorsqu'il écrira de nouveau le registre du PORTA. C'est un exemple de situation dont un logiciel de simulation ne pourra pas deviner l'issue. En général, lorsqu'on utilise BCF et BSF, il est préférable d'écrire dans les registres de transfert LATA et LATC. Leur utilisation est toutefois sans risque si l'on écrit le contenu du port entier, pris comme un registre, p. ex. avec MOVWF PORTA.

Différences entre les PIC 12F1822 et 16F1823

Vous le savez, le PIC 12F1822 possède 8 broches tandis que le 16F1823 en offre 14 grâce à ses deux ports d'E/S. Ils se programment presque de la même façon. Le contrôleur et le fichier *include* utilisés doivent être déclarés au début du code source de la façon suivante :

```
LIST P=12F1822
#include <p12f1822.inc>
```

```
LIST P=16F1823
#include <p16f1823.inc>
```

L'avantage d'une puce plus petite est qu'elle prend moins de place sur un circuit imprimé et qu'il y a moins de broches à souder si on n'a pas besoin du port supplémentaire. Lisez la fiche technique de votre PIC pour ne pas vous faire piéger par une de ses particularités. Votre code ne fonctionne pas comme prévu ? Là encore la réponse se trouve dans la fiche technique.

Interruptions et diagrammes d'état

Les interruptions n'apparaissent pas sur les diagrammes d'état car elles ne changent pas l'état de l'automate. Nous l'avons vu dans une section précédente, le programme principal est suspendu lorsqu'une interruption se produit. S'il y a un changement d'état, ce sera donc uniquement sur reprise, et si l'interruption a modifié l'entrée de l'automate (l'augmentation d'un compteur a p. ex. changé le résultat d'une comparaison). ◀

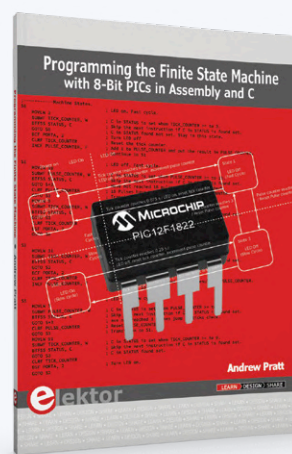


DANS L'E-CHOPPE D'ELEKTOR

Programming the Finite State Machine with 8-Bit PICs in Assembly and C

Andrew Pratt livre dans cet ouvrage (en anglais) des éditions Elektor une introduction détaillée à la programmation des microcontrôleurs PIC au moyen d'automates finis. Les automates finis fournissent une représentation abstraite de la structure d'un programme, et en cela facilitent leur planification, leur écriture et leur modification. Le choix de l'assembleur comme langage de programmation trouve là sa justification. Les deux derniers chapitres vous initieront à la programmation en C et vous aideront à comparer les deux techniques.

L'auteur présente ses explications sous l'autorité de la fiche technique de Microchip, dont vous découvrirez ainsi les parties les plus pertinentes. L'installation des outils nécessaires est décrite et vous n'aurez besoin que d'un câble FTDI pour lire et écrire les programmes sur votre PIC. Vous pourrez travailler depuis Windows ou Linux (Debian, Ubuntu, Fedora...) Le compilateur XC8 de Microchip est utilisé en ligne de commande pour la programmation en C.



> Version imprimée

www.elektor.fr/programming-the-finite-state-machine

> PDF

www.elektor.fr/programming-the-finite-state-machine-e-book

Ont contribué à cet article :

Auteur : **Andrew Pratt**

Traduction : **Hervé Moreau**

Rédaction : **Jan Buiting**

Maquette : **Giel Dols**

Votre avis, s'il vous plaît...

Posez vos questions ou adressez vos commentaires à redaction@elektor.fr

LIEN

[1] [Les programmes du livre, dont les trois décrits ici : www.elektor.fr/programming-the-finite-state-machine](http://www.elektor.fr/programming-the-finite-state-machine)

e-choppe Elektor

des produits et des prix surprenants

L'e-choppe Elektor a évolué. D'abord simple commerce de proximité pour nos productions (livres, magazines, kits et modules), c'est désormais une boutique en ligne bien rodée qui propose des produits surprenants à des

prix très étudiés. Ce sont les produits que nous aimons et testons nous-mêmes. Si vous avez une suggestion, n'hésitez pas : sale@elektor.com.
Seule exigence :
jamais cher, toujours surprenant !

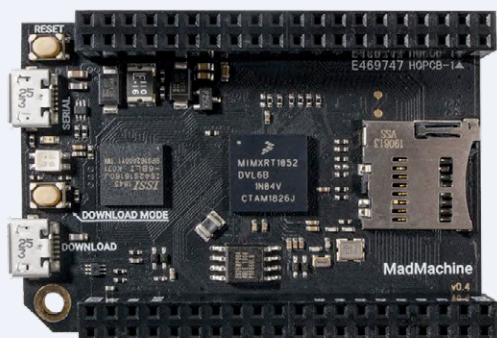


Kit de prise en main
Raspberry Pi 4

Prix : 79,95 €

Prix (membres) : 71,96 €

🛒 www.elektor.fr/19427

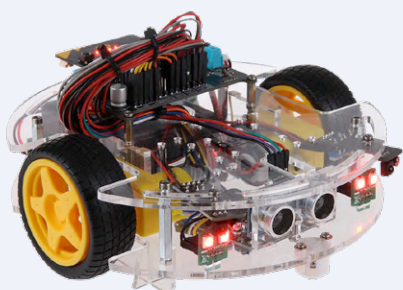


SwiftIO – Carte à micro-
contrôleur codé en Swift

Prix : 72,95 €

Prix (membres) : 65,66 €

🛒 www.elektor.fr/19426



Joy-Car Robot (carte micro:bit BBC incluse)

Prix : 84,95 €

Prix (membres) : 76,46 €

www.elektor.fr/19408



Get Started with the SensorTile.box Bundle

Prix : 69,95 €

Prix (membres) : 62,96 €

www.elektor.fr/19404



Peak UTP05E analyseur de câbles de réseau (Ultimate Atlas IT Kit)

Prix : 259,00 €

Prix (membres) : 233,10 €

www.elektor.fr/19373



Concentrateur GPS LoRaWAN Dragino PG1301 pour Raspberry Pi (868 MHz)

Prix : 124,95 €

Prix (membres) : 112,46 €

www.elektor.fr/19367

hexadoku

casse-tête pour elektorniciens

La dernière page de votre magazine propose toujours une grille de sudoku particulière. Éteignez le fer à souder, oubliez les écrans, rangez les pinces et les fils. Faites travailler votre matière grise d'une autre façon : attrapez un crayon pour remplir la grille d'hexadoku. N'oubliez pas d'envoyer votre réponse pour tenter de gagner un bon d'achat.

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 tirons au sort cinq des réponses internationales correctes reçues dans les délais ; leurs auteurs recevront chacun un bon d'achat Elektor d'une valeur de 50 €.

Où envoyer votre réponse ?

Envoyez votre réponse (les chiffres sur fond grisé) avec vos coordonnées par courriel, avant le **8 décembre 2020** à l'adresse **hexadoku@elektor.fr**

Les gagnants

La solution de la grille du numéro de septembre/octobre 2020 est **35904**.

Les cinq bons Elektor d'une valeur de **50 €** vont à

Charlotte Mies (Pays-Bas) – Jean-Marie Mahieu (Belgique) – Roberto Visentin (Italie) –
Herman Pusch (Allemagne) – Neil Wood (Royaume-Uni)

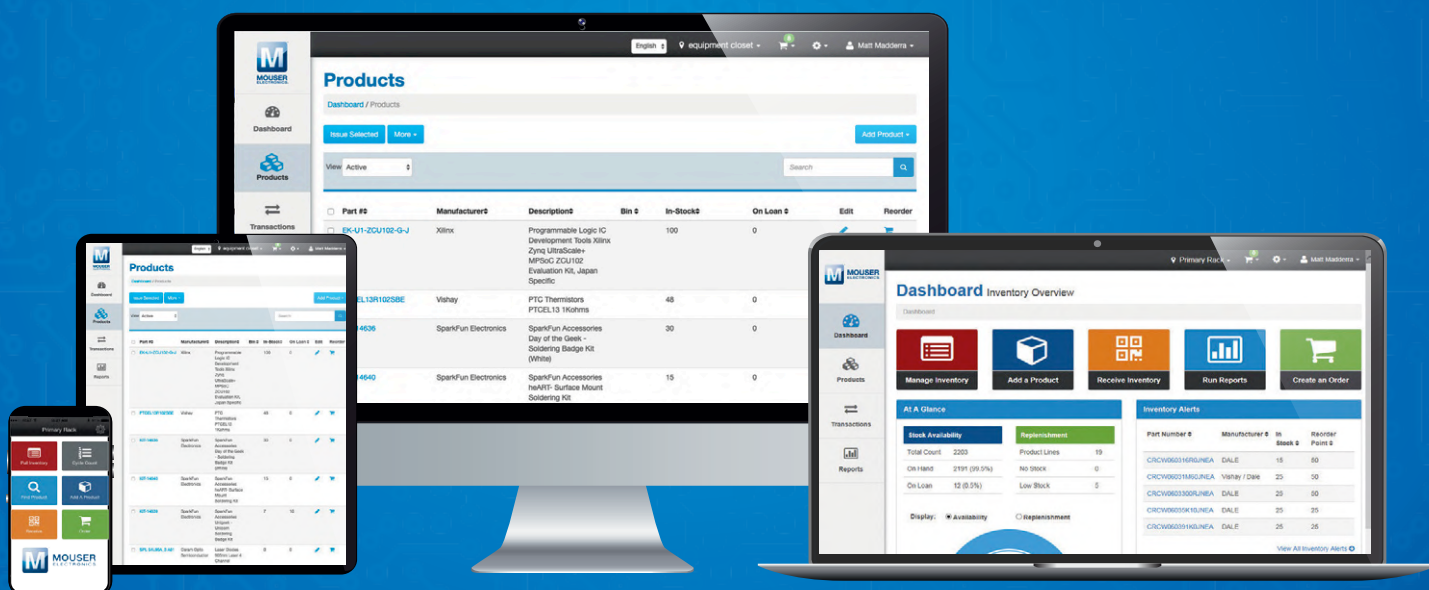
Bravo à tous les participants et félicitations aux gagnants !

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

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

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.

Outil de gestion des stocks gratuit



Gérer les stocks



Comptage du
cycle de stock



Trouver un produit



Ajouter un produit



Recevoir



Commander

mouser.fr/inventory-management



MOUSER
ELECTRONICS

Rejoignez les électroniciens de la communauté Elektor

Devenez membre



maintenant !



- ✓ accès à l'archive numérique depuis 1978 !
- ✓ 6x magazine imprimé Elektor
- ✓ 9x magazine numérique (PDF) dont Elektor Industry (EN)
- ✓ 10 % de remise dans l'e-shoppe et des offres exclusives pour les membres
- ✓ le DVD annuel d'Elektor
- ✓ accès à plus de 1000 fichiers Gerber, collaboration avec les milliers d'électroniciens d'Elektor LAB, et une ligne directe avec nos experts !
- ✓ possibilité de voir votre projet publié ou vendu par notre boutique en ligne

Également disponible

abonnement



sans papier !

- ✓ accès à l'archive numérique d'Elektor
- ✓ 10 % de remise dans l'e-shoppe
- ✓ 6x magazine Elektor (PDF)
- ✓ offres exclusives
- ✓ accès à plus de 1000 fichiers Gerber



www.elektormagazine.fr/membres