ISSN 0181-7450

F : 15,50 €

(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

elektor 499

créer > partager > vendre

JANVIER & FÉVRIER 2023
ELEKTORMAGAZINE.FR

Sortie vidéo sur les micro-contrôleurs

vidéo composite



p. 6



p. 70

THE TUBE

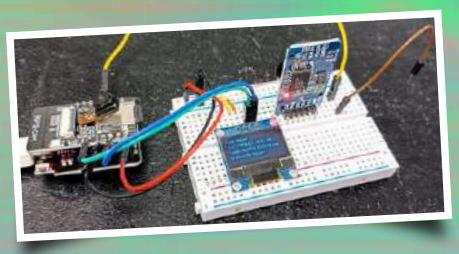
un surprenant amplificateur à tubes

FOCUS SUR **Audio et vidéo**

p. 16

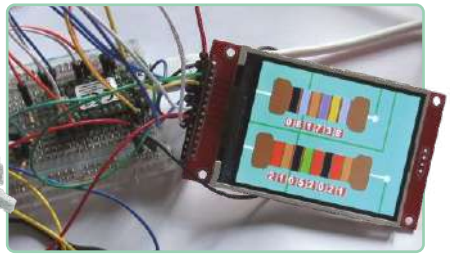
CAMÉRA ESP32

si simple qu'on n'a même pas besoin du wifi





Alimentation ATX pour Raspberry Pi tensions différentes à des niveaux de puissance élevés ! p. 20



Horloges radiocommandées SDR cinq signaux horaires, six affichages p. 29



Utiliser la lumière pour les effets sonores – filtre synthétiseur à base de LDR p. 98

L 19624 - 499 - F: 15,50 € - RD

NOTRE GAMME

PAR DES TECHNICIENS POUR LES TECHNICIENS

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

Uniquement le meilleur pour vous - provenant de plus de 1500 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.

STRUCTURE ENERGY : CONTRÔLE EFFICACE DE L'ÉNERGIE

Shelly

Thermostat de radiateur Shelly TRV

Robinet de radiateur thermostatique commandé par Wi-Fi

Gestion du chauffage extrêmement efficace pour économiser les coûts énergétiques. La large plage de température vous permet de régler la température avec précision selon vos besoins.

- Remplace votre ancien robinet de chauffage en seulement deux minutes
- Batterie de 6 500 mAh
- Durée de vie de la batterie allant jusqu'à deux ans
- Plage de température : 5 °C - 30 °C



PRODUIT RECOMMANDÉ

N° de commande :
SHELLY TRV

80,60 (67,16)



Découvrez notre vaste gamme de produits Shelly pour un confort optimal dans votre maison

▶ www.reichelt.com/shelly

INTELLIGENT. OPTIMAL. PRATIQUE

GÉRER EFFICACEMENT L'ÉNERGIE

Découvrir maintenant ▶
https://rch.lt/energy-fr



reichelt elektronik MAGAZINE

TUTORIELS:
Un chauffage efficace en toute simplicité – Ces conseils vous aident à économiser de l'énergie et des coûts

Découvrir maintenant ▶
https://rch.lt/MG530f



Types de paiement : Sofort. VISA MasterCard Maestro PayPal amazon pay

PRIX DU JOUR! Prix à la date du: 14.12. 2022

- Excellent rapport qualité prix
- Plus de 130 000 produits sélectionnés
- Livraison fiable - depuis l'Allemagne dans le monde entier

www.reichelt.com

Assistance téléphonique: +33 9 75 18 03 04

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://rch.lt./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

46ème année
n° 499 – janvier-février 2023

ISSN 0181-7450
Dépôt légal : janvier 2023
CPPAP 1125 T 83713
Directeur de la publication : Donatus Akkermans

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

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 :
Raoul Morreau
Tél. : +31 (0)6 4403 9907
Courriel : raoul.morreau@elektor.com

DROITS D'AUTEUR :
© 2023 Elektor International Media B.V.

Toute reproduction ou représentation intégrale ou partielle, par quelque procédé que ce soit, des pages publiées dans la présente publication, faite sans l'autorisation de l'éditeur est illicite et constitue une contrefaçon. Seules sont autorisées, d'une part, les reproductions strictement réservées à l'usage privé du copiste et non destinées à une utilisation collective, et, d'autre part, les analyses et courtes citations justifiées par le caractère scientifique ou d'information de l'oeuvre dans laquelle elles sont incorporées (Loi du 11 mars 1957 -art. 40 et 41 et Code Pénal art. 425).
Certains circuits, dispositifs, composants, etc. décrits dans cette revue peuvent bénéficier de droits propres aux brevets; la Société éditrice n'accepte aucune responsabilité du fait de l'absence de mention à ce sujet. Conformément à l'art. 30 de la Loi sur les Brevets, les circuits et schémas publiés dans Elektor ne peuvent être réalisés que dans des buts privés ou scientifiques et non commerciaux. L'utilisation des schémas n'implique aucune responsabilité de la part de la Société éditrice. La Société éditrice n'est pas tenue de renvoyer des articles qui lui parviennent sans demande de sa part et qu'elle n'accepte pas pour publication. Si la Société éditrice accepte pour publication un article qui lui est envoyé, elle est en droit de l'amender et/ou de le faire amender à ses frais; la Société éditrice est de même en droit de traduire et/ou de faire traduire un article et de l'utiliser pour ses autres éditions et activités, contre la rémunération en usage chez elle.

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

ÉDITO

Jens Nickel

rédacteur en chef d'Elektor Magazine



Pour les yeux et les oreilles

Tout est y dedans – et cela a toujours été le cas chez Elektor. Vous pouvez apprendre tout autant en vous plongeant dans un schéma de circuit qu'en vous plongeant dans le code source. Dans cette édition, nous abordons le sujet marquant de l'audio et de la vidéo – tant analogique que numérique, en théorie et en pratique. En effet, cela ne concerne pas que les yeux et les oreilles à vouloir en avoir pour leur argent, la matière grise de la tête veut également en avoir pour son argent.

Notre article de couverture est particulièrement invitant. Une carte à microcontrôleur peu coûteuse, quelques composants externes et un logiciel libre, ce sont tout ce dont vous avez besoin pour émettre des signaux vidéo qui peuvent être visualisés sur un moniteur. Cela ne fonctionne pas seulement avec un signal vidéo composite ; VGA et HDMI sont aussi possibles. Mon collègue Mathias Claußen a rédigé un article riche en informations de fond, mais ce dernier est devenu si volumineux qu'il a poussé notre flux de travail rédactionnel à ses limites. Au lieu d'imprimer un article de 20 pages dans ce numéro, nous avons décidé de diviser l'article de Mathias en deux parties. Ce numéro explique la manière avec laquelle on peut extraire un signal composite d'un petit contrôleur à 8 bits, en noir et blanc, sous forme d'image en niveaux de gris ou en couleur (page 6). Dans le prochain numéro, des artistes informatiques un peu plus costauds seront en action pour générer des signaux VGA, DVI et HDMI.

Pour les utilisateurs précoces, je recommande l'article de Tam Hanna, qui présente le cadre de developpement audio d'Espressif, en utilisant des applications pratiques comme un lecteur MP3 (page 88). J'ai été impressionné de voir ce qu'il est possible de réaliser avec un ESP32 et quelques lignes de code en termes de traitement audio. Certes, la courbe d'apprentissage est raide. Mais pour tous ceux qui veulent développer des projets professionnels avec une production vocale ou musicale, les efforts devraient être récompensés à tous égards.

Notre amplificateur pour casque 32 Ω ne nécessite pas du tout de microcontrôleur. Un double ampli-op audio OPA2134 est accompagné par deux tampons rapides BUF634A, qui se chargent de l'amplification du courant. Si vous le souhaitez, vous pouvez remplacer ces circuits intégrés par un circuit discret.

Rejoignez-nous et apprenez !

notre équipe

Rédacteur en chef : Jens Nickel
Rédaction : Asma Adhimi, Eric Bogers, Rolf Gerstendorf, Thomas Scherer, Brian Tristam Williams
Laboratoire : Mathias Claussen, Ton Giesberts, Clemens Valens
Maquette : Harmen Heida, Sylvia Sopamena, Patrick Wielders

DEUTSCHE FACHPRESSE

Elektor est membre de VDZ (association d'éditeurs de magazines allemands) qui « représente les intérêts communs de 500 éditeurs allemands grand public et B2B. »

DANS CE NUMÉRO

Elektor n° 499
janvier - février 2023



Rubriques

3 **Édito**

34 **démarrer en électronique**
diodes spéciales

38 **sur le vif**
errances qualitatives

52 **Zone D**
mesure du THD avec un oscilloscope et une FFT

77 **questions d'éthique**
à quand des biomatériaux pour l'électronique ?

103 **rétronique**
amplificateur Hi-Fi d'Elektor

106 **visite à domicile**
afficheur volumétrique made in Canada

110 **projet 2.0**
corrections, mises à jour et courriers des lecteurs

114 **Hexadoku**
Casse-tête pour elektorniciens

Articles de fond

FOCUS
6 **sortie vidéo sur les microcontrôleurs (1)**
vidéo composite

14 **electronica 2022**
nouvelles du plus grand salon mondial de l'électronique

47 **kit de développement MakePython ESP32**
tout dans une boîte

80 **commutateur d'antenne Opera Cake pour HackRF One**
connectez jusqu'à huit antennes à votre SDR

82 **entretien**
concevoir avec Arduino et plus

FOCUS
88 **les signaux audios et l'ESP32**
l'environnement ESP-ADF en pratique

Industrie

FOCUS
56 **machines à vision totale**
la technologie derrière les systèmes de vision industrielle d'aujourd'hui

FOCUS
60 **infographies**
faits et chiffres

FOCUS
62 **l'évolution de la commande vocale et audio des appareils électroniques**

66 **bilan du WEEF 2022**

68 **bilan du salon electronica 2022**
les innovateurs n'ont pas manqué d'impressionner

The Tube
un surprenant amplificateur à tubes



70





Projets

FOCUS
16 **caméra ESP32**
si simple qu'on n'a même pas besoin du wifi

20 **alimentation ATX pour Raspberry Pi**

FOCUS
26 **amplificateur pour casque 32 Ω**
une solution simple mais de qualité à trois CI

29 **horloges radiocommandées SDR**
cinq signaux horaires, six affichages

40 **rétro-ingénierie d'un badge LED Bluetooth Low Energy**
comment commander un appareil BLE avec un script Python

FOCUS
70 **The Tube**
un surprenant amplificateur à tubes

85 **jauge de longueur à lidar**
mesure jusqu'à 12 m

FOCUS
98 **utiliser la lumière pour les effets sonores**
filtre synthétiseur à base de LDR, contrôlé en tension, 24 dB/oct

Bientôt dans ces pages

Le numéro de mars – avril 2023
Vous retrouverez dans le prochain magazine Elektor l'habituel mélange stimulant de réalisations originales, de circuits, d'articles de fond, de sujets nouveaux, de trucs et d'astuces pour les électroniciens. Le thème de ce numéro sera « l'embarqué et l'IA. ».

Quelques-uns des points forts :

- réveil 2.0
- vidéo avec microcontrôleurs : VGA, HDMI, DVI
- Raspberry Pi Pico : PIO en pratique
- satellites Spacebee pour communication IdO
- programmation d'applications IdO commandées par la voix
- l'API wifi d'Android
- ma première radio définie par logiciel
- ChipTweaker
- générateur de nombres aléatoires USB True

Et bien d'autres choses encore !

Le numéro de mars – avril 2023 du magazine Elektor sera publié aux alentours du 11 mars 2023. La date d'arrivée du magazine papier chez les abonnés dépend des aléas d'acheminement. Le contenu et les titres des articles peuvent être modifiés.

47
kit de développement MakePython ESP32
tout dans une boîte





sortie vidéo sur les microcontrôleurs (1)

vidéo composite



Mathias Claussen (Elektor Lab)

Le thème de la sortie vidéo sur les microcontrôleurs remonte aux débuts de ces petites puces polyvalentes. Les microcontrôleurs d'aujourd'hui ont une puissance de calcul bien supérieure à celle de l'ordinateur familial Sinclair ZX81 d'il y a presque 42 ans, mais même les microcontrôleurs les plus récents sont loin de rivaliser en taille mémoire avec les cartes graphiques modernes où elle se mesure en gigaoctets. Cela n'empêche pas les développeurs de produire d'étonnantes images animées avec un ATmega, un ESP32 ou un RP2040. Dans la première partie de cette série, nous aborderons la sortie de vidéo composite. Ensuite, nous poursuivrons avec le VGA et même le DVI. Dans tous les cas, quelques astuces et un timing précis sont nécessaires. Il ne s'agit donc pas seulement de théorie, mais aussi d'exemples pratiques comme point de départ pour vos propres expériences.

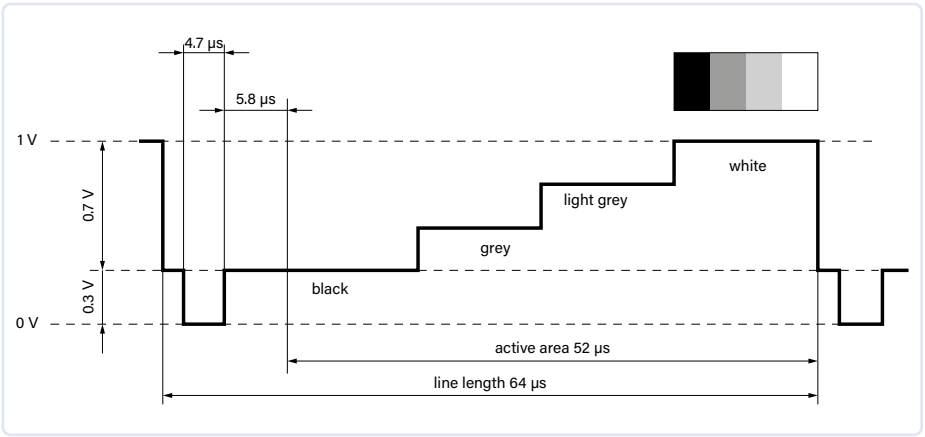


Figure 1. Signal VBS avec timing PAL. (Source : Wikipedia)

L'histoire des formats vidéo remonte aux débuts de la télévision. Comme pour la radio, on a élaboré des standards et des normes pour la télévision. À l'époque de la télévision couleur analogique, les normes les plus courantes étaient le National Television Systems Committee (NTSC - Amérique du Nord et du Sud, Japon), l'Alternance de Phase par Ligne (PAL - Europe, Amérique du Sud, Afrique et Asie) et le Séquentiel Couleur à Mémoire (SECAM - France, Afrique et URSS).

La procédure de base pour la transmission analogique des signaux vidéo a suivi la méthode VBS (Video Blanking and Sync) pour ces normes. La résolution et la fréquence d'images pour la VBS dépendent de la norme de télévision sous-jacente. Pour la norme NTSC, elle est de 480 lignes visibles avec 640 pixels visibles à 59,94 trames par seconde. PAL et SECAM ont 576 lignes visibles avec 720 pixels visibles et 50 trames par seconde (576i). En outre, NTSC, PAL et SECAM diffèrent également au niveau de la modulation et de la manière dont les informations de couleur sont ajoutées. À l'ère de la transmission numérique des images, ces trois méthodes ont largement perdu de leur importance. Cependant, elles subsistent sous la forme de formats d'images numériques pour les DVD ou la définition standard pour la télévision (SDTV).

Vidéo composite avec CVBS

Les premiers signaux de télévision normalisés étaient conçus pour la transmission d'images monochromes. Différents formats d'image ont été développés aux Etats-Unis et en Europe. La **figure 1** illustre le timing d'un signal VBS au format PAL, à titre d'exemple de ces normes.

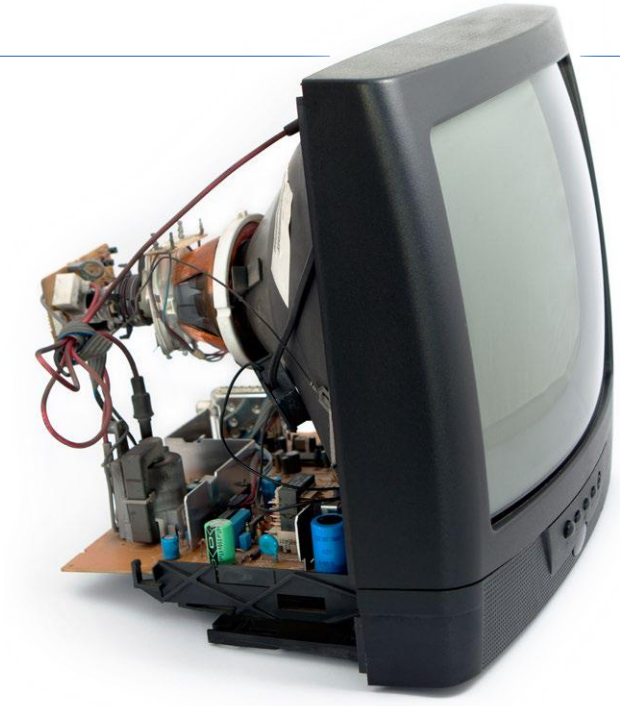


Figure 2. Téléviseur ouvert avec tube cathodique. (Source : Shutterstock/ Sergio Sergo)

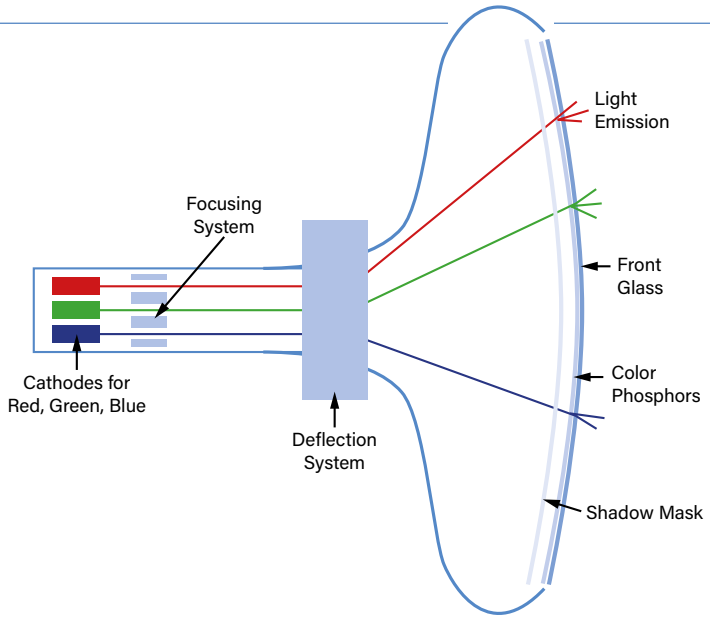


Figure 3. Structure schématique d'un tube image couleur. (Source : ITWissen.info)

Cette norme était particulièrement adaptée à la reproduction d'images avec des tubes cathodiques (**figure 2**). Le principe de fonctionnement de ces tubes diffère considérablement de celui des écrans LCD ou OLED. Dans ces tubes, un faisceau d'électrons modulé, dévié horizontalement et verticalement, est dirigé sur une couche phosphorescente. Au point d'impact, cette couche s'illumine proportionnellement à l'intensité du faisceau d'électrons (**figure 3**). Le balayage horizontal et vertical produit une image bidimensionnelle.

Du point de vue de l'observateur, une image (en noir et blanc) se compose de lignes balayées de gauche à droite et de haut en bas (**figure 4**). Le premier pixel d'une image est donc situé en haut à gauche. Cette convention est largement conservée dans les écrans numériques modernes.

Quelques connaissances historiques sur la génération d'images au moyen d'un faisceau d'électrons en mouvement facilitent la compréhension des signaux vidéo modernes. Le signal de la **figure 1** contient la structure d'une ligne d'image. Au début (**figure 5**) du signal, on distingue l'extinction (*blanking*) horizontale. Au début de cette extinction, le faisceau d'électrons se trouve sur le bord droit du tube image et doit donc retourner à l'extrême gauche pour la ligne suivante. Pendant ce saut de droite à gauche, le faisceau d'électrons est éteint en réglant le signal de luminosité sur le noir ou même sur « l'infra-noir ».

L'extinction horizontale prend 12 µs pour le PAL et 10,9 µs pour le NTSC. Elle se compose de trois sections appelées « palier avant » (*front porch*), « top de synchro » (*sync tip*) et « palier arrière » (*back porch*). Le niveau de luminosité du palier avant (PAL = 1,65 µs/NTSC = 1,4 µs) est égal ou légèrement inférieur à la valeur du noir à 0,3 V, tandis que le faisceau d'électrons se déplace encore plus vers la droite, hors de l'image visible. Le niveau du top de synchro (PAL et NTSC = 4,7 µs) est de 0 V,

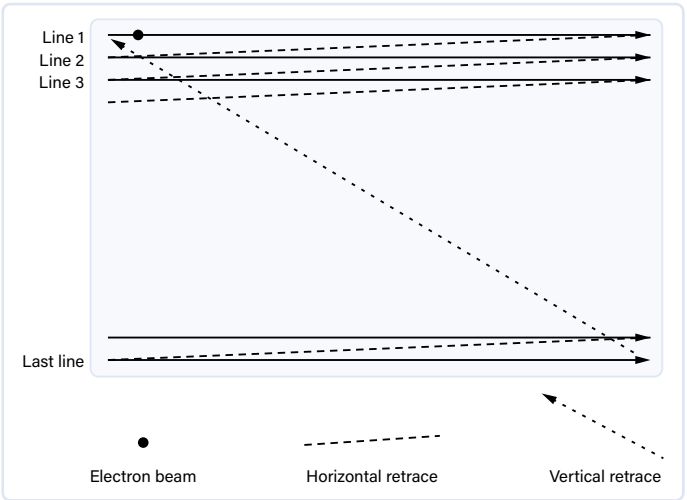


Figure 4. Composition d'image basée sur les lignes.

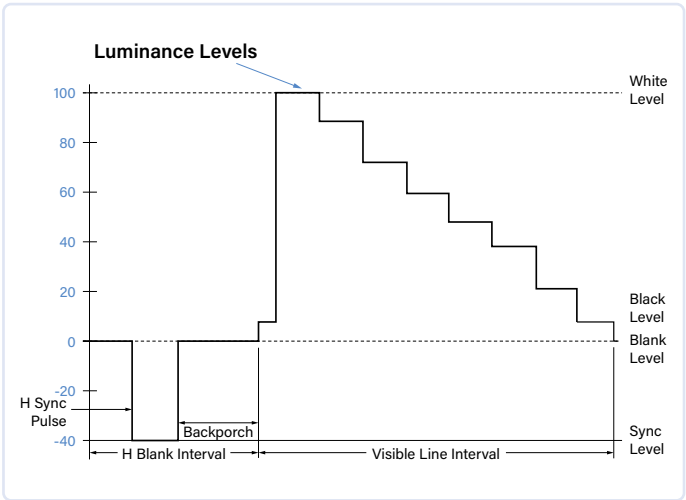


Figure 5. Extinction horizontale. (Source : www.edn.com)

nettement inférieur à la valeur du noir. Le top de synchro ramène le faisceau d'électrons de droite à gauche. Le palier arrière (PAL et NTSC = 1,6 µs) fournit le niveau de référence du noir (0,3 V) pour la ligne. Le contenu de l'image à afficher commence à la fin du palier arrière. On dispose alors de 52 µs pour la partie visible de la ligne. Le nombre de pixels affichés pendant ces 52 µs dépend du format vidéo. En PAL, il y a généralement 720 pixels visibles. Lorsque toutes les lignes sont tracées et que le faisceau d'électrons a atteint le bas à droite, il doit être renvoyé au point de départ de l'image (en haut à gauche). Ce processus sera décrit dans une section séparée.

Synchronisation et entrelacement

La bande passante analogique pour la transmission des images de télévision était suffisante pour seulement 25 images par seconde pour le PAL et 29,97 images par seconde pour le NTSC. Pour l'œil humain, cela se traduit par une image en mouvement, mais elle n'est pas très fluide et crée un effet de scintillement très désagréable. Avec une fréquence d'images plus rapide de 50 ou même 59,94 images par seconde, le scintillement est sensiblement réduit. La plupart des gens perçoivent alors cette image comme une impression visuelle lisse. La bande passante radio disponible étant limitée, l'amélioration est venue de la technique du balayage entrelacé. Au lieu de transmettre toutes les lignes d'une image à une fréquence plus élevée sous forme de trames complètes, les lignes dont le numéro est pair ou impair sont transmises alternativement à deux fois la fréquence, c'est-à-dire 50 ou 59,94 Hz, respectivement (**figure 6**).

Ainsi, on transmet une première trame avec les lignes de numéros impairs, puis une deuxième trame avec les lignes de numéros pairs. Pour que les lignes d'une trame puissent s'insérer dans les espaces de l'autre trame, un signal de détection est émis au début de chaque trame.

Les lignes d'une image complète étant numérotées en fonction de la séquence de transmission, la première trame en PAL comporte les lignes 1 à 313 et la numérotation de la deuxième trame continue jusqu'à la ligne 625. La synchronisation verticale pour la fin d'une trame et le début de la suivante est un peu délicate : les lignes 311 à 317 à la transition entre les trames impaire et paire ainsi que les lignes 623 à 625 à la transition entre les trames paire et impaire de l'image complète suivante ne sont pas visibles. Ces lignes comprennent des impulsions de synchronisation plus courtes avec des niveaux compris entre 0,3 V et 0 V. La **figure 7** montre la séquence pour une trame PAL complète. Grâce à ces impulsions, l'électronique peut reconnaître la parité des numéros de ligne de la trame.

Vidéo monochrome avec les contrôleurs AVR

Vous trouverez des informations de base sur les signaux composites tels que CVBS ou FBAS sur Wikipedia [1]. Pour plus de détails et des astuces de circuits, consultez le livre Analogue Video [2] d'Angelo La Spina.

Si le signal vidéo est transmis en tant que signal VBS (uniquement la luminance, sans

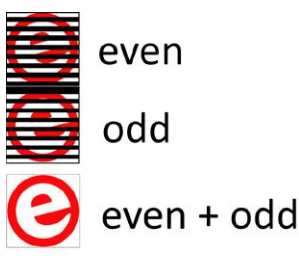


Figure 6. Composition de l'image à partir de deux trames.

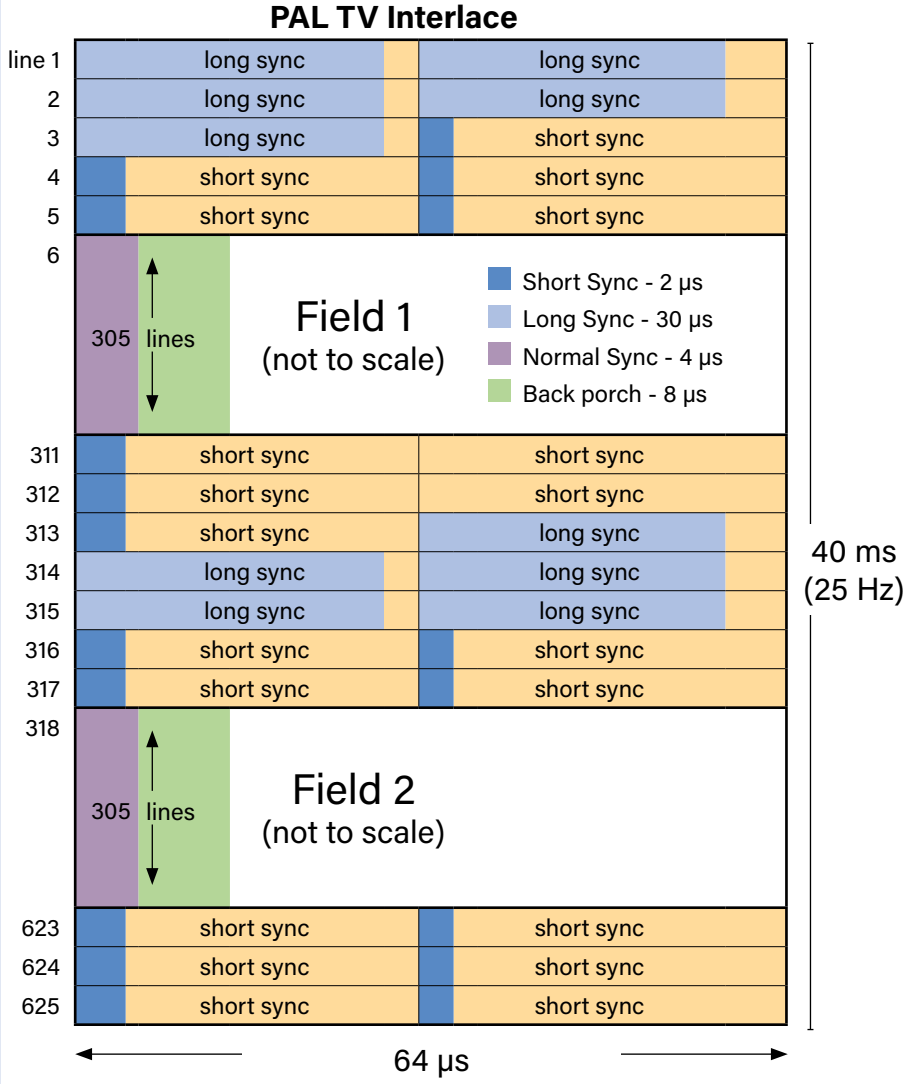


Figure 7. Composition d'une image PAL complète. (Source : martin.hinner.info)

la chrominance) à l'aide d'un câble vidéo composite approprié (**figure 8**), la génération des informations de chrominance peut être assez simple. Si vous n'utilisez que le blanc et le noir au lieu de l'échelle de gris, il vous suffit d'émettre deux tensions appropriées plus la synchronisation. Outre la simplification du circuit, cela permet également de réduire la mémoire nécessaire à la génération de l'image.

Le microcontrôleur Atmega328P d'un Arduino Uno est donc capable de générer une image monochrome de 128x96 pixels. Tous les pixels peuvent être stockés dans la mémoire interne de l'Arduino Uno, car seuls 1536 octets sont nécessaires. Le câblage d'un Arduino Uno avec sortie composite est illustré à la **figure 9.**

Comme vous pouvez le constater, deux résistances sont suffisantes pour produire un signal adéquat. Pour produire une image, vous pouvez utiliser la bibliothèque *TVOut* [3] qui prend en charge la synchronisation pour NTSC et PAL. La sortie vidéo qu'elle permet de réaliser va du simple texte à vos propres jeux. Hackvision [4] est une plateforme de console de jeu. Il s'agit d'un matériel ouvert pour lequel, dans les cas extrêmes, une carte d'expérimentation et quelques composants suffisent. Un Arduino peut également être converti en plateforme Hackvision (**figure 10**) et ainsi utiliser la bibliothèque de jeux associée.

La sortie vidéo avec un microcontrôleur AVR nécessite un de ses timers. La synchronisation horizontale et verticale est assurée par le Timer1 et sa broche de sortie PB1 (OC1A). Sous le contrôle du timer, les pixels apparaissent ligne par ligne sur une broche dédiée (PD7).

Les valeurs des résistances à PB1 et PD7 peuvent être déterminées assez facilement. Une entrée composite sur les moniteurs ou les téléviseurs a une impédance de 75 Ω. Le niveau pour la synchronisation est compris entre 0 V et 0,3 V. La tension pour les valeurs de luminosité des pixels est comprise entre 0,3 V pour le noir et 1 V pour le blanc, respectivement. Dans ce qui suit, il est supposé que le microcontrôleur est alimenté en 5 V.

La **figure 11** montre le diviseur de tension pour la sortie vidéo à l'impédance typique de 75 Ω. La broche 9 fournit les signaux de synchronisation. Mathématiquement, une



Figure 8. Câble pour vidéo composite avec fiche RCA. (Source : Shutterstock/Woodpond)

valeur de 1175 Ω serait nécessaire pour R1, de sorte que la chute de tension sur R2 soit de 0,3 V avec un niveau haut sur la broche 9. 1 kΩ fournit un maximum de 0,34 V sur R2, valeur suffisamment précise. Les valeurs de luminosité apparaissent sur la broche 7. Sur R2, il faut maintenant des valeurs comprises entre 0,34 V (noir) et 1 V (blanc). Pour obtenir 1 V sur R2, une valeur de 375 Ω est nécessaire pour R1 et R3 en parallèle, ce qui donnerait 600 Ω pour R3. La valeur immédiatement supérieure de 470 Ω de la série E-12 garantit que la tension à R2 ne pourra jamais dépasser la valeur de 1 V.

Le fait que cela fonctionne avec un Arduino Uno ou un ATmega328P prouve qu'une faible puissance de calcul et peu de mémoire suffisent pour afficher des graphiques sur un écran. Comme l'image complète peut être conservée dans la mémoire interne, le dessin d'une nouvelle image n'est pas critique en termes de temps, seule la génération des signaux vidéo à l'aide de Timer1 est critique.

Est-il également possible de produire des niveaux de gris, c'est-à-dire plusieurs valeurs de luminosité avec l'Arduino ? Oui, mais la mémoire interne est le facteur limitant. Avec 16 nuances de gris (niveaux de gris 4 bits) et des résistances adéquates, il faut 6144 octets pour la mémoire vidéo si l'on veut conserver la résolution de 128x96 pixels et qu'une image complète puisse tenir dans la mémoire.

Raspberry Pi Pico et Composite

Un Raspberry Pi Pico peut également émettre un signal composite, et il supporte aisément plus de 50 nuances de gris. Pour cela, une échelle de résistances R-2R [5] est utilisée comme convertisseur numérique/analogique. La **figure 12** montre un circuit adéquat composé de résistances de 180, 320 et 360 Ω. Comme on peut le voir sur la page GitHub

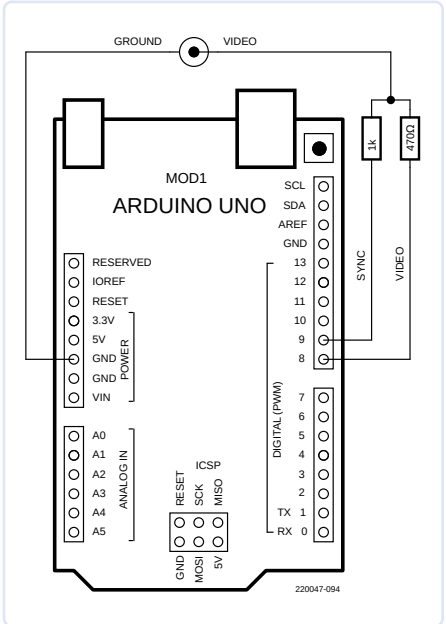


Figure 9. Sortie vidéo N/B avec deux résistances sur un Arduino Uno.

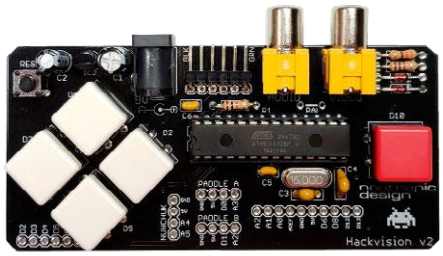


Figure 10. Matériel Hackvision. (Source : nootropic design)

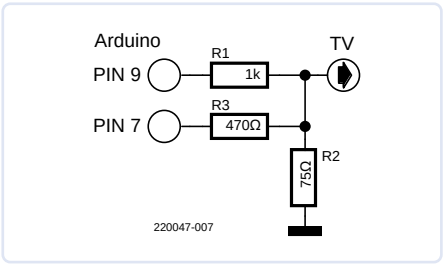


Figure 11. Le circuit de sortie vidéo de la figure 9.

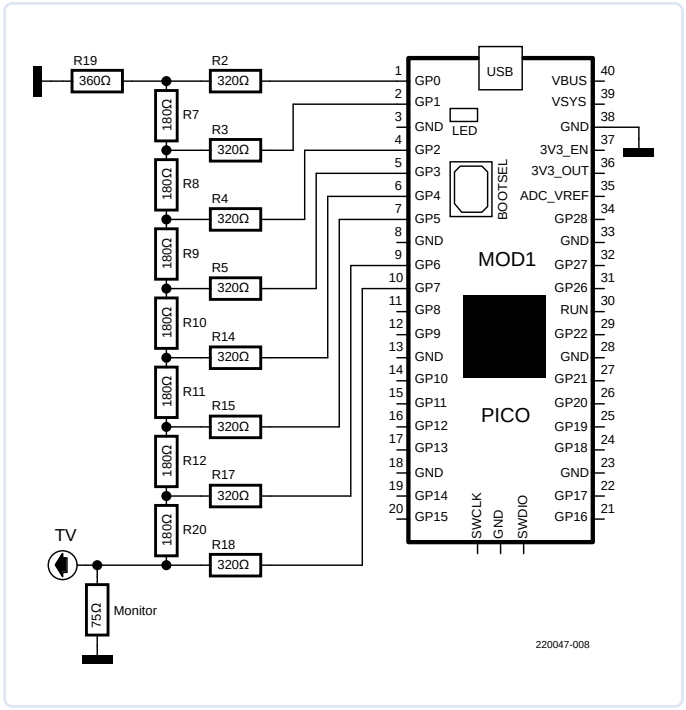


Figure 12. Un réseau R-2R comme CNA sur le Raspberry Pi Pico.

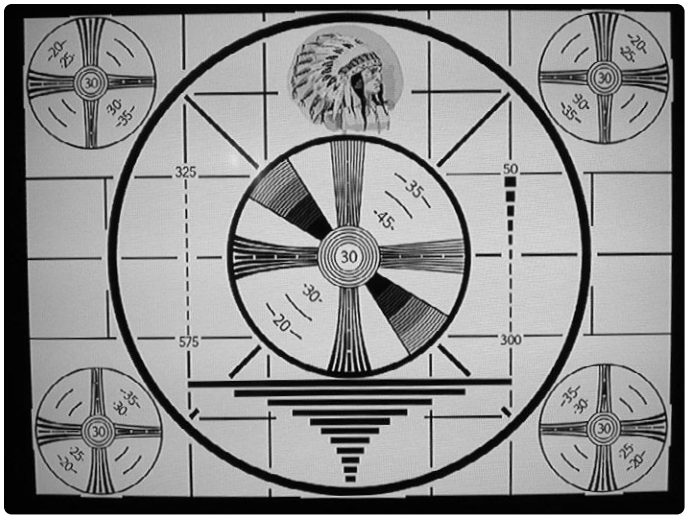


Figure 13. Image en niveaux de gris provenant d'un Raspberry Pi avec 512x384 pixels. (Source : tinyurl.com/2p8z27a2)

du projet « pico-composite8 » [6], la résistance interne des broches GPIO ne peut pas être négligée ici. Le développeur du projet l'a évaluée à 40 Ω environ. Le Pico émet un signal composite conforme à la norme NTSC. Il charge les images soit à partir de la RAM soit à partir de la Flash et peut fournir un maximum de 512x384 pixels, comme le montre la **figure 13**.

Toutefois, le projet n'est qu'une démonstration de faisabilité et n'inclut pas de bibliothèque générique prête à l'emploi. Le fait qu'un Raspberry Pi Pico puisse produire une vidéo en niveaux de gris de 512x384 pixels montre que ce n'est pas tant une question de puissance de calcul que de bon timing. Si une image complète est stockée dans la RAM du Raspberry Pi Pico, il ne reste disponibles qu'environ 64 ko des 264 ko pour vos propres applications. Mais si vous considérez que même un ATmega peut gérer des jeux comme Tetris ou Pong, y compris la sortie vidéo, cet espace devrait être plus que suffisant pour vos propres créations.

Couleur pour la vidéo composite

Plus de 50 nuances de gris, c'est bien, mais la couleur, c'est bien mieux. Lorsqu'il s'agit de vidéo composite et de couleur, il devient beaucoup plus difficile qu'auparavant de générer un signal analogique adéquat. En 2003, lors d'un événement Hackaday [7], Rickard Gunée a démontré la génération d'un signal composite (PAL ou NTSC) en utilisant un Scenix/Ubicom SX28 à environ 50 MHz.

Mais ajouter de la couleur à un signal vidéo, à quel point est-ce difficile ? La réponse dépend de la manière dont les informations de couleur sont ajoutées au signal composite.

Plein de couleurs grâce à PAL et NTSC

Lors du passage à la télévision en couleur, il y a quelques décennies, aucun signal entièrement nouveau n'a été introduit, car il fallait s'assurer que les téléviseurs en noir et blanc existants restaient compatibles. Ce problème a été résolu au niveau international de trois manières différentes, ce qui explique la coexistence des normes NTSC, PAL et SECAM. Leur point commun est que l'information couleur a été ajoutée au signal monochrome existant.

Alors que le PAL et le NTSC sont similaires dans leur principe (modulation en quadrature pour la couleur), le SECAM diffère par l'utilisation de la modulation de fréquence.

Les couleurs qui sont fournies dans leurs trois composantes de base (rouge, vert et bleu) doivent être converties dans un espace couleur YUV ou YCbCr [9] avant de pouvoir être émises par un signal vidéo composite. La conversion de RGB en YUV se calcule comme suit :

$$Y = 0{,}299 * R + 0{,}578 * G + 0{,}144 * B$$
$$U = 0{,}493 * (B - Y)$$
$$V = 0{,}877 * (R - Y)$$

Cela transforme les valeurs de R, G et B en un niveau compris entre 0 et 1. Les formules montrent qu'une mise en œuvre pour les microcontrôleurs nécessite soit une puissance de calcul, soit des compétences en programmation.

Voici une explication de base de l'incorporation de la couleur basée sur les normes PAL et NTSC. La **figure 1** montre un signal PAL monochrome auquel aucune information de couleur n'a encore été ajoutée. Pour les informations de couleur, PAL utilise une sous-porteuse à 4,433 618 75 MHz, qui sert de signal de référence. Cette référence est envoyée

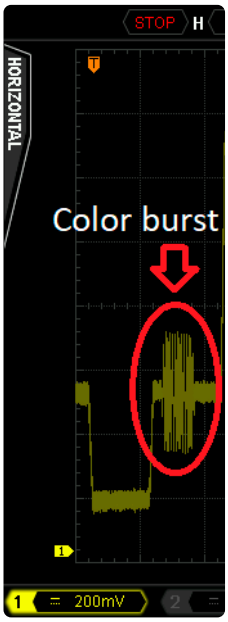


Figure 14. Oscillogramme de la salve de couleurs d'un signal PAL.

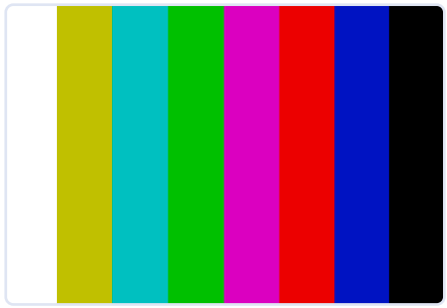


Figure 15. Mire avec barres de couleur.

dans chaque ligne sous forme de « salve de couleur » (**figure 14**) pendant la synchronisation horizontale. La mire (**figure 15**) sert à expliquer comment sont codées les différentes couleurs de l'image test. L'image de test correspond aux barres de couleur de l'UER [10] ; un Raspberry Pi Zero sert de générateur de signal d'image.

Le signal d'une ligne d'image est représenté sur la **figure 16**. La luminosité (ligne blanche comme valeur moyenne du signal) et l'amplitude de l'information de couleur sont facilement visibles. Il existe une troisième information dans le signal, qui est cachée dans la phase du signal. La **figure 17** montre (entouré en rouge) le changement de phase lors de la transition vers une autre couleur.

Ainsi, les informations relatives à la luminosité et à l'amplitude du signal couleur et au déphasage sont contenues dans le signal. S'il y a une erreur dans la phase ou son évaluation pendant la transmission ou la réception du signal, la couleur de l'image change, ce qu'on pouvait compenser manuellement sur les récepteurs NTSC au moyen d'un bouton de « réglage de la teinte » [11] (plus tard, il y eut aussi des solutions électroniques). Le PAL évite ce problème en décalant l'information de phase de 180° une ligne sur deux, ce qui compense l'erreur de phase entre deux lignes consécutives.

Vidéo composite couleur avec ESP32

Un ESP32 peut facilement produire un signal composite monochrome avec quelques petites astuces, et même un Arduino est capable de le faire. Mais la génération d'un signal composite couleur impose des exigences nettement plus élevées en matière de modulation.

En 2018, bitluni a démontré sa méthode de génération d'un signal composite couleur en utilisant un ESP32 [12]. Il a montré que les 13,33 MSa/s des CNA de l'ESP32 sont suffisants pour ajouter une porteuse et des informations de couleur au signal. Comme pour le

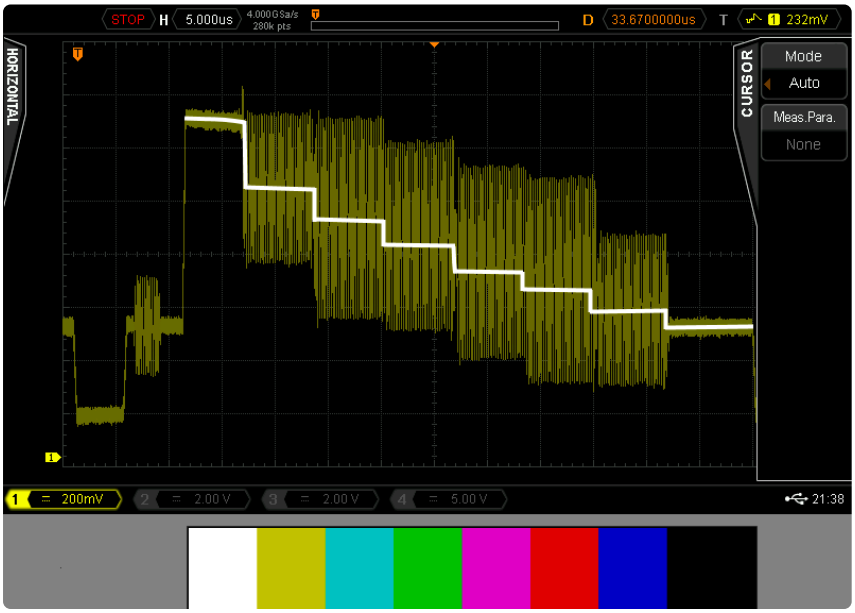


Figure 16. Oscillogramme d'un signal couleur PAL.



Figure 17. Changements de phase pendant le changement de couleur.

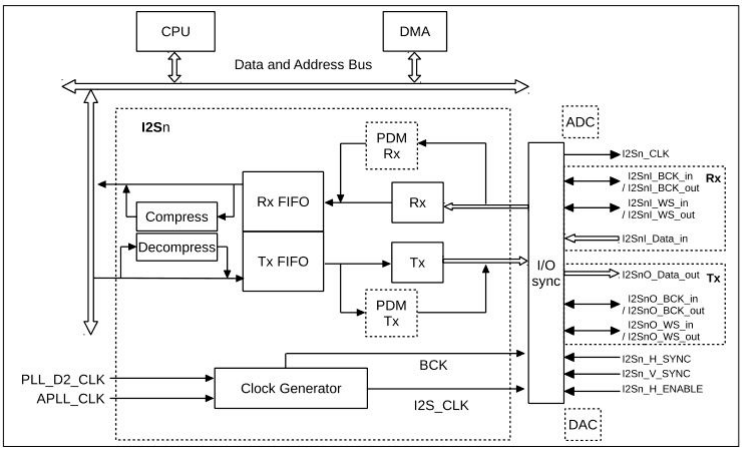


Figure 18. Le bloc I2S de l'ESP32. (Source : Espressif/tinyurl.com/yrrbnjak)

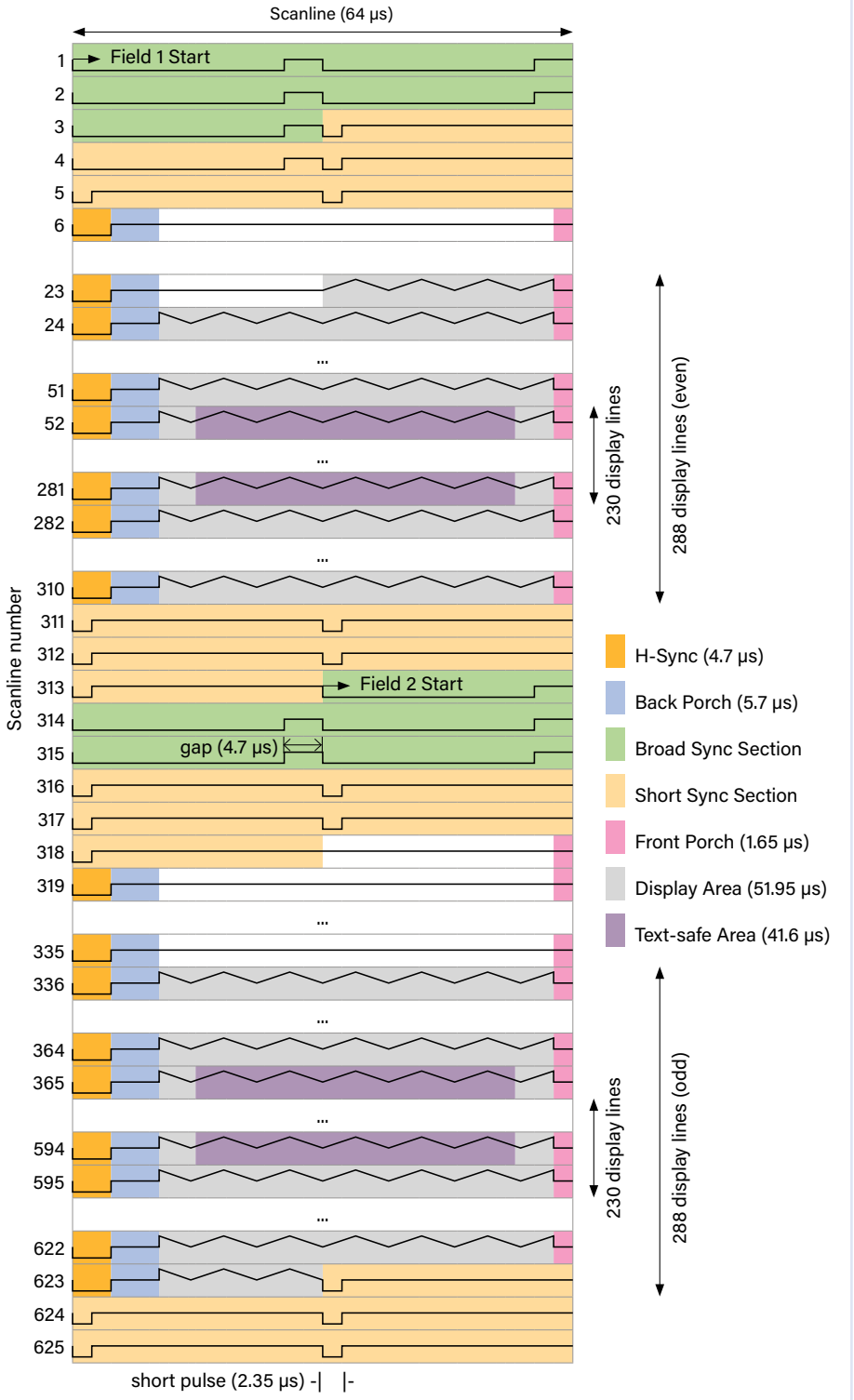


Figure 19. Signal de sortie d'image avec une demi-ligne de trame. (Source : batsocks.co.uk/https://tinyurl.com/4fychhmk)

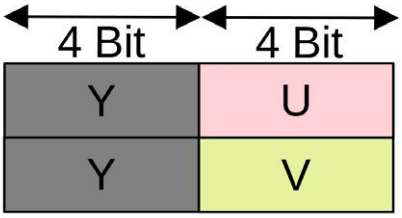


Figure 20. Stockage des informations sur les couleurs par le bitluni.

signal monochrome, un cœur de processeur de l'ESP32 est utilisé pour générer le signal vidéo. La configuration utilise le bloc I2S de l'ESP32 pour envoyer les données au CNA (**figure 18**).

Dans les systèmes PAL et NTSC, l'image est entrelacée, c'est-à-dire qu'elle est composée de deux trames successives avec les lignes impaires et les lignes paires. Comme nous l'avons déjà mentionné, cela permet de doubler le taux de rafraîchissement perçu (pour une même largeur de bande), ce qui réduit considérablement le scintillement perçu. La synchronisation de ces deux trames présente quelques particularités, comme le fait qu'il n'y a qu'une demi-ligne à la fin de l'image dans l'une des deux trames (**figure 19**).

Comme la RAM d'un ESP32 ne suffit que pour produire une image avec la moitié de la résolution, les lignes doivent être dessinées deux fois : le contenu de la première ligne apparaît donc également sur la deuxième. L'entrelacement garantit donc que le même contenu est dessiné deux fois. On peut alors se demander s'il faut continuer à émettre les deux trames ou s'il suffit d'utiliser deux fois la trame paire, ce qui simplifie considérablement le code d'émission. C'est cette astuce qui a déjà été utilisée par la Super Nintendo Entertainment System (SNES). Le code de Bitluni l'utilise également pour simplifier le code pour la synchronisation. Ainsi, seules 288 lignes réelles à 50 Hz sont émises (288p). Pour NTSC, le mode équivalent serait de 240 lignes à 60 Hz (240p).

Contrairement à la sortie d'images monochromes, les informations sur les couleurs doivent désormais être conservées dans la mémoire vive de l'ESP32. En général, les couleurs sont représentées par leurs composantes rouge, verte et bleue. Cependant, pour générer un signal composite conforme aux normes, il est préférable de stocker les informations de couleur dans la RAM sous forme de valeurs YUV. C'est le seul moyen pour l'ESP32 et son CNA de sortir ces données assez rapidement. Comme la quantité de RAM de l'ESP32 est limitée, il faut recourir à quelques astuces pour coder les données. bitluni stocke les informations dans la RAM sous forme de valeurs YU, YV et V combinées avec une résolution de 4 bits chacune (**figure 20**).

Malheureusement, avec l'approche de bitluni, la sortie vidéo composite n'est possible que pour le standard PAL, avec le NTSC cela ne fonctionne pas. Le rapport entre les 3,579 545 MHz de la salve de couleur NTSC et la fréquence d'échantillonnage du CNA est si défavorable qu'aucune salve de couleur utilisable ne peut être émise et que les récepteurs ne peuvent pas se synchroniser.



Figure 21. Emulateurs pour consoles de jeux 8 bits. (Source : tinyurl.com/ykd9ezap)

Couleur avec ESP32 en PAL et NTSC

Avec son projet « esp_8_bit », l'utilisateur de GitHub rossumur a prouvé qu'il est également possible de créer de la couleur conforme au NTSC avec un ESP32. Ce projet est une collection d'émulations basées sur l'ESP32 de diverses consoles 8 bits, de l'Atari 400 à la Nintendo Entertainment System et à la Sega Master System (**figure 21**).

Ce projet utilise simplement une broche de l'ESP32 pour sortir un signal composite en NTSC ou PAL. L'astuce est cachée dans le PLL audio de l'ESP32. Elle peut être utilisée pour faire fonctionner le CNA avec des fréquences d'échantillonnage allant jusqu'à environ 20 MHz. Le quadruple de la fréquence de la porteuse couleur NTSC, est 14,318 182 MHz (pour PAL 17,734 475 MHz). Avec l'APLL, on peut générer 14,318 180 MHz et 17,734 476 MHz, ce qui est assez proche des fréquences nécessaires pour le PAL et le NTSC. De cette façon, le CNA émet un multiple entier de la fréquence de la sous-porteuse couleur, ce qui donne des signaux vidéo traitables.

Avec l'APLL, le CNA de l'ESP32 peut non seulement produire un signal vidéo, mais cette approche est également très intéressante pour d'autres applications de synthèse numérique directe (DDS). Cependant, l'utilisation de l'APLL présente un inconvénient pour les fréquences d'échantillonnage élevées. Le CNA de l'ESP32 possède deux canaux. Si l'APLL alimente l'un d'entre eux avec des données pour la sortie vidéo et que l'on tente d'alimenter le second avec des données audio provenant de l'interface I2S, le CNA souffrira d'interruptions apparaissant sur les deux canaux, ce qui oblige à recourir à d'autres méthodes pour la sortie audio.

« espflix » a été développé basé sur le projet « esp_8_bit ». Il permet à l'ESP32 de jouer des vidéos qui sont stockées sur un serveur du projet espflix. Toutefois, cela nécessite quelques concessions sur le matériel vidéo. Des détails à ce sujet peuvent être trouvés sur la page Github associée [14].

Partie 2 : perspectives d'avenir

La deuxième partie de l'article traite du VGA, du DVI et des sprites. En particulier avec un ESP32 ou le RP2040 du Raspberry Pi Pico, on peut obtenir des effets surprenants. Si vous ne voulez pas attendre la deuxième partie, vous pouvez vous inscrire au webinaire *« Microcontroller as Pixel Artist »* [15], où les sujets VGA, DVI et sprites seront également abordés.

220047-04 — VF : Helmut Müller



Webinaire « Le microcontrôleur comme artiste du pixel »

Étant donné que les images animées en disent plus que les mots, surtout sur ce sujet, et que cet article ne pouvait offrir la place pour une revue étendue, nous vous invitons à participer au webinaire Elektor consacré aux applications graphiques avec Arduino, ESP32 et Raspberry Pi Pico. Les inscriptions sont d'ores et déjà ouvertes sur [15].
www.elektormagazine.com/webinars

Des questions, des commentaires ?

Envoyez un courriel à l'auteur (mathias.claussen@elektor.com) ou contactez Elektor (redaction@elektor.fr).

LIENS

[1] Vidéo composite : https://fr.wikipedia.org/wiki/Vid%C3%A9o_composite
[2] Livre en anglais « Vidéo analogique », Angelo La Spina : https://www.elektor.fr/analogue-video-e-book
[3] Bibliothèque Arduino TVOut : https://github.com/Avamander/arduino-tvout
[4] Hackvision : https://nootropicdesign.com/hackvision/
[5] Échelle de résistances : https://fr.wikipedia.org/wiki/%C3%89chelle_de_r%C3%A9sistances
[6] pico-composite8 : https://github.com/obstruse/pico-composite8
[7] Événement Hackaday : https://hackaday.com/2022/08/17/chips-remembered-the-scenix-ubicom-parallax-sx
[8] Couleur avec les puces SX : https://elinux.org/images/e/eb/Howtocolor.pdf
[9] Espace couleur YCbCr : https://fr.wikipedia.org/wiki/YCbCr
[10] Barres de couleur de l'UER : https://fr.wikipedia.org/wiki/YUV
[11] Contrôle de la teinte NTSC : https://en.wikipedia.org/wiki/Tint_control
[12] bitluni, « ESP32 Composite Video » : https://bitluni.net/esp32-composite-video
[13] Utilisateur Github rossumur : https://github.com/rossumur
[14] espflix : https://github.com/rossumur/espflix
[15] Webinaire « Le microcontrôleur comme artiste du pixel » : https://www.elektormagazine.com/webinars

Produits

- **Raspberry Pi Pico RP2040 (SKU 19562)**
  www.elektor.fr/19562
- **ESP32-DevKitC-32D (SKU 18701)**
  www.elektor.fr/18701
- **Arduino Uno Rev3 (SKU 15877)**
  www.elektor.fr/15877

electronica 2022

nouvelles du plus grand salon mondial de l'électronique

Stuart Cording et Jens Nickel

Plus de 70 000 visiteurs ont afflué à electronica en novembre dernier, on peut appeler cela un succès total. Il y avait parfois beaucoup de monde, du moins sur les stands des grands distributeurs et des fabricants de semi-conducteurs. Et ils n'ont pas fait dans la demi-mesure, pour le plus grand salon professionnel du monde de ce type, ils ont proposé des stands extravagants et des expositions attrayantes. On pouvait s'émerveiller devant des motos dotées d'IA intégrée, des camionnettes spéciales pour la détection de drones, beaucoup de voitures de course, et aussi un simulateur de quadricoptère pour une personne. Voici une petite sélection des nouvelles technologies passionnantes qui ont été présentées.



Aaronia, spécialiste de la technologie de mesure RF, a exposé ce fourgon pour la détection des drones. Le véhicule entièrement équipé est disponible pour environ 1,5 million d'euros, mais des solutions bien moins coûteuses pour la détection des aéronefs sans pilote sont également disponibles sous forme d'équipements montés en rack.
https://drone-detection-system.com



Les microscopes optiques sont de plus en plus remplacés par les microscopes numériques. Ils sont conviviaux, même pour les opérateurs peu familiarisés avec la microscopie, et permettent de scanner automatiquement les échantillons. Le modèle Olympus présenté ici fournit même des images en 3D qui peuvent être visualisées sous tous les angles.
https://olympus-ims.com/en/microscope/dsx1000/high-end-model



Alors que la réalité virtuelle à du mal à se généraliser, la réalité augmentée (RA) offre des avantages réels pour l'industrie. Les techniciens peuvent avoir accès à des instructions ou à des conseils projetés devant leurs yeux pendant qu'ils réparent une machine, tout en gardant une vue complète de leur environnement. Le module laser couleur de TDK, intégré à certains prototypes de lunettes de réalité augmentée, projette des images en couleur directement sur la rétine - et nous avons survécu pour raconter notre expérience !
www.tdk.com/de/news_center/press/20221013_01.html



C'était certainement le point fort du stand du fabricant de semi-conducteurs Infineon : ceux qui étaient un peu patients ont eu la chance de pouvoir essayer un vol virtuel avec ce simulateur de quadricoptère pour une personne.
https://infineon.com/cms/en/product/promopages/electronica/



Les affichages sur papier électronique (e-paper) ont le vent en poupe, car ils offrent une image nette, quel que soit l'angle de vision. Le distributeur Beck Elektronik a montré ces exemplaires, en monochrome et en couleur, de taille supérieure au format A4. En revanche, ces afficheurs sont (encore) chers par rapport aux afficheurs LCD. De plus, ils ne disposent pas de leur propre source de lumière, comme c'est le cas du papier imprimé.
https://beck-elektronik.de/en/products/displays/e-paper-display-epd



L'industrie automobile s'appuiera de plus en plus sur les logiciels pour définir les capacités de leurs véhicules, en utilisant du matériel standardisé et programmable. NXP a lancé son microcontrôleur S32K39, qui contient deux périphériques de commande de moteur dédiés. Prenant en charge des boucles de régulation de 100 kHz, ils sont prêts à être combinés avec des semi-conducteurs à large bande, tels que le carbure de silicium, pour fournir des onduleurs de traction à haut rendement pour les véhicules électriques.
https://elektor.link/NXPS32K



OKdo est une filiale de RS Components qui propose des cartes et des kits pour les fabricants et les professionnels. Les ordinateurs monocartes de la marque ROCK sont bien équipés, puissants et (pour la plupart) disponibles. Le nouveau Rock 5 modèle B possède une sortie vidéo 8K, qui a été utilisé pour une démonstration dans une application de surveillance.
https://okdo.com
https://wiki.radxa.com/Rock5



L'angoisse de l'autonomie est au centre des préoccupations des consommateurs alors qu'ils passent des voitures à carburant fossile aux voitures électriques. Dans l'espoir de dissiper ces craintes, Mercedes-Benz a récemment annoncé que sa VISION EQXX a effectué un voyage de 1 202 km (747 miles) avec une seule charge. Présenté sur le stand d'Onsemi, le véhicule utilise les modules VE-Trac SiC dans l'onduleur de traction, pour aider à fournir 95 % de l'énergie des batteries aux roues.
https://elektor.link/onsemiSiC

Les rédacteurs d'Elektor ont réalisé plusieurs entretiens en personne, avec des ingénieurs et des leaders d'opinion du secteur de l'industrie, lors du salon electronica 2022 à Munich. Visitez la chaîne Elektor TV pour regarder toutes les vidéos : *www.elektor.tv.*

220652-04 — VF : Laurent Rauber

caméra ESP32

si simple qu'on n'a même pas besoin du wifi

Bera Somnath (Inde)



Vous avez une idée d'application avec une caméra ? Avant d'en acheter une nouvelle en magasin, pourquoi ne pas en bricoler une vous-même ? Avec une carte ESP32 équipée d'une caméra et quelques composants supplémentaires, vous pouvez construire une solution personnalisée simple, mais efficace.

Tous les smartphones peuvent prendre et stocker des photos et les afficher. Par conséquent, personne n'aurait sérieusement l'idée de fabriquer encore un autre appareil photo, et certainement pas moi. Sauf si c'est vraiment facile à faire. Avec quelques modules peu coûteux comme une carte ESP32 avec caméra, un petit écran OLED, une horloge en temps réel et un mécanisme de déclenchement, vous pouvez en faire un qui fonctionne sur batterie et prendra des photos entièrement sous votre contrôle. En appuyant sur un bouton ou par détection de présence avec un capteur de mouvement ou même déclenché par un capteur de température sans contact, il prend (discrètement) la photo de quelqu'un dont la température corporelle est trop élevée. De telles applications ont du sens car vous ne pouvez utiliser votre smartphone pour cela.

Caméra ESP32

En ajoutant un capteur IRP à la carte ESP32-Cam, vous pouvez la configurer pour prendre secrètement des photos d'intrus ! Ou bien, en utilisant uniquement la caméra ESP32 et l'écran OLED, vous pouvez réaliser une petite webcam qui se connecte à n'importe quel réseau disponible dans le voisinage et publie ensuite le flux vidéo sur l'Intranet. De même, grâce à la fonction de redirection de port de votre modem ou routeur, vous pouvez publier l'image sur l'Internet. C'est une fonction très pratique pour ceux qui ont besoin d'une deuxième caméra pour leurs webinaires et/ou cours en ligne. Le module ESP32-Cam que nous utiliserons dans cet article coûte environ 10 €.

Une démo sympa pour commencer

Après avoir installé la dernière version de l'EDI Arduino et le paquetage de cartes ESP32 le plus récent en utilisant le Gestionnaire de cartes de l'EDI, sélectionnez comme carte la carte «AI Thinker ESP32-CAM» (Outils -> Type de carte -> ESP32 Arduino) et le port auquel elle est connectée (Outils -> Port). Ouvrez ensuite l'exemple de projet de la webcam ESP32 (Fichier -> Exemples -> ESP32 -> Caméra -> Serveur Web). Essayez-le, vous serez surpris. Les deux projets qui suivent sont des extensions de ce projet.

Projet 1 : Webcam avec écran OLED

Le projet webcam ne permet pas de sélectionner un réseau wifi à la volée (les informations d'identification du réseau sont codées en dur dans le programme) et la seule façon de connaître l'adresse IP de la webcam est de la connecter à un PC puis d'utiliser un terminal série. Ici, nous améliorons ce projet pour pouvoir se connecter à n'importe quel réseau disponible à proximité. Une fois la webcam connectée, le petit écran OLED indique le réseau auquel elle est connectée et son adresse IP. Si elle se déconnecte pour une raison quelconque, elle essaie de se reconnecter à l'un des réseaux connus.

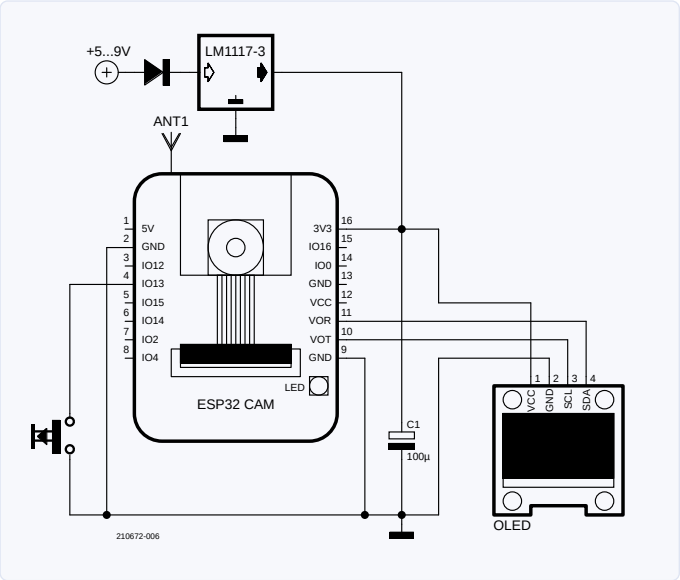


Figure 1. Ajoutez un écran OLED à la caméra ESP32-Cam pour plus de commodité.

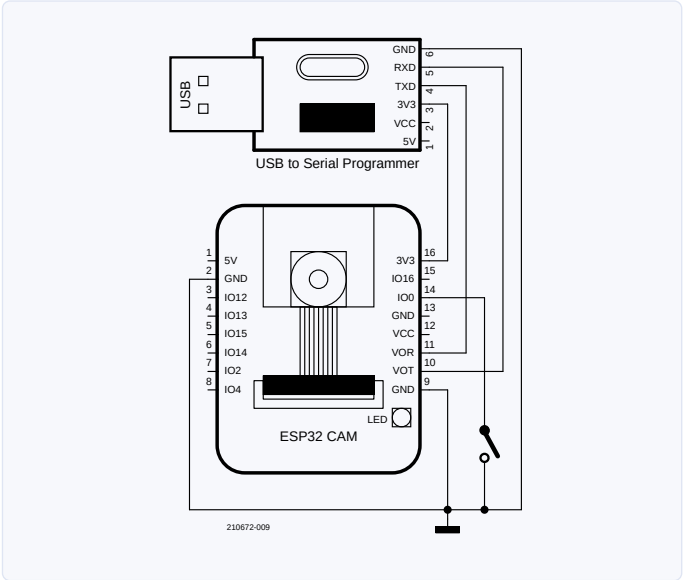


Figure 2. L'ESP32-Cam n'a pas d'interface USB-série intégrée. Par conséquent, pour télécharger le croquis, il faut disposer d'un adaptateur USB-série. Pour entrer dans le mode de chargement du programme, il faut appuyer sur le bouton RST de l'ESP32 (ou l'éteindre et le rallumer) pendant que IO0 est forcé à la masse.

Fabrication de la Webcam

La **figure 1** montre comment connecter l'écran OLED à la carte ESP32-Cam. Il reste très peu de broches GPIO utilisables après la connexion de la carte SD et de la caméra. Les seules broches libres disponibles pour notre utilisation sont IO1, IO3, IO4, IO12, IO13 (voir encadré « Notes du labo d'Elektor »). IO1 (« U0T ») et IO3 (« U0R ») sont également utilisées comme port série pour télécharger le programme vers l'ESP32 en mode de programmation flash. Par conséquent, ces broches doivent être libres de toute connexion pendant le téléchargement des programmes.

Dans ce projet de webcam, nous pouvons facilement utiliser les broches réservées à la carte SD car nous n'utilisons pas cette fonction. Cependant, nous avons utilisé IO1 et IO3 comme broches I²C pour connecter l'écran OLED. Nous avons ainsi maximisé le nombre de broches GPIO disponibles. N'oubliez pas de déconnecter le bus I²C lorsque vous téléchargez un programme sur l'ESP32 (**figure 2**).

Utilisation

Le principe de fonctionnement de la webcam ESP32 avec écran OLED est resté très simple. L'ESP32 recherche les réseaux wifi fournis sous forme de liste dans le programme et essaie de se connecter à l'un d'eux. Dès que la connexion réussit, le statut « camera ready » s'affiche sur l'écran OLED avec le SSID du réseau et l'adresse IP attribuée à la webcam. Pour accéder à la caméra, connectez-vous au réseau indiqué et faites pointer un navigateur sur l'adresse IP de la webcam. Par défaut, l'adresse du port reste 8080.

On peut télécharger le logiciel pour la webcam ESP32 avec écran OLED à partir de [1]. Notez qu'il nécessite la bibliothèque « *ESP8266 and ESP32 OLED driver for SSD1306 displays* » (nous avons utilisé la version 4.1.0) [2]. Elle peut être installée avec le gestionnaire de bibliothèque de l'EDI Arduino. N'oubliez pas de déconnecter le bus I²C avant de télécharger un croquis.

Projet 2 : Caméra à déclenchement avec OLED & RTC

Dans ce projet, une photo est prise chaque fois qu'un événement se produit. Les photos sont stockées sur une carte SD avec un numéro de série et un horodatage.

Le circuit (**figure 3**) utilisé est une extension du circuit précédent. Un module d'horloge en temps réel (RTC) DS3231 (ou DS1307) est ajouté sur le bus I²C, et une source de déclenchement est connectée à IO13. Il peut s'agir d'un bouton poussoir ou d'un capteur PIR

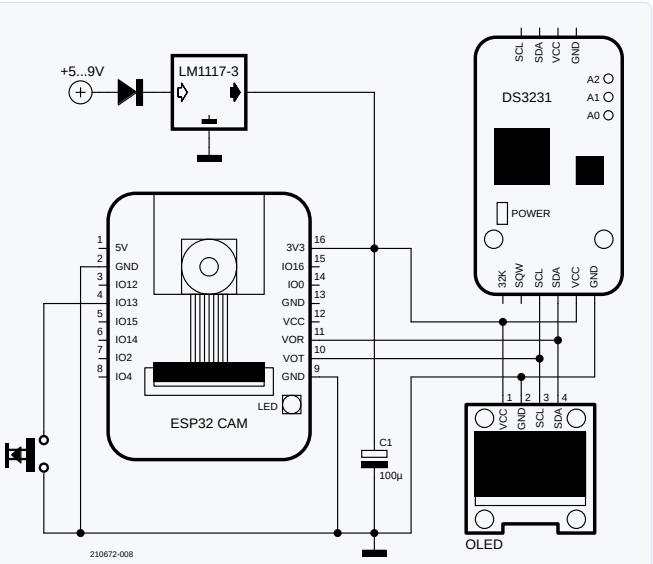


Figure 3. Un bouton poussoir et un module d'horloge en temps réel transforment la webcam de la figure 1 en une caméra de surveillance à déclenchement et horodatage.

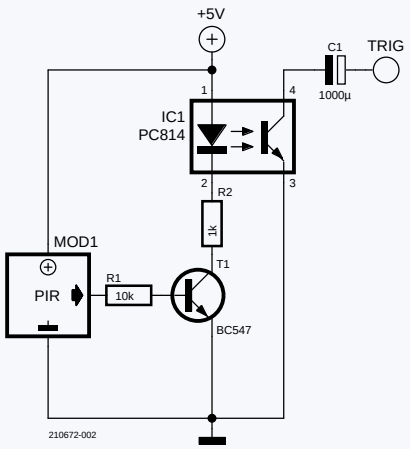


Figure 4. On peut utiliser un capteur IRP en 5 V - certains fonctionnent même en 3,3 V - pour déclencher la caméra ESP32. Cependant, pour l'isoler complètement, nous recommandons d'utiliser un optocoupleur comme celui-ci (ou un relais).

(**figure 4**) ou de tout autre élément qui génère une courte impulsion active-basse lorsqu'une photo doit être prise.

IO4 est utilisé comme flash, en utilisant la LED blanche super lumineuse intégrée de l'ESP32-Cam comme source d'éclairage supplémentaire pour la caméra. IO13 est utilisé comme broche de déclenchement pour prendre des photos. Le programme appose la date et l'heure sur le nom du fichier et le stocke sur la carte SD. Il passe ensuite en mode de sommeil profond pour préserver la batterie pendant les périodes d'inactivité où aucune photo n'est prise. Une impulsion active-basse sur IO13 réveille la caméra pour prendre une nouvelle photo.

Ce programme se compose uniquement de la fonction `setup` ; la fonction `loop` n'est pas utilisée. Ceci est dû à l'utilisation du mode de sommeil profond. Lorsque l'appareil se réveille de ce mode, il redémarre, exécute la fonction `setup` et prend une photo avant de retourner au mode de sommeil profond, de sorte que la fonction `loop` n'est jamais atteinte.

On peut télécharger le logiciel pour la caméra à déclenchement à partir de [1]. En plus de la bibliothèque de pilotes OLED requise pour le premier projet [2], ce projet nécessite également la bibliothèque « Rtc by Makuna » (nous avons utilisé la version 2.3.5) [3]. Cette bibliothèque peut également être installée avec le gestionnaire de bibliothèque de l'EDI. N'oubliez pas de déconnecter le bus I²C avant de télécharger un croquis (**figure 2**).

Conclusion

La caméra ESP32-Cam est très bon marché tout en offrant de vastes possibilités. En alignant plusieurs de ces caméras, on peut s'en servir comme scanner fixe et grand angle (50 à 100 mètres) pour prendre des photos des plaques d'immatriculation des véhicules qui s'approchent afin de les contrôler (pour la vitesse par ex.). En mode solo, avec un capteur PIR, elle peut servir à « capturer » des oiseaux ou d'autres animaux. On peut aussi imaginer des applications plus élaborées. Par exemple, en ajoutant la détection de visage et un scanner de température sans contact, elle peut contribuer à une certaine sécurité anti-COVID.

210672-04 — VF : Denis Lafourcade

Des questions, des commentaires ?

Contactez Elektor (redaction@elektor.fr).

COMPOSANTS

Carte ESP32-Cam
Carte SD
Écran OLED de 0,96 pouce compatible I²C avec SSD1306
Module RTC DS3231 / DS1307 I²C
Adaptateur USB-série (câble)
Régulateur 3,3 V LM1117
Capteur PIR ou bouton poussoir
En option : batterie, condensateurs, etc.



PRODUITS

- **Carte de développement ESP32-Cam-CH340(SKU 19333)**
  www.elektor.fr/19333
- **Écran OLED de 0,96 pouce compatible I²C (SKU 18747)**
  www.elektor.fr/18747
- **Module RTC de SparkFun – RV-8803 (Qwiic) (SKU 19646)**
  www.elektor.fr/19646

LIENS

[1] Téléchargements pour cet article à Elektor Labs : https://www.elektormagazine.fr/labs/esp32-camera
[2] Pilote OLED ESP8266 et ESP32 pour écrans SSD1306 : https://github.com/ThingPulse/esp8266-oled-ssd1306
[3] Rtc par Makuna : https://github.com/Makuna/Rtc
[4] Caméra à déclenchement version Elektor Labs : https://www.elektormagazine.com/210672-01

Notes du labo d'Elektor

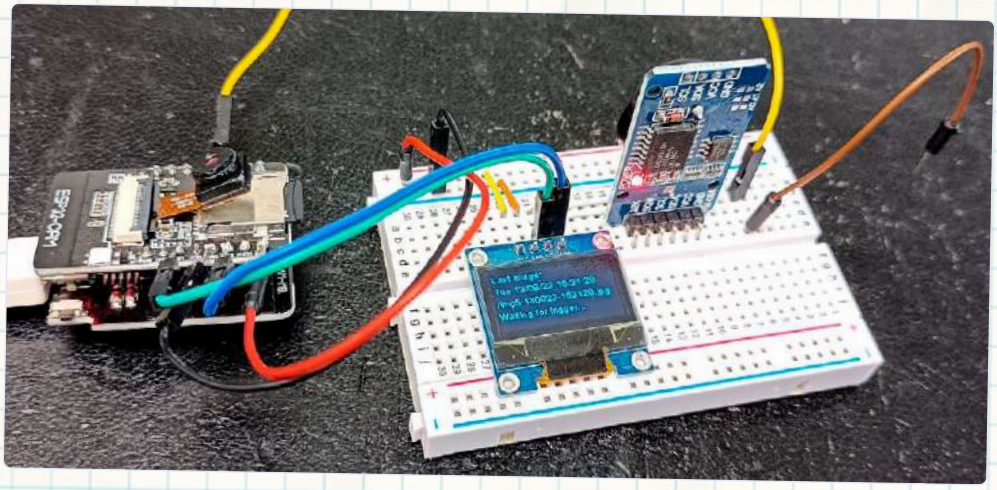


Figure 5. Le prototype Elektor Labs construit sur une platine d'expérimentation. L'ESP32-Cam avec la carte fille est à gauche, le module vertical à droite est la RTC.

De nombreuses cartes ESP32-Cam sont vendues avec une ingénieuse carte fille convertisseur USB-série* (on devrait peut-être l'appeler carte mère puisque le module ESP32 s'y branche et en tire son alimentation). Ce combo (figure 5) rend la programmation de l'ESP32 très facile, mais il entre en conflit avec le port I²C sur les ports IO1 et IO3 tels qu'utilisés dans le projet de l'auteur. En utilisant ces broches, le port I²C doit être déconnecté à chaque fois que l'on veut reprogrammer l'ESP32, et nous avons donc cherché un autre moyen de connecter le bus I²C.

En théorie, sur l'ESP32, le bus I²C peut être configuré pour utiliser presque n'importe laquelle de ses broches d'entrée/sortie, mais cette configuration sur la carte ESP32-Cam entraîne toutes sortes de problèmes de démarrage et d'erreurs de mémoire PSRAM. Ceci à cause des résistances de rappel sur le bus I²C.

La carte ESP32-Cam possède deux connecteurs d'extension à 8 voies. À première vue, cela peut sembler correct, mais en y regardant de plus près, les choses ne sont pas si simples. Tout d'abord, six des seize broches sont utilisées pour l'alimentation, ce qui n'en laisse que dix pour les ports d'entrée/sortie. Six d'entre eux (IO2, IO4, IO12, IO13, IO14 et IO15) sont partagés avec le connecteur de la carte SD. Il reste donc les ports IO0, IO1, IO3 et IO16.

Le téléchargement d'un programme sur l'ESP32 avec la carte fille attachée nécessite les ports IO0, IO1 et IO3.

IO16 est partagée avec la broche CS de la puce PSRAM, elle ne peut donc être utilisée en toute sécurité que lorsque la PSRAM n'est pas nécessaire.

La bibliothèque de la carte SD utilise par défaut le mode de données 4 bits, mais elle peut être mise en mode de données 1 bit avec

```
SD_MMC.begin(«/sdcard»,true);
```

Cela libère les ports IO4, IO12 et IO13. Le port 4 est partagé avec la (lumineuse) LED flash blanche. Il possède une résistance de rappel haut de 47 kΩ et une résistance de rappel bas de 10 +1 kΩ qui pilote un transistor. Le port 13 possède également une résistance de rappel de 47 kΩ et est utilisé dans ce projet comme entrée de déclenchement de la caméra.

IO12 est l'une des broches de l'ESP32 qui doit être manipulée avec précaution à la mise sous tension car elle détermine la valeur de VDD_SDIO, la tension d'interface de la mémoire flash. Elle doit être forcée vers le bas à la mise sous tension de la carte ESP32-Cam.

Après quelques tâtonnements, nous avons découvert que le bus I²C peut être déplacé en toute sécurité vers les ports IO0 (SDA) et IO3 (SCL, ou l'inverse. Peu importe tant que le logiciel est configuré correctement). En utilisant ces ports, le bus I²C n'interfère pas avec le port de programmation série ni avec le mode de démarrage, car IO0 et IO3 sont censés être à l'état haut à la mise sous tension.

Enfin, et sans rapport avec ce qui précède, sachez que IO33 est connecté à une LED rouge du côté du module ESP32 de la carte, utilisable comme bon vous semble (tant que vous gardez à l'esprit qu'elle est active à l'état bas).



Figure 6. Ça marche, bravo !

Nous nous sommes un peu emportés avec le deuxième projet et avons modifié le programme (figure 6). Il n'utilise plus l'EEPROM émulée pour stocker le numéro de l'image mais un fichier sur la carte SD. Il suffit de supprimer le fichier « counter.txt » pour remettre le compteur à zéro. On peut télécharger notre programme à partir de [4].

** Voir l'encadré Produits connexes pour une autre version du module ESP32-Cam.*

alimentation ATX pour Raspberry Pi

Sébastien Guerreiro de Brito (France)

Vous avez un RPi qui doit être alimenté, ainsi qu'une vieille alimentation ATX ? Un circuit imprimé, quelques composants et un ATtiny feront l'affaire.



Figure 1. Alimentation ATX. Source : Shutterstock.

L'alimentation pour PC au format ATX (**figure 1**) a été introduite par Intel en 1995. Depuis les processeurs de type Pentium II, c'est le facteur de forme le plus répandu pour les alimentations à découpage pour PC. Toutes ses caractéristiques électroniques, mécaniques, environnementales et autres sont spécifiées par Intel et constituent la norme ATX.

Le but de ce circuit est d'utiliser une alimentation ATX pour alimenter une carte RPi ainsi que les nombreux périphériques qui peuvent y être connectés. L'avantage de cette alimentation est le nombre de tensions différentes disponibles - à des niveaux de puissance intéressants.

Le connecteur d'alimentation ATX

Nous n'allons pas décrire ici le fonctionnement interne de l'alimentation à découpage ATX, mais seulement essayer de la faire fonctionner. Pour commencer, examinons en détail le connecteur (**figure 2**) qui doit être relié à la carte mère du PC.

Le connecteur est composé de

- un connecteur femelle Molex Mini-Fit Jr à 24 positions et deux rangées (réf : 39-01-2240) ou équivalent ;
- des contacts femelles Molex Mini-Fit HCS (réf : 44476-1112)

Le **tableau 1** présente le brochage de ce connecteur.

Les signaux intéressants pour piloter l'alimentation sont PWR_OK et PS_ON#. L'alimentation +5 VSB sera également utilisée dans notre montage.

Le signal PWR_OK est un signal « Power good » utilisé par l'alimentation pour indiquer au système que les sorties +5 VCC, +3,3 VCC et +12 VCC sont présentes et conformes aux seuils.

Le signal PS_ON# est celui qui va nous permettre de démarrer l'alimentation. C'est un signal compatible TTL, actif-bas, qui permet à la carte mère de contrôler l'alimentation à distance. Cela permet l'allumage et l'extinction de l'alimentation par logiciel, le Wake-on-LAN, le wake-on-USB, etc.

Lorsque PS_ON# est tiré vers le bas, l'alimentation doit activer les quatre principaux rails d'alimentation CC : +12 VCC, +3,3 VCC, +5 VCC et -12 VCC. Lorsque PS_ON# est tiré vers un niveau TTL haut, ou un circuit ouvert, les rails de sortie CC ne doivent pas délivrer de courant et doivent être maintenus à un potentiel nul par rapport à la masse. Une alimentation ATX dispose d'un circuit antirebond interne pour empêcher les oscillations

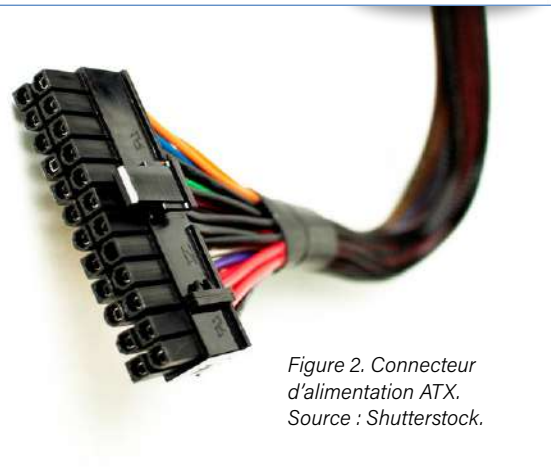


Figure 2. Connecteur d'alimentation ATX. Source : Shutterstock.

Pin	Signal	Wire color
1	+3.3 VDC	
2	+3.3 VDC	
3	COM	
4	+5 VDC	
5	COM	
6	+5 VDC	
7	COM	
8	PWR_OK	
9	+5 VSB	
10	+12 V1DC	
11	+12 V1DC	
12	+3.3 VDC	
13	+3.3 VDC	
14	–12 VDC	
15	COM	
16	PS_ON#	
17	COM	
18	COM	
19	COM	
20	Reserved	NC
21	+5 VDC	
22	+5 VDC	
23	+5 VDC	
24	COM	

Tableau 1. Brochage du connecteur ATX.

de mise sous/hors tension si le signal PS_ON# est activé par un interrupteur mécanique.

Les caractéristiques du signal PS_ON# sont énumérées dans le **tableau 2** et présentées sous forme de graphiques à la **figure 3**.

Le signal PS_ON# n'a aucun effet sur la sortie +5 VSB. La tension +5 VSB (SB = standby) est présente tant que l'alimentation est connectée au secteur. Cette sortie permet d'alimenter les circuits qui doivent rester opérationnels lorsque les rails d'alimentation principaux sont désactivés.

Description de la carte électronique

Pour permettre à l'alimentation ATX d'être commandée par le RPi, on a conçu un circuit imprimé (PCB) [1]. Il contient la logique nécessaire pour permettre le démarrage et l'arrêt du RPi en y ajoutant une petite UC. Cela offre aussi d'autres avantages intéressants. Comme nous disposons d'une alimentation ATX, nous utiliserons le rail 5 V pour ajouter un jeu de ports USB au PCB, ce qui permettra aux périphériques exigeant des courants plus élevés d'en profiter. La **figure 4** montre le PCB en action.

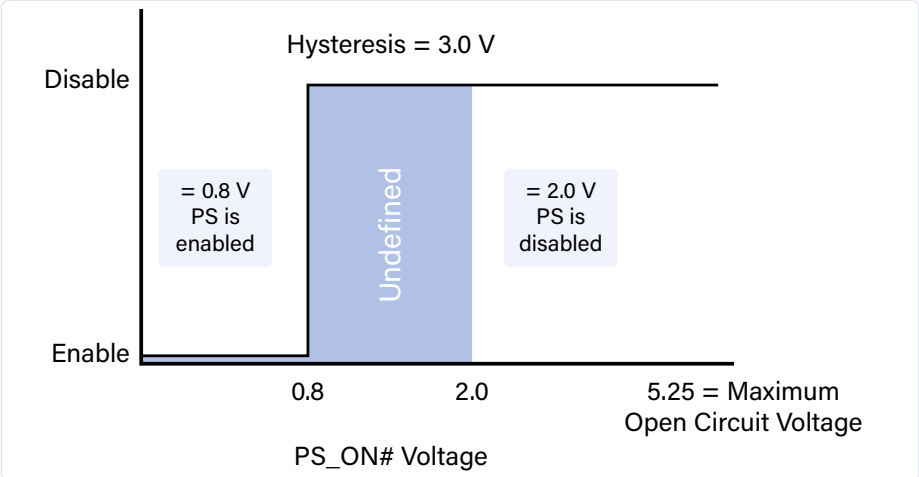


Figure 3. Caractéristiques du signal PS_ON#. (Source : ATX/ATX12V Power Supply Design Guide Version 1.1, Section 3.3.2, figure 3)

Paramètre	Minimum	Maximum
VIL	0 V	0.8 V
IIL	–	–1.6 mA[1]
VIH	2.0 V	–
VIH en circuit ouvert	–	–5.25 V
Ondulation/ Bruit		400 mV crête

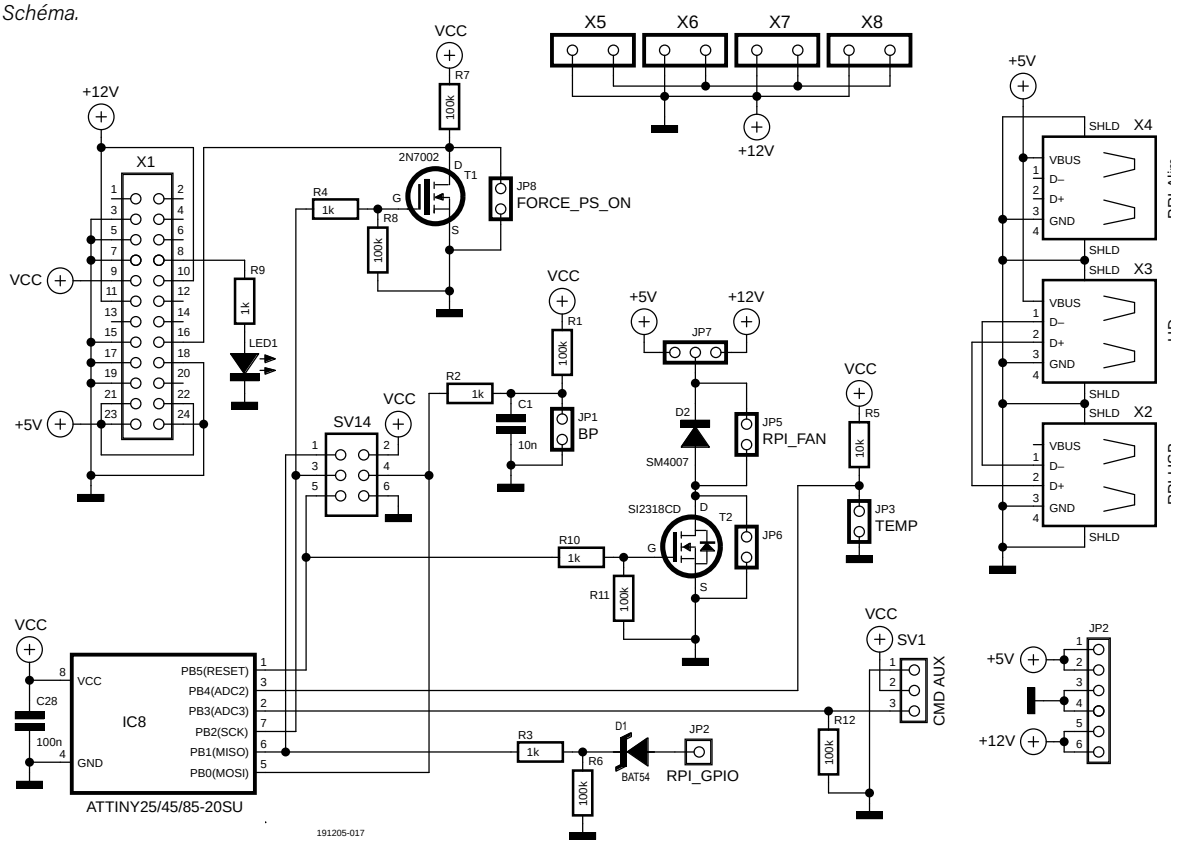
[1] Remarque : un courant négatif indique que le courant circule de l'alimentation à la carte mère.

Tableau 2. Caractéristiques du signal PS_ON#.



Figure 4. Alimentation ATX connectée au PCB.

Figure 5. Schéma.



On peut voir le schéma du PCB à la **figure 5**. Le principe de commande de l'alimentation ATX est relativement simple : Il est basé sur l'utilisation d'un microcontrôleur ATtiny85 alimenté directement par +5 VSB. L'avantage d'utiliser un microcontrôleur est que, bien que la commande de l'alimentation soit simple, on peut y ajouter d'autres fonctionnalités.

Démarrage de l'alimentation électrique

L'ATtiny reçoit l'ordre de démarrer l'alimentation par un bouton poussoir externe soudé au cavalier JP1.

Lorsque le bouton poussoir est activé, le microcontrôleur commande le transistor T1, qui démarre l'alimentation. Le cavalier JP8 permet de démarrer l'alimentation en mode automatique.

Gestion de la température du processeur du RPi

Nous dotons la carte du RPi d'un refroidissement par air supplémentaire. Pour cela, on y place dessus une thermistance NTC de 10 kΩ et, selon la température mesurée, la carte déclenchera un ventilateur. On sélectionne la tension du ventilateur (12 V ou 5 V) avec le cavalier JP7.

Communication avec le RPi

Pour pouvoir éteindre l'alimentation ATX depuis le RPi, comme sur un PC, nous avons mis en place un petit stratagème.

Le signal RPI_GPIO est connecté au GPIO27 sur la carte RPi. On voit le connecteur GPIO du RPi sur la **figure 6**.

Au niveau du système d'exploitation, il faut d'abord s'assurer que le GPIO27 est accessible. Pour cela, nous créons le fichier d'initialisation de GPIO (exécuté au démarrage) avec l'éditeur nano :

```
nano S75gpioinit
```

Ensuite, nous entrons le code du **listage 1**.

Nous rendons le fichier exécutable et le plaçons dans le dossier d'initialisation.

```
sudo chmod +x S75gpioinit
sudo mv S75gpioinit /etc/init.d
```

Vous remarquerez dans le script d'initialisation que le GPIO27 est initialisé avec une valeur de 1. Ainsi, lorsque nous éteindrons le

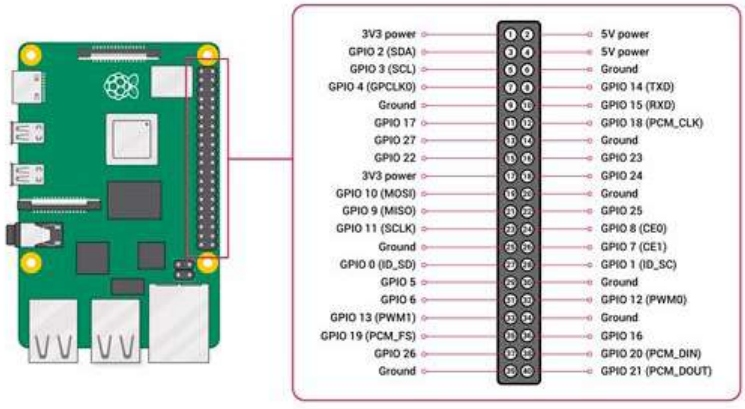


Figure 6. Brochage du RPi (Source : [2]).

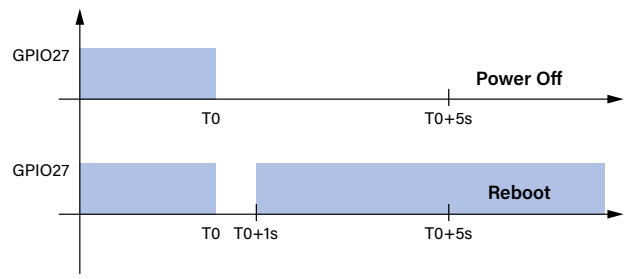


Figure 7. Gestion de la broche GPIO27.

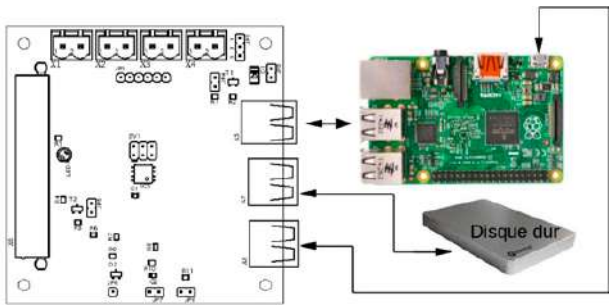


Figure 8. Connexion de la carte RPi.

RPi via le système d'exploitation (poweroff, par exemple), cette entrée repassera à 0 et le microcontrôleur saura que c'est le moment d'éteindre l'alimentation ATX.

Mais, plus amusant, nous voulons pouvoir redémarrer la carte RPi par réinitialisation de l'alimentation électrique. Pour cela, l'astuce que nous utilisons est de faire varier le GPIO27 afin de signaler au microcontrôleur que ce n'est pas une mise hors tension, mais une réinitialisation, qui est demandée.

On peut voir sur la **figure 7** comment est géré le GPIO27 : Lorsque le microcontrôleur voit la broche GPIO27 à 0, il attend 5 secondes, puis, si la broche est toujours à 0, cela signifie que le RPi a été éteint, donc il éteint l'alimentation ATX. Sinon, c'est qu'il reçoit une commande de redémarrage et provoque l'arrêt de l'alimentation ATX pendant 2 secondes, avant de la redémarrer.

Pour que tout cela soit possible, nous devons

Listage 1. Script Shell.

```
#!/bin/sh
 ##################################################
 # IO Init Script #
 # Author : Sebastien Guerreiro #
 # Versions : #
 # ----------- #
 # Mars 2019 V1.0 Création #
 ##################################################
CHEM_GPIO=/sys/class/gpio
#-------------------------------------------------
 register()
 {
 #Registers the outputs
 if [ $SENS="out" ]; then
 #Outputs
 echo "$NUM" > $CHEM_GPIO/export
 echo "$SENS" > $CHEM_GPIO/gpio$NUM/direction
 echo "$VALEUR" > $CHEM_GPIO/gpio$NUM/value
 chmod g+w $CHEM_GPIO/gpio$NUM/value
 else
 #Inputs
 echo "$NUM" > $CHEM_GPIO/export
 echo "$SENS" > $CHEM_GPIO/gpio$NUM/direction
 fi
 ret=$?;
 if [ $ret -eq 0 ]; then
 #echo_success;
 echo "Registering GPIO$NUM : OK"
 else
 # echo_failure;
 echo "Registering GPIO$NUM : ERROR"
 exit $ret;
 fi
 }
#-------------------------------------------------
 unregister()
 {
 echo "$NUM" > /sys/class/gpio/unexport
 ret=$?;
 if [ $ret -eq 0 ]; then
 echo_success;
 else
 echo_failure;
 exit $ret;
 fi
 }
#-------------------------------------------------
 start()
 {
 echo "Registering GPIO--Setting Power GPIO ON"
 #Output for power GPIO
 NUM=27; SENS=out ; VALEUR=1
 register;
 }
 #-------------------------------------------------
start
exit 0
```

Listage 2. Le script d'arrêt.

```
#!/bin/sh
#########################################################
# Reboot Poweroff Management #
# Author : Sebastien Guerreiro (www.sebelectronic.com)#
# Versions : #
# ----------- #
# Fev 2020 V1.0 Création #
#########################################################
case $1 in
-r)
echo 0 > /sys/class/gpio/gpio27/value
sleep 1
echo 1 > /sys/class/gpio/gpio27/value
sleep 1
/sbin/shutdownSeb $@
;;
*)
/sbin/shutdownSeb $@
;;
esac
```

Listage 3. Le script de redémarrage.

```
#!/bin/sh
#########################################################
# Reboot Management #
# Author : Sebastien Guerreiro #
# Versions #
# ----------- #
# Fev 2020 V1.0 Création #
#########################################################
echo 0 > /sys/class/gpio/gpio27/value
sleep 1
echo 1 > /sys/class/gpio/gpio27/value
sleep 1
#/sbin/rebootSeb
/sbin/shutdownSeb -r now
```

PRODUITS

- **Raspberry Pi 4 2 GB (SKU 18965)**
  www.elektor.fr/18965
- **Velleman VTSS220 station de soudage avec contrôle de température (SKU 19865)**
  www.elektor.fr/19865

intervenir avec quelques commandes Linux. Du côté du système d'exploitation, nous renommons les instructions shutdown et reboot contenues dans sbin en shutdownSeb et rebootSeb, respectivement

```
sudo mv /sbin/shutdown /sbin/shutdownSeb
sudo mv /sbin/reboot /sbin/rebootSeb
```

Ensuite, nous créons deux scripts : shutdown et reboot, que nous plaçons dans le dossier `/sbin`. Dans le terminal, tapez :

```
nano shutdown
```

et saisissez ce qui suit (**listage 2**) dans le fichier. Enregistrez le fichier et quittez nano.

Dans le terminal, tapez :

```
sudo cp shutdown /sbin
```

Ceci copiera votre script dans `/sbin` afin que vous puissiez l'exécuter plus tard.

Ensuite, nous devons faire le script de reboot. Toujours dans le terminal, entrez :

```
nano reboot
```

et saisissez le script suivant (**listage 3**) dans ce fichier. Enregistrez le fichier et quittez nano. Ensuite, dans le terminal, tapez :

```
sudo cp reboot /sbin
```

Ceci copiera votre script de redémarrage dans `/sbin` et lui permettra d'être exécuté plus tard.

C'est tout !

Connexion des périphériques

L'utilisation d'une alimentation ATX nous permet de connecter la carte à un disque dur IDE, par exemple, ou à un lecteur de CD-ROM IDE dans mon cas. En bref, l'avantage est que vous pouvez utiliser les périphériques d'un vieux PC pour vos nouveaux projets.

LIENS

[1] Projet sur Elektor Labs : https://elektormagazine.fr/labs/atx-powersupply-for-rpi
[2] Image Source : https://raspberrypi.com/documentation/computers/images/GPIO-Pinout-Diagram-2.png

La connexion se fait comme vous pouvez le voir sur la **figure 8**.

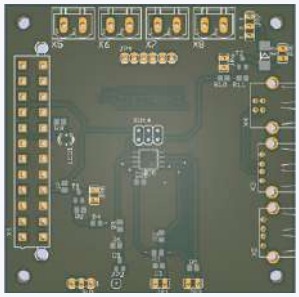
Si vous souhaitez en savoir plus sur le projet, vous pouvez visiter la page Elektor Labs [1].

(191205-04) — VF : Denis Lafourcade

Des questions, des commentaires ?

Vous avez des questions ou des commentaires sur cet article ? Contactez Elektor à l'adresse redaction@elektor.fr ou laissez un commentaire sur la page des projets à Elektor Labs.

LISTE DES COMPOSANTS



Résistances (0805, 0.1 W)
R1, R3, R6, R7, R8 = 1 k, 1 %
R2, R4, R5, R9, R10 = 100 k, 1 %
R11 = 10 k, 1 %

Condensateurs
C1 = 100 nF, 0805
C2 = 10 nF, 0805

Semi-conducteurs
D1 = SM4007
D2 = BAT54
IC1 = ATTINY85
T1 = SI2318CDS
T2 = 2N7002
LED1 = LED, verte, 3 mm

Divers
SV1, JP7 = connecteur mâle 3 points, pas 2,54 mm
JP1, JP3, JP5, J6, JP8 = connecteur mâle 2 points, pas 2,54 mm
JP2 = connecteur mâle 1 point, pas 2,54 mm
SV14 = connecteur mâle 2 rangées, 6 points, pas 2,54 mm
X5, X6, X7, X8 = connecteur MSTBVA 2.5 / 2-G-5.08 à 2 broches
X4 = Connecteur USB Type B
X1 = Connecteur Molex Mini-Fit Jr droit, 5566, 24 voies, 2 rangées
X3, X2 = Connecteur USB Type A
Circuit imprimé

Advertisement

Rejoignez la communauté Elektor

- accès à l'archive numérique depuis 1978 !
- 8x magazine imprimé Elektor
- 8x magazine numérique (PDF)
- 10 % de remise dans l'e-choppe et des offres exclusives pourles membres
- accès à plus de 5000 fichiers Gerber



Devenez membre **maintenant !**

elektor GOLD membership



www.elektormagazine.fr/membres

elektor

amplificateur pour casque 32 Ω

une solution simple mais de qualité à trois CI

Thierry Clinquart (Belgique)

Par le passé, la plupart des bons casques avaient une impédance de 600 Ω. Aujourd'hui, les écouteurs de 32 Ω dominent dans les catégories de prix moyennes et supérieures. Par conséquent, les amplificateurs de casque modernes ont besoin d'un peu plus de puissance.

Alors qu'un un ampli-op basique est suffisant pour piloter à fort volume des casques de 600 Ω, des courants nettement plus élevés circulent à volume comparable dans les bobines des casques modernes de 32 Ω, ce qui provoque l'écrêtage des ampli-op ordinaires. La solution classique avec les anciens mais bons amplificateurs opérationnels audio NE5532 et les circuits intégrés comparables est donc obsolète. Pour que les casques modernes n'aient pas un son déformé, il faut donc modifier le circuit classique des ampli-op. Cela est très facile aujourd'hui, non seulement en principe mais aussi en pratique. Tout ce dont vous avez besoin est de rajouter un BUF634A par canal.

Le BUF634A

Il s'agit essentiellement d'un étage amplificateur de puissance push-pull petit et rapide dans un boîtier de circuit intégré. Il suffit de le connecter derrière un ampli-op et de l'inclure dans la boucle de rétroaction négative – c'est tout !

La **figure 1** montre le circuit stéréo complet. Un ampli-op audio double de type OPA2134 [1] forme les deux étages d'entrée et est responsable de l'amplification de la tension. Deux BUF634As s'occupent de l'amplification du courant. Son circuit interne de base peut être admiré dans la **figure 2**.

La particularité de cet étage de sortie push-pull intégré est sa bande passante très élevée, qui peut être réglée entre 35 et 210 MHz via le courant traversant la broche 1. Si la résistance Rx de la **figure 1** est

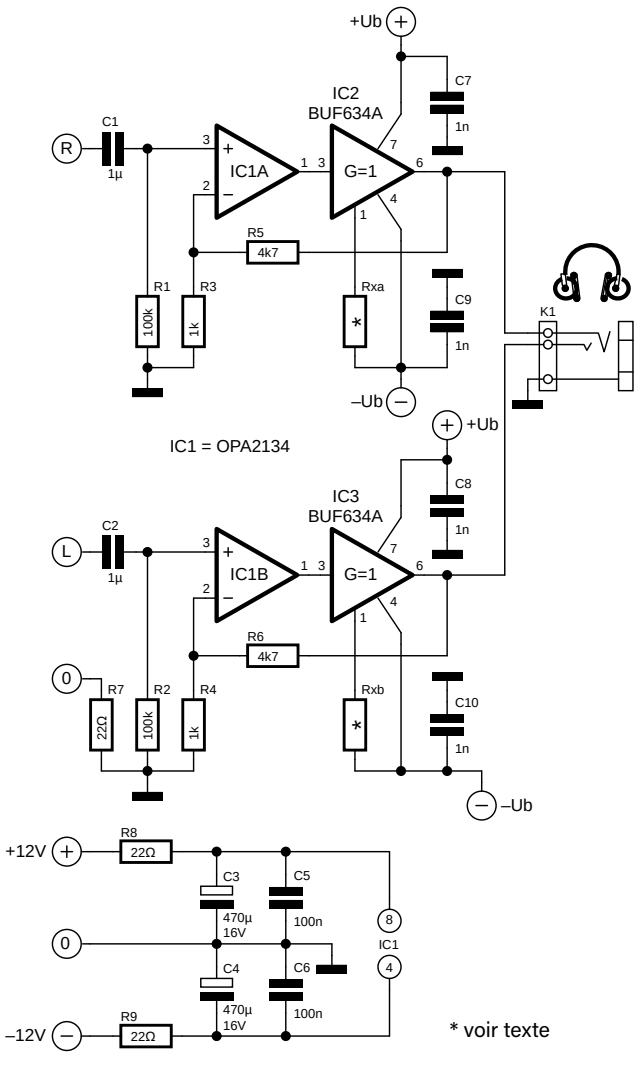


Figure 1. Le circuit complet de l'amplificateur de casque repose sur des semi-conducteurs intégrés.

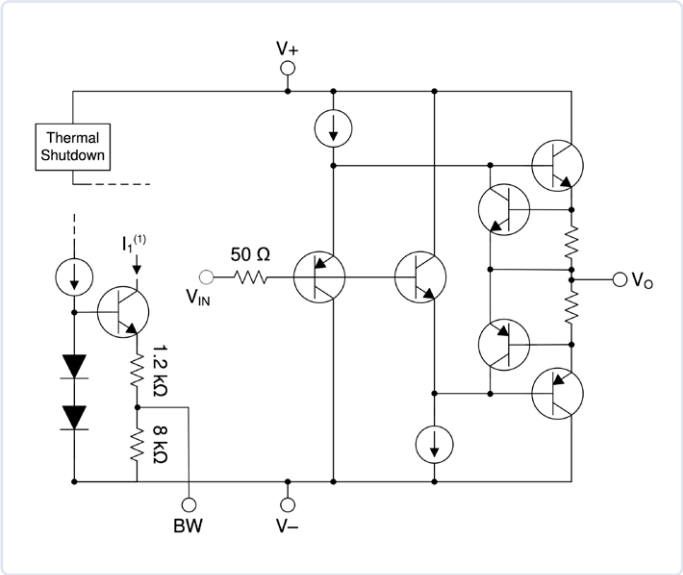


Figure 2. Le circuit de principe de l'étage de sortie push-pull intégré BUF634A. Source : [2].

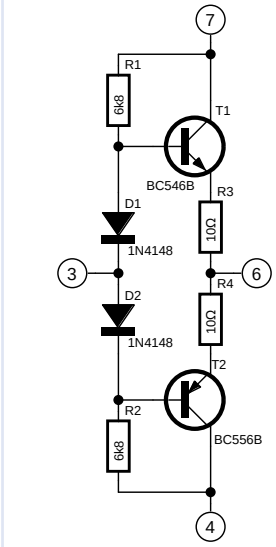


Figure 3. Ce circuit discret peut être utilisé pour remplacer le circuit intégré BUF634A.

omise (broche 1 = en l'air), la bande passante est de 35 MHz, et le courant de repos n'est que de 1,5 mA environ Si une valeur de 0 Ω est choisie pour Rx, alors la bande passante est maximale. Dans ce cas, le courant de repos passe à une valeur tolérable de 8,5 mA. L'étage de sortie peut délivrer des courants allant jusqu'à 250 mA - ce qui est suffisant pour les casques dont l'impédance ne dépasse pas 8 Ω. Pour des informations détaillées, veuillez consulter la fiche technique [2].

Quelle est la bonne bande passante, et donc le bon courant de repos ? Pour un petit amplificateur de casque, normalement la faible bande passante devrait convenir, et vous pouvez vous passer de Rx. Pour des charges capacitives plus élevées par contre, le courant de repos plus élevé avec Rx = 0 Ω est le meilleur choix. Ce dernier point s'applique également aux ampli-op, qui ne sont pas stables à gain unitaire. Les courants de repos entre les extrêmes sont facilement ajustables avec d'autres valeurs de Rx.

Réglages fins et modifications

Avec des composants CMS, vous pouvez construire un superbe amplificateur de casque avec une distorsion minimale, un bruit très faible et une bande passante élevée dans un espace minuscule. Mais ceux qui aiment les traversants seront également satisfaits de ce circuit. IC1 est également disponible dans une version DIP à 8 broches, et pour IC2 et IC3, vous pouvez facilement utiliser un substitut de type BUF634. Ce CI est toujours disponible dans un boîtier DIP à huit broches. Il fonctionnera également, bien qu'il soit un peu plus lent, avec un courant de repos légèrement plus élevé, et ne soit plus trop recommandé pour les nouvelles conceptions.

Une alternative consisterait à remplacer le BUF634A intégré par un circuit équivalent discret de transistors à petits signaux, comme illustré à la **figure 3**. Le courant de repos dépend alors des caractéristiques combinées des semi-conducteurs respectifs, et peut être ajusté via les valeurs de R1 et R2. De plus, pour la stabilité thermique, il faut s'assurer que D1 et D2 soient en contact direct avec T1 et T2. Si nécessaire, ils peuvent également être mis ensemble. Comme le courant de repos dépend également de la tension d'alimentation, l'amplificateur doit être utilisé avec une alimentation stabilisée de ±12 V, par exemple en utilisant des circuits intégrés régulateurs de tension 7812 ou 7912. Pour des casques de 32 Ω, une alimentation de ±100 mA est suffisante. Pour un casque de 8 Ω, il faut compter le double. R7 réduit les ronflettes provenant de potentielles boucles de masse.

LISTE DES COMPOSANTS

Résistances
(par défault: à film métallique, 1%)
R1,R2 = 100 k
R3,R4 = 1 k
R5,R6 = 4k7
R7…R9 = 22 Ω
Rxa,Rxb = voir texte

Condensateurs
C1,C2 = 1 µ, 25 V, film
C3,C4 = 470 µ, 16 V, électrolytique
C5,C6 = 100 n, 25 V
C7…C10 = 1 n, 25 V, céramique

Semi-conducteurs
IC1 = OPA2134
IC2,IC3 = BUF634A (voir texte)

Autre
K1 = fiche casque, stéréo, 6,3 mm

Bien sûr, vous pouvez expérimenter avec d'autres modèles pour IC1 ou T1 et T2 et obtenir également de bons résultats. Lorsque vous utilisez des écouteurs avec des impédances ≥32 Ω, il n'y a pas de problèmes de chaleur avec le BUF634A. À 8 Ω et une écoute prolongée à un volume élevé, les circuits intégrés en boîtier SOIC peuvent assez vite monter en température. Alors, lors de la conception d'un circuit, assurez-vous que la chaleur est dissipée via les pastilles vers une surface de cuivre suffisante, ou choisissez le boîtier DRB à 8 broches qui dispose d'un dissipateur thermique en-dessous. Le type BUF634 (sans le « A ») est même disponible en boîtier TO-220 ou TO-263 qui peut être facilement refroidi.

Il est préférable d'utiliser des condensateurs à film pour C1 et C2 à l'entrée. Il n'est pas nécessaire d'utiliser des condensateurs à la sortie car la tension d'offset typique sera de l'ordre de quelques mV, ce qui ne provoque pas de mouvement significatif des bobines mobiles. Par contre, si vous voulez être rassuré par rapport à cela, vous pouvez mettre un condensateur entre la sortie et le casque. Pour une impédance de 32 Ω , un condensateur de type bipolaire de 470 µF (25 V) serait suffisant. Les casques de 8 Ω ont besoin d'au moins 1000 µF. La sortie est protégée contre les courts-circuits, mais pas si vous utilisez le remplacement du BUF634 de la **figure 3**.

Conclusion

L'électronique analogique moderne permet de réaliser des amplificateurs de casque minuscules avec des caractéristiques audiophiles remarquables. Des facteurs de distorsion inférieures à 0,01 % à des volumes moyens sont facilement réalisables, et le bruit est pratiquement inaudible grâce aux bonnes propriétés de IC1. L'amplificateur peut délivrer jusqu'à 300 mW à un casque de 32 Ω (et encore jusqu'à 250 mW à des charges de 8 Ω avec un bon refroidissement), ce qui est bien plus que ce que vos oreilles ne peuvent supporter.

200441-04 — VF : Laurent Rauber

À propos de l'auteur

Electronicien de formation, Thierry Clinquart a orienté sa passion vers l'audio. Tous ses projets tournent autour de l'audio analogique – préamplification, traitement dynamique, corrections, distribution du signal, etc. Il réalise ses propres circuits imprimés avec Sprint-Layout d'Abacom et sPlan pour les schémas.

Des questions, des commentaires ?

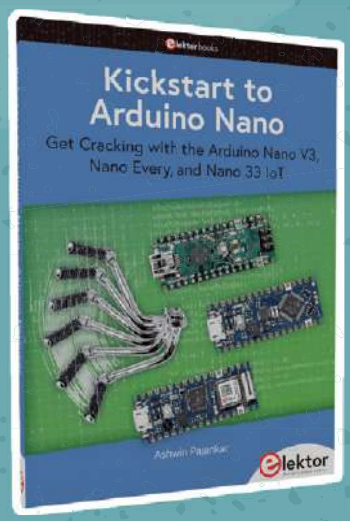
Contactez Elektor (redaction@elektor.fr).

Produits

- **OWON SDS1102 oscilloscope 2 voies (100 MHz) (SKU 18782)**
  www.elektor.fr/18782
- **PeakTech 3350 True RMS multimètre numérique (6000 points) (SKU 19986)**
  www.elektor.fr/19986
- **Joy-IT JDS6600 générateur de signaux et compteur de fréquence (SKU 18714)**
  www.elektor.fr/18714

LIENS

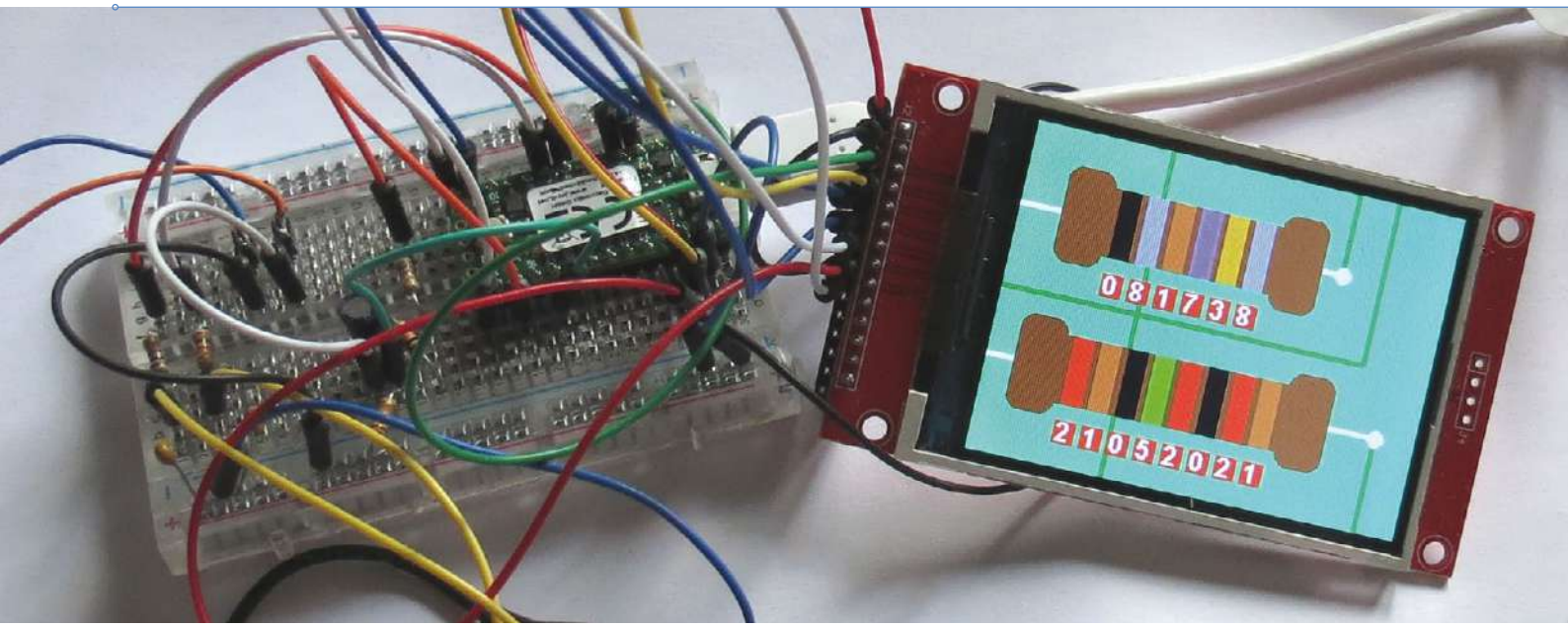
[1] Fiche technique de l'OPA2134 : https://ti.com/product/OPA2134
[2] Fiche technique du BUF634A : https://ti.com/product/BUF634A



Kickstart to Arduino Nano

Ce livre constitue la première étape pour les novices et les amateurs de microcontrôleurs qui souhaitent s'initier à la programmation Arduino. Il suit une approche pas à pas pour expliquer les concepts et le fonctionnement des choses. Chaque concept est suivi d'un schéma de circuit et d'exemples de code. Viennent ensuite des explications détaillées sur la syntaxe et la logique utilisées. En suivant de près ces concepts, vous deviendrez plus à l'aise avec la construction de circuits.

www.elektor.fr/20241



L'horloge radio SDR composée d'un microcontrôleur Teensy 4.0 avec un écran est montée sur une plaque d'expérimentation. Ici, l'heure est affichée en code couleur de résistance.

horloges radiocommandées SDR

Martin Ossmann (Allemagne)

Les microcontrôleurs d'aujourd'hui sont si puissants qu'ils peuvent être utilisés pour construire des horloges radiocommandées basées sur le principe de la radio définie par logiciel (SDR). Le fonctionnement de ce principe a été démontré par l'horloge radio MSF [1] construite autour d'un Raspberry Pi Pico. Cet article aussi porte sur les signaux horaires, mais cette fois, la carte rapide Teensy 4.0 est utilisée comme microcontrôleur.

Tableau 1. Des récepteurs sont implémentés pour ces émetteurs.

60,0 kHz	MSF	Émetteur de signaux horaires britannique NPL
77,5 kHz	DCF77	Signal horaire allemand
129,1 kHz	EFR	Contrôle de l'ondulation de la radio
162 kHz	TDF	Signal horaire français
198 kHz	BBC	Radiodiffusion AM avec modulation de phase

Dans ce projet, nous développons un récepteur pour les signaux du service radio EFR, qui est utilisé par les fournisseurs d'énergie pour le contrôle des ondulations. Il transmet également des informations sur l'heure à intervalles réguliers. Construire une horloge radio avec ce service est quelque chose de nouveau. Le même concept et le même matériel basés sur la RRL peuvent être utilisés pour recevoir d'autres signaux horaires, que nous prévoyons largement d'utiliser (**tableau 1**).

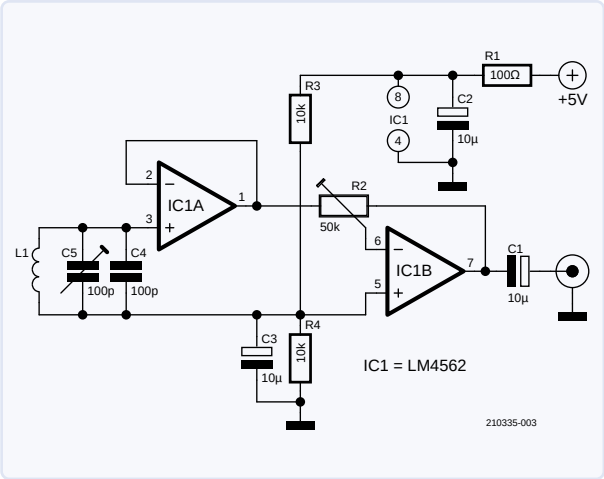


Figure 1. Schéma du circuit de l'antenne active pour 129.1 kHz. Les 250 spires de fil de cuivre émaillé de 0,2 mm² sont enroulées sur une formeuse de bobine ETD29. La bobine est poussée sur une barre de ferrite de 150 mm × 8 mm en matériau 3B1. En ajustant Cx, le circuit est utilisable pour des fréquences de 60 kHz à 200 kHz.

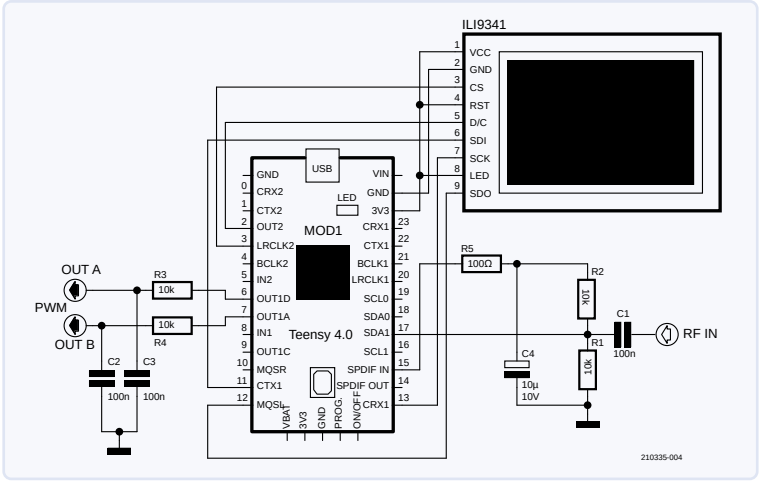


Figure 2. Schéma de circuit de notre radio définie par logiciel pour les horloges radiocommandées.

Un autre objectif de développement important est d'afficher l'heure sous de diverses formes innovantes.

Un concept pour toutes les fréquences

Pour recevoir les signaux radio, la première chose dont nous avons besoin est une antenne active, comme dans le schéma de la **figure 1**. Le signal de la barre de ferrite (150 mm × 8 mm, matériau 3B1, forme de la bobine ETD29, enroulement 250 tours CuL 0,2 mm2) est filtré par un circuit résonnant, puis tamponné par un ampli-op. Le deuxième étage est un autre ampli-op, dont le gain peut être ajusté aux conditions locales à l'aide du potentiomètre R2. Notre récepteur fonctionne mieux lorsqu'un signal d'environ 50 mVSS est présent à la sortie du circuit amplificateur à deux étages. Le circuit est adapté à toutes les fréquences de signal temporel utilisées ici ; si vous voulez l'utiliser pour d'autres fréquences, vous devrez adapter le condensateur C4.

Le schéma complet du système est présenté dans la **figure 2**. Le composant principal est une carte Teensy 4.0. Avec les résistances R1 et R2, nous réglons l'offset du CA/N, et avec C1 nous couplons le signal de l'antenne active au CA/N de la carte Teensy. Nous avons construit notre prototype sur une carte de prototypage, comme le montre la **figure 1**. Les deux filtres passe-bas, R4/C2 et R3/C3, filtrent les signaux PWM des sorties 6 et 7. Nous utilisons les signaux PWM pour le débogage, par exemple, pour afficher les signaux I et Q sur un oscilloscope. L'écran est connecté via le connecteur SPI et quelques broches supplémentaires. L'alimentation est fournie par le port USB, qui est également utilisé pour programmer la carte et sortir les messages de débogage. Cela vous permet de voir la réception des données dans le moniteur série de l'EDI Arduino. La carte Teensy peut être programmée dans l'environnement de développement normal d'Arduino.

Réception des codes horaires

Le concept « universel » fait référence non seulement au matériel du récepteur, mais aussi à son logiciel. La structure reste similaire pour tous les émetteurs de signaux horaires répertoriés ici. Mais, comme les routines de démodulation et de décodage sont très différentes, il m'a semblé plus logique d'écrire un micrologiciel séparé pour chaque émetteur de signaux horaires que de les regrouper dans un seul micrologiciel et d'activer celle que l'on souhaite via un commutateur, par exemple. Le logiciel complet et documenté peut être téléchargé sur la page du projet à [2]. Cependant, le fonctionnement de base du récepteur sera d'abord décrit plus en détail, en utilisant la réception de l'émetteur à ondes longues EFR sur 129.1 kHz comme exemple.

Réception de l'EFR

EFR GmbH exploite trois émetteurs à ondes longues (129,1 kHz, 139 kHz et 135,6 kHz) pour le contrôle des ondulations dans les centrales énergétiques [3]. Les émetteurs situés en Allemagne et en Hongrie ont chacun une portée d'au moins 500 km et couvrent ainsi la majeure partie de l'Europe centrale. Le type de modulation utilisé est la modulation par déplacement de fréquence (FSK) avec une déviation de ±170 Hz et un débit de données de 200 bps. En plus des messages propriétaires cryptés, des informations temporelles sont également envoyées. Le protocole est spécifié dans la norme IEC 60870-5.

Le concept de notre récepteur EFR est illustré à la **figure 3**. Le signal de l'antenne active atteint le CA/N interne de la carte Teensy 4.0, qui est cadencé à 333,333 kHz. Ainsi, le théorème d'échantillonnage est rempli. Comme d'habitude avec les récepteurs SDR, la multiplication IQ est ensuite effectuée. Le signal de l'oscillateur est généré par un générateur DDS implémenté dans le logiciel. Après la multiplication du signal d'entrée avec le signal de l'oscillateur I ou Q, un filtrage passe-bas est effectué. Les filtres passe-bas sont dimensionnés de manière à ce que le signal reçu ne soit pas encore écrêté,

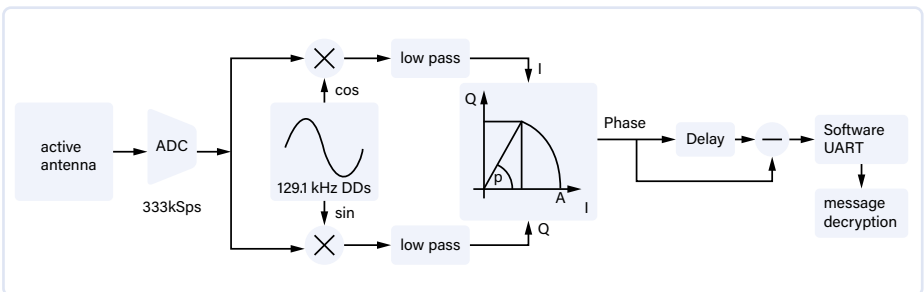


Figure 3. Le concept du récepteur EFR.

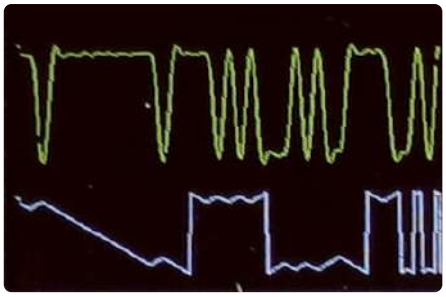


Figure 4. Informations de débogage sur l'écran. Graphique supérieur : signal RTTY démodulé (fréquence instantanée) ; graphique inférieur : phase instantanée.

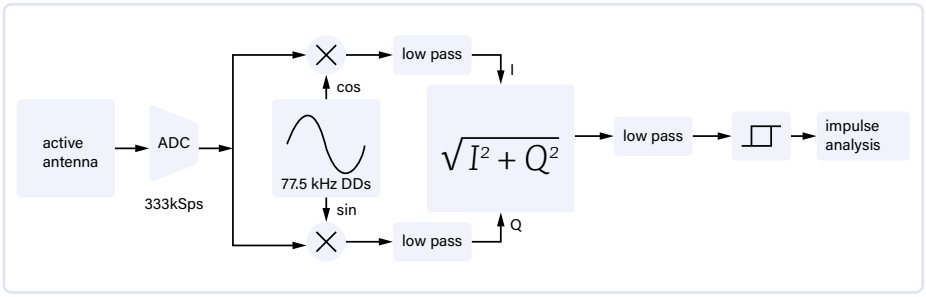


Figure 5. Concept du récepteur DCF.

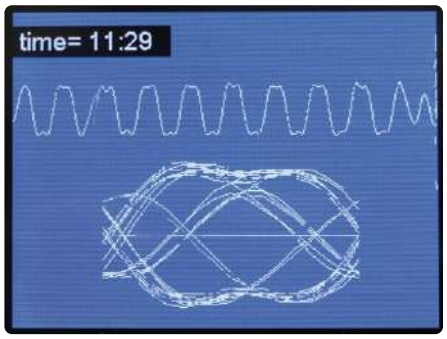


Figure 6. Informations de débogage de la réception de la BBC. En haut : signal de données récupéré ; en bas : diagramme oculaire.

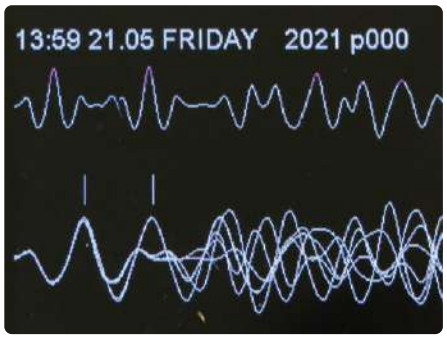


Figure 7. Signaux de débogage de la réception TDF.

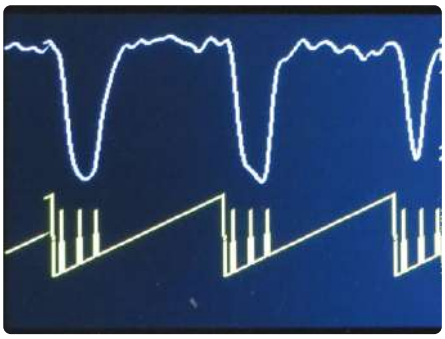


Figure 8. Info de débogage MSF. En haut : niveau de réception avec les seconds marqueurs ; en bas : variable de temporisation avec points d'échantillonnage.

mais qu'aucune autre composante du signal ne puisse non plus le traverser.

Pour démoduler le signal RTTY, nous devons déterminer la fréquence actuellement reçue. Au décalage de +170 Hz, c'est un « 1 », à -170 Hz, c'est un zéro. Pour cela, nous déterminons la phase du courant à partir des signaux I/Q. Nous le faisons avec la fonction `atan2()`. A partir du changement de phase dans le temps, nous pouvons déterminer la fréquence. Pour ce faire, nous calculons la différence entre le signal original et le signal retardé, ce qui donne le signal démodulé, que nous transmettons à un UART logiciel pour récupérer les octets transmis. Ainsi, nous récupérons les messages transmis, desquels nous pouvons extraire l'information temporelle. La **figure 4** montre les signaux de débogage sur l'écran, en haut le signal RTTY démodulé (fréquence instantanée), et en bas la phase instantanée.

Réception du DCF

Le concept du récepteur DCF (**figure 5**) est très similaire à celui de l'EFR. Cependant, cette fois, l'amplitude d'émission instantanée est déterminée à partir des signaux I/Q. Pour ce faire, on calcule $\sqrt{I^2+Q^2}$.

Ce signal est filtré par un filtre passe-bas et acheminé vers un trigger de Schmitt. Les secondes impulsions sont ainsi disponibles pour l'évaluation, et nous pouvons déterminer la date et l'heure à partir des bits.

Réception de la BBC

La BBC diffuse un signal radio sur 198 kHz. La porteuse est modulée en phase et utilisée pour la transmission de données, comme le système RDS de la radio FM [4]. La modulation de phase est démodulée de la même manière que dans le système EFR. L'horloge binaire de 25 bps doit être récupérée à partir du signal démodulé (**figure 6** ci-dessus). Pour ce faire, le récepteur forme en quelque sorte le diagramme en œil et fait varier l'horloge de manière à ce que l'œil s'ouvre au maximum. Les données sont alors échantillonnées au centre de l'œil. À partir du flux binaire ainsi obtenu, il faut maintenant récupérer les limites des blocs. Les données sont envoyées par blocs de 50 bits chacun, dont 13 bits sont des bits de contrôle. Le récepteur recherche dans le flux de bits les blocs sans erreur et reconnaît leurs limites par leur position. Il existe différents types de blocs qui contiennent des données différentes. Le type 0 est destiné à la transmission de l'information temporelle et est évalué par notre récepteur.

Réception TDF

Jusqu'en 2016, TDF (anciennement *Télédiffusion de France*) diffusait un programme radio sur 162 kHz, mais, aujourd'hui, seule la porteuse est transmise en modulation de phase. L'information temporelle est incluse dans les données. Un bit zéro est envoyé sous forme d'une impulsion unique, un bit « 1 » sous forme d'une double impulsion, chacune commençant au début de la seconde. La **figure 7** montre les informations de phase reçues (courbe supérieure) et en bas, comme une sorte de diagramme de l'œil, des courbes superposées montrant les impulsions à gauche entre les lignes verticales. À la 59e seconde de chaque minute, le signal n'est pas modulé en phase. Notre récepteur utilise cette pause pour la synchronisation. L'information binaire individuelle est structurée de manière assez similaire à celle du DCF77.

MSF sur 60 kHz

Sur 60 kHz, le NPL (National Physical Laboratory) en Grande-Bretagne exploite un émetteur à ondes longues qui diffuse des informations temporelles à 15 kW. Une simple modulation on-off de la porteuse est utilisée [5]. La minute commence par une pause de 500 ms. Les secondes suivantes commencent chacune par un intervalle de porteuse de 100 ms, puis deux autres intervalles de 100 ms si nécessaire, en fonction des bits à transmettre. Notre récepteur MSF est construit de manière assez similaire au récepteur DCF, seul le décodage des impulsions est effectué différemment. Comparé aux

autres signaux, le signal MSF est relativement faible, mais peut encore être assez bien reçu à la frontière ouest de l'Allemagne (**figure 8**).

Options d'affichage

Lorsque l'auteur a décidé de construire une horloge radiocommandée basée sur l'EFR, il a également dû choisir l'écran à utiliser. C'est un LCD de 2,8 pouces avec 320 × 240 pixels qui peut être utilisé pour réaliser une large gamme d'options graphiques. L'écran est basé sur un contrôleur ILI9341 et peut être obtenu, par exemple, sur eBay. Il est également possible d'utiliser le petit frère de 2,2 pouces avec la même résolution provenant de l'e-choppe d'Elektor (voir l'encadré « Produits »). L'écran est connecté à la carte Teensy via l'interface SPI.

Affichage de la résistance

Notre horloge EFR offre une nouvelle option d'affichage quelque peu inhabituelle, l'« affichage à résistances » (**figure 9**). Les chiffres de l'heure et de la date sont affichés selon un code couleur comme sur des résistances électriques. Il y en a deux dans l'affichage, une avec six anneaux pour les six chiffres de l'heure (heures:minutes:secondes, par exemple, 12:23:45). La seconde résistance a huit anneaux pour la date (jour.mois.année, par exemple, 23.08.2012). L'heure est mise à jour toutes les secondes, et l'horloge locale est toujours synchronisée lorsque des données horaires sont reçues du service EFR. Pour les utilisateurs qui ne connaissent pas encore le code de résistance, l'heure et la date sont également affichées à l'aide de chiffres standard. Les lecteurs puristes pourraient faire remarquer ici que les résistances ont un maximum de six anneaux, mais cela permet au moins aux débutants d'apprendre le code couleur DIN41429 de manière ludique.

Présentoir à dominos

Comme autre idée d'affichage amusante, un affichage de dominos a été mis en œuvre (**figure 10**). Sur les dominos, vous pouvez afficher les chiffres de 0 à 9. Chaque domino offre un espace pour deux chiffres, il en faut donc trois pour représenter l'heure et quatre pour représenter la date. L'utilisation de l'affichage à résistance, de l'affichage à dominos ou d'une autre option est déterminée dans le code source à l'aide des bits d'option (**listage 1**). L'option à choisir est sélectionnée avec un « 1 », et les autres options sont désactivées avec un « 0 ».

Horloge à mots et style analogique

En plus des affichages décrits jusqu'à présent, vous pouvez également en sélectionner un à mots, où l'heure est affichée en texte (allemand) à l'écran (**figure 11**). Avec cette option, seule l'heure est affichée, pas la date. Elle est mise à jour toutes les cinq minutes. Le format d'affichage classique d'une horloge analogique est également mis en œuvre (**figure 12**). L'heure est affichée sur un cadran rond et la date sous forme de texte en dessous.

Listage 1. L'affichage est sélectionné via ces constantes.

```
#define scopeShow     0
#define scopeClkShow  1
#define resiShow      0
#define dominoShow    0
#define wordShow      0
#define sevenSegShow  0
#define AP6571Show    0
```

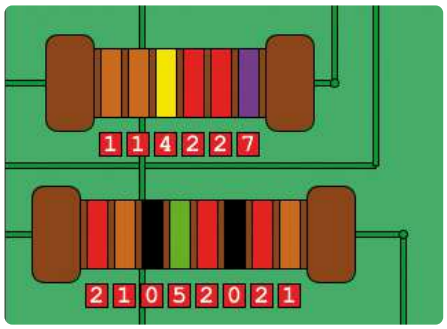


Figure 9. Option d'affichage en code couleur de résistance.

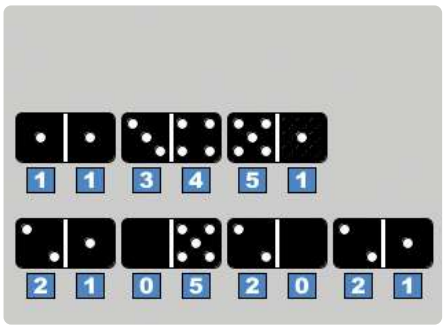


Figure 10. Affichage en domino.



Figure 11. Une horloge à mots est également implémentée.

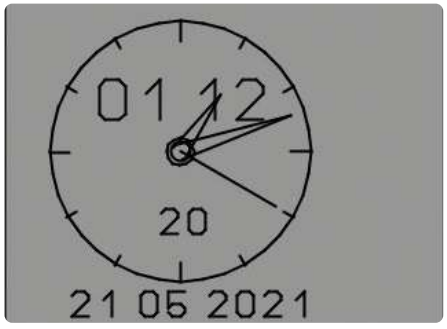


Figure 12. Horloge analogique classique.



Figure 13. Affichage nostalgique à sept segments.



Figure 14. Le temps représenté comme s'il provenait d'une imprimante matricielle.

Affichage à sept segments

Bien entendu, nous proposons également un affichage à sept segments, simple et nostalgique. Donc, si vous voulez être rappelé des vieux jours, c'est le bon choix (**figure 13**).

Affichage du générateur de caractères

Comme dernière option, nous proposons un affichage de texte simple basé sur le légendaire générateur de caractères 6571AP ROM, autrefois largement utilisé dans les imprimantes matricielles à 9 broches (**figure 14**).

Dans l'ensemble, ce projet offre de nombreuses combinaisons et options. Vous pouvez choisir entre cinq variantes d'horloge radio, de l'EFR au DCF, et entre six options d'affichage, de sorte que chacun devrait pouvoir trouver son bonheur. Si vous le souhaitez, vous pouvez également utiliser ce logiciel, par exemple les options d'affichage, comme point de départ pour vos propres développements.

210335-04 — VF : Maxime Valens

Des questions, des commentaires ?

Envoyez un courriel à l'auteur (ossmann@fh-aachen.de) ou contactez Elektor (redaction@elektor.fr) !

Produits

- **Teensy 4.1 carte de développement (SKU 19311)**

  www.elektor.fr/19311

- **2.2» SPI TFT module afficheur ILI9341 (240x320) (SKU 18419)**

  www.elektor.fr/18419

LIENS

[1] Martin Ossmann, « Radio logicielle MSF utilisant un Raspberry Pi Pico », Elektor 7-8/2022 : https://www.elektormagazine.fr/magazine/elektor-267/60603
[2] Téléchargements pour cet article : https://elektormagazine.fr/210335-04
[3] Le signal EFR : https://mee.hu/files/images/5/B_Sbick_EFR-CEE_Lakihegy.pdf
[4] Spécifications L.F. Radio Data : https://downloads.bbc.co.uk/rd/pubs/reports/1984-19.pdf
[5] Codage horaire MSF 60 kHz : https://www.npl.co.uk/products-services/time-frequency/msf-radio-time-signal/msf_time_date_code

elektor e-zine
Your dose of electronics



Chaque semaine passée sans vous abonner à l'e-zine d'Elektor, vous manquez des articles et des projets électroniques intéressants.

Alors, pourquoi attendre plus longtemps ? Abonnez-vous dès aujourd'hui à www.elektor.fr/ezine et recevez un livre numérique gratuit de projets Raspberry Pi.

elektor

démarrer en électronique

diodes spéciales



Eric Bogers (Elektor)

Dans cet épisode, nous terminons le chapitre sur les diodes par quelques exemples de diodes plutôt rares, qui existent encore aujourd'hui, notamment les thyristors, les triacs, les LED et les optocoupleurs.

La première génération de circuits gradateurs était construite avec des thyristors – les triacs n'existaient pas encore (du moins pas pour les niveaux de puissance requis). Aujourd'hui, les gradateurs utilisent presque exclusivement des triacs.

Le thyristor

Un thyristor (**figure 1**) n'est en fait qu'une diode avec une borne de commande – le thyristor ne conduit que dans le sens direct, tout comme une diode ordinaire. Par conséquent, nous avons besoin de deux thyristors (ou d'un redresseur) pour les applications en courant alternatif.

Un thyristor dispose d'une troisième borne appelée gâchette (les deux autres bornes sont appelées anode et cathode, comme pour une diode conventionnelle). Pour passer en mode conducteur, il nécessite une « impulsion de déclenchement ». Une fois qu'un thyristor est amorcé, il continue à conduire tant qu'un courant le traverse – il n'est pas nécessaire de maintenir une tension sur la gâchette. Pour bloquer le thyristor (le commuter), il faut interrompre le courant qui le traverse. Dans un circuit à courant continu, un thyristor reste conducteur indéfiniment ; en courant alternatif, il ne reste conducteur que jusqu'au prochain passage à zéro.

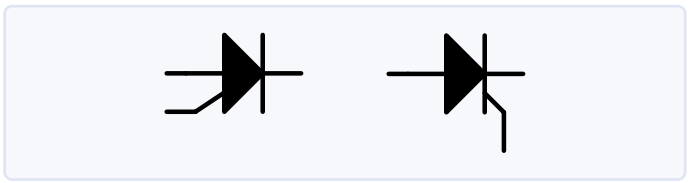


Figure 1. Le thyristor.

Il y a deux façons de construire un gradateur avec un thyristor : la tension alternative peut être redressée, ou bien, on peut utiliser deux thyristors antiparallèles. Comme les circuits de gradateurs construits de cette manière n'ont plus aucune utilité pratique, nous n'entrerons pas dans les détails.

On utilise souvent les thyristors comme protection contre les surtensions, comme le montre la **figure 2**. Si le stabilisateur ou un transistor de puissance claque dans l'alimentation du réseau, alors, sans protection, une tension (beaucoup) plus élevée sera appliquée aux composants et appareils connectés, ce qui pourrait les détruire. C'est une situation à éviter à tout prix.

Cependant, dès que la tension dans le circuit de la **figure 2** devient trop élevée, la diode Zener devient conductrice et allume le thyristor, et, par conséquent, le thyristor continue à conduire. Cela provoque un court-circuit, entraînant le fusible à sauter. Mais c'est moins pire qu'avoir un circuit complètement détruit.

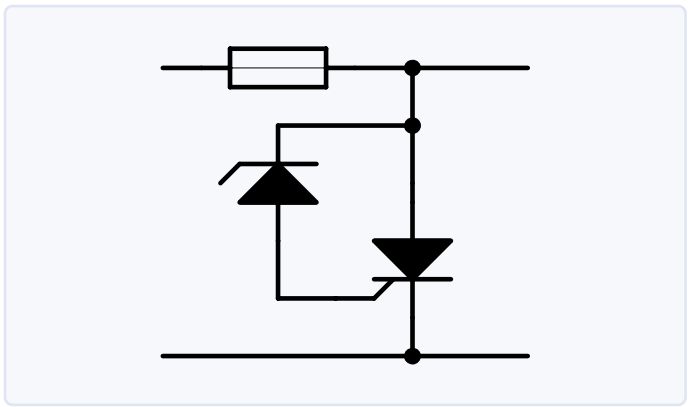


Figure 2. Protection contre les surtensions (principe).

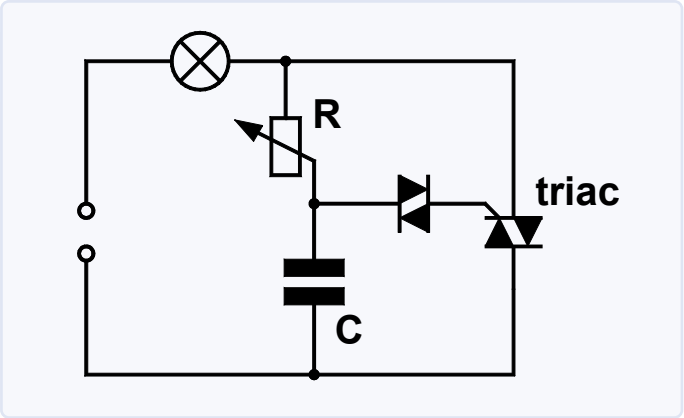


Figure 3. Circuit de gradateur conventionnel (lire : obsolète).

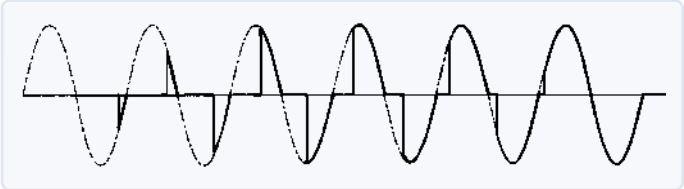


Figure 4. Coupure de phase.

Selon son type, un thyristor exige une tension d'amorçage comprise entre 0,8… 2,5 V. La tension Zener doit donc être soigneusement sélectionnée, de sorte qu'en cas de surtension, ce niveau de tension sera certainement atteint.

Triac

Tout comme un diac, qui est essentiellement une combinaison de deux diodes Zener, un triac est une combinaison de deux thyristors – il est uniquement utile dans les applications en courant alternatif utilisant un seul composant. Examinons le triac dans un circuit pratique – un circuit de gradateur classique, comme le montre la **figure 3**. Ce type de gradateur fonctionne sur le principe de la coupure de phase : la luminosité de la lumière connectée dépend de la période. Référez-vous à la **figure 4** pour plus de détails.

Le triac est enclenché lorsque la tension aux bornes du diac dépasse approximativement 33 V fournissant ainsi l'impulsion d'amorçage. Lorsque le potentiomètre R de la **figure 3** est en butée, la tension est divisée par le réseau RC d'une part et déphasée de 90° d'autre part – la tension aux bornes du diac n'atteindra jamais une valeur de 33 V. Par la suite, si l'on ajuste progressivement le potentiomètre, la tension aux bornes du diac augmente (et sa phase change progressivement), ce qui a pour effet d'accélérer l'enclenchement du triac.

Le circuit de la **figure 3** est l'un des gradateurs (de tension de réseau) les plus simples que l'on puisse imaginer : il ne dispose d'aucune forme d'antiparasitage, et aucune mesure n'a été prise non plus contre l'hystérésis de commutation. De nombreux circuits de gradateurs améliorés sont disponibles de nos jours, mais ils sont beaucoup plus compliqués.

Bien entendu, il est possible de varier l'intensité d'une ampoule en connectant en série un potentiomètre réglable (de puissance), mais une grande partie de l'énergie du secteur dans ce potentiomètre serait dissipée en chaleur. L'avantage du principe de coupure de phase est que cette perte de puissance ne se produira pas.

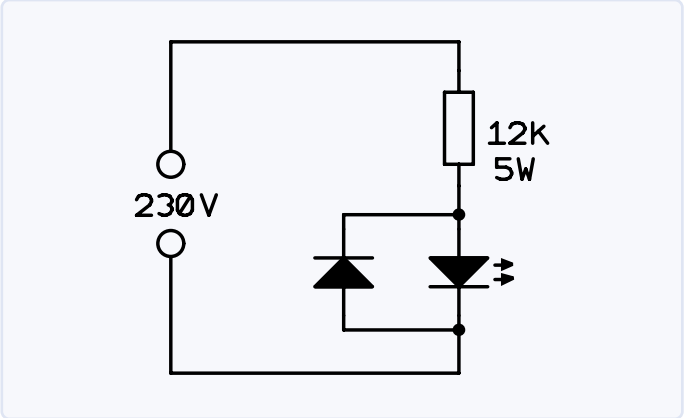


Figure 5. LED avec diode flyback.

Diodes électroluminescentes et optocoupleurs

Les diodes électroluminescentes sont des diodes qui émettent de la lumière dès qu'elles sont traversées par un courant. La tension directe est d'environ 2 V, tandis que le courant maximal est normalement d'environ 20 mA (pour les diodes dites à faible courant, ce courant maximal est de l'ordre de 2 mA seulement).

Depuis des années, les LED remplacent les ampoules ordinaires dans les panneaux de commande. Elles se caractérisent par une très longue durée de vie, une consommation d'énergie relativement faible et une grande résistance aux chocs et aux vibrations.

Aujourd'hui, les LED sont disponibles dans toutes les couleurs possibles et il existe également des LED laser (auxquelles nous devons le lecteur CD et DVD) tandis que les LED Neopixel dotés d'un microcontrôleur intégré permettent d'émettre toutes les couleurs, sans parler de divers effets lumineux. Les LED ont également fait leur entrée dans la technologie de l'éclairage pour remplacer (obligatoirement) les ampoules à incandescence.

Bien que tout cela soit très intéressant, une analyse détaillée de ces sujets dépasse largement le cadre de ce cours de base. Nous nous limiterons donc à une brève discussion générale, en hommage à ce petit mais très important composant.

LED alimentées par tension alternative

Dans de nombreux cas, la tension inverse des LED n'est que de quelques volts. Dans le cas où elles sont alimentées par une tension alternative, il faut connecter une diode flyback pour limiter la tension inverse aux bornes de la LED à une valeur sûre (voir **figure 5**). À propos, vous pouvez constater que le symbole schématique d'une LED est une diode conventionnelle, mais avec une double flèche symbolisant son émission de lumière.

Optocoupleurs

Un optocoupleur se compose d'une diode électroluminescente et d'un composant photosensible (souvent un phototransistor) intégrés dans un seul boîtier (**figure 6**). Dès qu'une tension est appliquée aux bornes de la diode électroluminescente, celle-ci s'allume, entraînant la conduction du phototransistor.

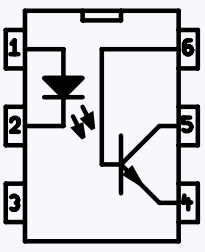


Figure 6. Optocoupleur.

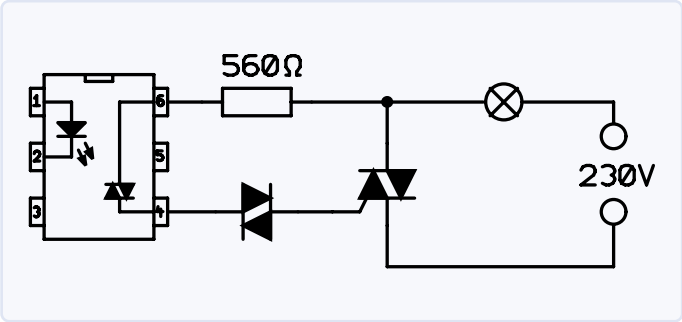


Figure 7. Optocoupleur avec phototriac.

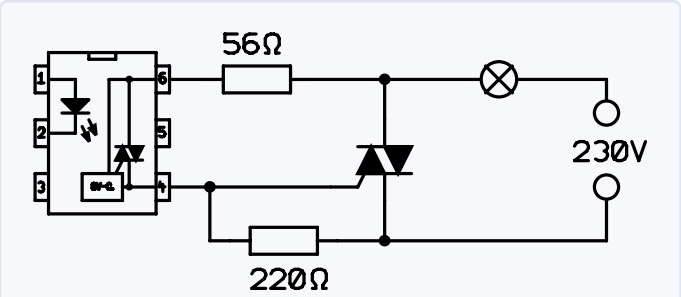


Figure 8. Optocoupleur avec détecteur de passage à zéro.

L'avantage majeur de cette approche apparemment compliquée est la séparation galvanique entre la LED et le phototransistor : une différence de tension de (souvent) quelques centaines de volts entre ces composants est possible sans entraîner de problèmes. Les optocoupleurs sont largement utilisés pour maintenir la partie puissance des gradateurs ou d'autres circuits séparés des circuits de commande. Par exemple, il est possible d'alimenter la section haute puissance à partir du réseau tandis que le circuit de commande n'en est pas affecté (pour des raisons de sécurité).

Optocoupleur avec Phototriac

Un transistor ne conduit que dans un seul sens (nous verrons ça dans le prochain épisode), et cela peut être peu pratique pour certaines applications. Si nous souhaitions commander un triac, nous aurions besoin d'une tension dédiée et le triac serait enclenché par une impulsion de polarité inverse toutes les deux moitiés d'une période de tension alternative, ce qui donnerait un comportement asymétrique. Pour faire bref, un optocoupleur conventionnel n'est absolument pas idéal pour ce type d'applications. Par conséquent, dans de tels cas, il est préférable d'utiliser un optocoupleur avec un phototriac plutôt qu'un phototransistor (**figure 7**).

Lorsqu'un courant est fourni à la LED et qu'elle s'allume, le phototriac devient conducteur et commute à son tour le triac externe. Pour les charges faibles (par exemple un petit ventilateur), vous pouvez vous passer du triac externe, si nécessaire, et alimenter la charge directement par le phototriac.

Le circuit de la **figure 7** est censé pouvoir fonctionner même sans résistance ni diac. Cependant, deux problèmes peuvent survenir :

- Sans le diac, le circuit a tendance à se déclencher de manière incontrôlée : le triac commence à conduire sans une impulsion de commande de la LED.
- De nombreux triacs de puissance causent le grillage du phototriac si le courant n'est pas limité par une résistance.

Optocoupleurs avec détection du passage à zéro

Lorsque le courant est établi au milieu d'une periode, il génère une impulsion d'interférence haute fréquence qui peut, par exemple, provoquer des interférences audibles sur les systèmes audio. Ces interférences HF sont à l'origine du fameux « bourdonnement du gradateur ». Malheureusement, les gradateurs fonctionnant selon le principe de la coupure de phase présentent un tel inconvénient : nous pouvons seulement essayer de maîtriser le problème en utilisant des filtres antiparasites appropriés.

Mais, dans toutes les situations où vous devez juste allumer ou éteindre une charge, le moment de la commutation doit correspondre au passage par zéro de la tension alternative du réseau : dans ce cas, il n'y aura pas d'interférences. Vous pouvez concevoir et construire vos propres circuits complexes à cet effet, ou vous pouvez utiliser un optocoupleur qui intègre déjà la détection du passage par zéro.

Bien entendu, il n'est pas possible de l'utiliser pour commuter un triac exactement au point de passage par zéro, car, à ce stade, il n'y a aucune tension disponible pour fournir le courant d'allumage requis. Cependant, il est possible de commuter aussi près que possible du point de passage à zéro, ce qui entraîne un niveau d'interférence négligeable.

Nous concluons ici notre exploration du monde merveilleux des diodes. Dans le prochain épisode, nous aborderons le transistor.

(220446-04) — VF : Asma Adhimi

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

Des questions, des commentaires ?

Contactez Elektor (redaction@elektor.fr).

Produits

- **Livre en anglais « Basic Electronics for Beginners » B. Kainka (Elektor, 2020) (SKU 19212)**
  www.elektor.fr/19212
- **Livre en anglais « Basic Electronics for Beginners » B. Kainka (version numérique, SKU 19213)**
  www.elektor.fr/19213

elektor TV

Elektor TV Shows



Elektor **Engineering Insights**

Elektor Industry Insights est une source d'information incontournable pour les ingénieurs et les électroniciens qui souhaitent rester informés sur le monde de l'électronique. Au cours de chaque épisode, Stuart Cording (rédacteur, Elektor) discutera des défis et des solutions d'ingénierie avec des experts de l'industrie électronique.
www.elektormagazine.com/elektor-engineering-insights

Elektor LabTalk

Vous êtes passionné par l'électronique, la programmation ou la théorie de l'ingénierie ? Rejoignez les ingénieurs et rédacteurs de l'équipe Elektor Lab qui partagent leurs astuces, planifient leurs projets électroniques, discutent du magazine Elektor et répondent aux questions de la communauté.
www.elektormagazine.com/elektor-lab-talk





elektor academy

Vous voulez améliorer vos compétences en électronique ? Découvrez dans Elektor Academy les ressources qui vous permettront de développer vos compétences techniques. Notre expert Stuart Cording vous guidera à travers les cours d'Elektor Academy.
www.elektormagazine.com/elektor-academy

Restez informé et rejoignez notre chaîne YouTube Elektor TV.
www.youtube.com/c/ElektorIM

elektor



Photo: Shutterstock/ Jackrit Singhanutra

sur le vif

errances qualitatives

Ilse Joostens (Belgique)

La rengaine « c'était mieux avant » est souvent reprise par des personnes d'un certain âge. Elles évoquent le « bon vieux temps », et parfois s'en prennent aussi aux « jeunes d'aujourd'hui ». Du haut de ma cinquantaine passée, je peux pourtant affirmer que tout n'était pas « mieux avant ». Si cette idée est ancrée en nous depuis toujours, c'est pour l'essentiel que nous retenons mieux les bonnes expériences que les mauvaises.

Malgré cela, je ne pense pas être victime d'illusion mentale lorsqu'un produit me parait avoir été de meilleure qualité autrefois. Certains n'étaient bien sûr que pure camelote, mais la situation actuelle n'est guère plus reluisante. Le progrès technologique n'est pas toujours synonyme d'améliorations. Il arrive aussi, hélas, qu'il restreigne la liberté des consommateurs. La réduction de la durée de vie de certains produits – aussi discrète que délibérée – assombrit un peu plus le tableau.

Ampoules royales et batteries non-remplaçables

Datée de 1993, l'invention des LED bleues a révolutionné le domaine de l'éclairage – même si la mise sur le marché des ampoules à LED prit un certain temps. Ces ampoules affichaient une très faible consommation doublée, nous promettait-on, d'une durée de vie de plusieurs dizaines de milliers d'heures. En pratique pourtant, ces heures donnaient parfois l'impression de n'avoir été que des minutes – je ne compte plus le nombre d'ampoules à LED défectueuses qu'il m'a fallu remplacer. Une entreprise cherche avant tout à vendre ses produits, et un produit à longue durée de vie n'est pas bon pour les affaires. Exploiter des composants à la limite – voire au-delà – de leurs caractéristiques nominales est devenu une pratique courante. Ce n'est toutefois pas aussi simple qu'on pourrait le croire : le produit doit tomber en panne aussi vite que possible après la fin de la garantie, mais pas avant au risque d'avoir à le rembourser. Le consommateur, malheureusement, cède toujours aux mêmes fausses promesses et achète de nouveaux modèles en espérant qu'ils soient meilleurs. Ainsi en est-il des ampoules à LED. Beaucoup sont conçues pour que leurs LED absorbent le courant maximal permis, voire un peu plus. Or la relation entre l'intensité lumineuse et le courant n'est pas linéaire. Autrement dit une

plus grande quantité d'énergie est convertie en chaleur et l'efficacité lumineuse chute. Cette configuration malmène les composants et conduit à une panne prématurée de l'ampoule. Le consommateur court en acheter une nouvelle, le fabricant peut en fabriquer une de plus.

Les choses n'ont pas à être ainsi. Pour rendre la ville de Dubaï moins énergivore, l'émir Mohammed ben Rachid Al Maktoum a conclu avec Philips un accord sur la production de millions d'ampoules à LED [1][2][3] opérant à la moitié de leur puissance nominale. Leur durée de vie en est considérablement allongée, et leur consommation électrique réduite de moitié par rapport à des LED classiques de même luminosité. Inutile de dire que ces ampoules sont longtemps restées l'apanage de Dubaï. Ce n'est que récemment que Philips a lancé chez nous une gamme d'ampoules comparables – appelée Ultra Efficace.

On m'a un jour demandé de réparer un récepteur satellite souffrant de démence précoce – il affichait des caractères aléatoires. Tous les condensateurs électrolytiques du rail d'alimentation de 12 V et de la carte principale étaient des 10 V. J'ignore la durée de vie d'un électrolytique de 10 V opérant sous 12 V, mais ce choix devait être intentionnel car la garantie venait d'expirer. Je les ai remplacés par des 16 V et tout est rentré dans l'ordre. Il ne s'agit pas d'un cas isolé. En inspectant le chargeur de batterie défectueux d'une tondeuse à gazon de marque réputée, j'ai découvert que son circuit de filtrage contenait un électrolytique de 10 V devant supporter des pics de tension de 14 V.


Photo : Shutterstock / Jirakorn

Tout le monde sait que les batteries lithium-ion ne sont pas éternelles et devraient, par conséquent, être remplaçables. Celle de mon aspirateur sans fil ne l'est pas. Peu de temps après l'expiration de sa garantie (encore !), l'appareil s'est mis à se décharger de plus en plus rapidement. J'ai pu remplacer ses accus 18650 – au prix de quelques jurons et d'une coupure au doigt – mais ce travail minutieux exige des compétences en soudage hors de portée de l'utilisateur moyen. Notre vieil et énergivore aspirateur-balai, par contre, fonctionne toujours après 24 ans de service. Il en va de même avec les smartphones récents, le remplacement de leur batterie s'apparente de plus en plus à une opération à cœur ouvert.

Gremlins logiciels et nouveau Bauhaus européen

Les pratiques précédentes sont celles de la « vieille école ». La numérisation offre désormais d'innombrables façons d'abuser le consommateur. Cela va des cartouches d'imprimante à puce intégrée aux mises à jour logicielles rendant un appareil (un smartphone p. ex.) plus lent, voire inutilisable car prétendument obsolète. Certains ont même vu un beau matin leur imprimante leur annoncer joyeusement qu'elle était arrivée en fin de vie, et de fait cesser toute activité [4]. Dans la même veine, un célèbre fabricant d'imprimantes d'étiquettes [5] a équipé ses étiquettes de RFID, forçant ainsi l'utilisateur à acheter son coûteux papier, le tout sous prétexte d'offrir une meilleure expérience utilisateur. Ce genre de pratique est bien sûr présenté comme une amélioration au service du consommateur, mais en fin de compte celui-ci en est simplement la victime.

En matière d'obsolescence programmée, les caciques du nouveau Bauhaus européen [6] sont peut-être les pires de tous. Je ne parle pas ici du courant artistique allemand, mais de cette initiative de l'UE nous incitant à jeter des appareils en parfait état de marche et à les remplacer par des modèles « verts » moins énergivores. Outre l'idiotie financière, je pense que la production et le recyclage d'appareils ménagers entraînent des émissions de CO_2 considérables. Et voilà qu'une loi m'obligera à remplacer mon chauffe-eau au gaz – qui marche parfaitement – par un nouvel appareil n'affichant que quelques pour cent d'efficacité en plus. Au moins je sais pourquoi je dois encore travailler…

(220461-04) — VF : Hervé Moreau



LIENS

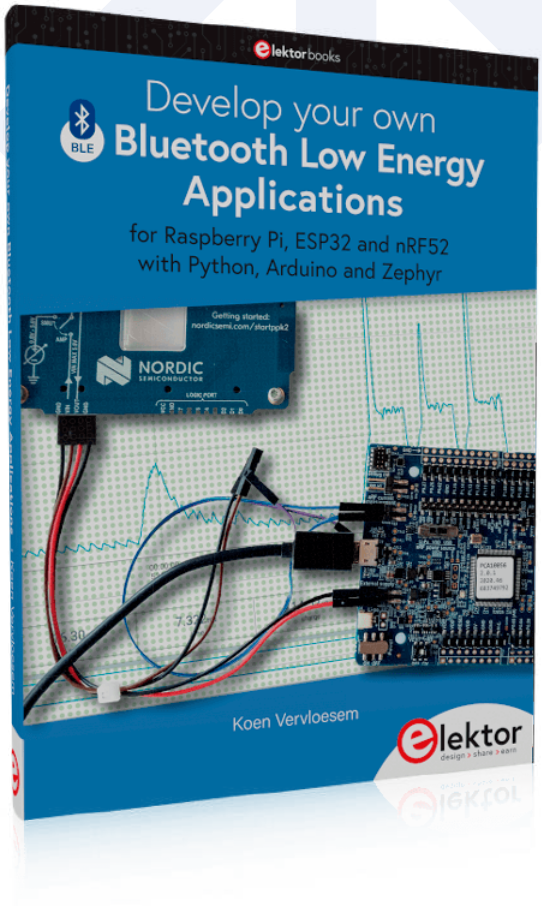
[1] Les ampoules royales de Dubaï que vous ne pouvez pas acheter : https://hackaday.com/2021/01/17/leds-from-dubai-the-royal-lights-you-cant-buy/
[2] Les ampoules interdites. Étude des ampoules de Dubaï : https://youtu.be/klaJqofCsu4
[3] Comment fabriquer des « ampoules de Dubaï » à partir d'ampoules Philips de 13 W : https://youtu.be/rXJq2vLJhLA
[4] Epson bafoue le droit à la réparation : https://elektor.link/epsonobsolescence
[5] EEVblog 1462 - Imprimante d'étiquettes Dymo, un matos craignos ! : https://youtu.be/xzSDJRC0F6c
[6] Rapport sur le nouveau Bauhaus européen : https://www.europarl.europa.eu/doceo/document/A-9-2022-0213_EN.html

rétro-ingénierie d'un badge LED **Bluetooth Low Energy**

comment commander un appareil BLE avec un script Python

Koen Vervloesem (Belgique)

De nombreux appareils BLE (Bluetooth Low Energy) sont livrés avec leurs propres applications mobiles qui implémentent généralement un protocole personnalisé destiné à l'appareil et l'application correspondante, permettant à l'utilisateur de les commander. Souvent, il n'existe pas de spécification que vous pouvez lire pour créer votre propre application. La bonne nouvelle est que la rétro conception d'un appareil BLE est possible, permettant de le désassembler et de l'utiliser avec votre propre logiciel. Lisez la suite pour en savoir plus.



Note de la rédaction. *Cet article est un extrait du livre de 257 pages « Develop your own Bluetooth Low Energy Applications » (Elektor, 2022). L'extrait a été légèrement modifié par l'auteur pour correspondre aux normes éditoriales du magazine Elektor et pour constituer un projet complet et reproductible.*

Dans cet article, je vais rétro concevoir un badge LED BLE comme exemple de cette tâche complexe et très instructive d'ingénierie inverse. Rejoignez-moi ; nous découvrons le fonctionnement de l'appareil et examinons ses services et ses caractéristiques BLE. Je vais décompiler l'application mobile associée et analyser le trafic BLE entre l'application et l'appareil. Mon objectif est de créer un script Python pour contrôler le badge LED afin de se passer de l'application mobile officielle.

Étude du badge LED

Le badge LED d'AliExpress illustré à la **figure 1** est doté d'un écran LED de 11 x 55 pixels disponible en plusieurs couleurs. Il est compatible avec Bluetooth, mais sa version n'est pas précisée.

En scannant le code QR à l'arrière du badge, j'ai obtenu une erreur « HTTP 404 ». Sur Google Play, j'ai trouvé une application appelée

« *Bluetooth LED Name Badge* », développée par Shenzhen Lesun Electronics Co., Ltd (**figure 2**). Cette application permet d'envoyer du texte et des icônes vers le badge et dispose de quelques paramètres, y compris la vitesse de défilement.

Allumons le badge LED et utilisons l'application d'analyse BLE *« nRF Connect for Mobile »* [1] pour voir ce que l'appareil donne lorsqu'on le connecte. Appuyez deux fois sur le bouton inférieur. L'écran affiche alors une icône BLE. Vous remarquerez que l'application *nRF Connect* affiche des données spécifiques au fabricant et deux services BLE personnalisés : 0xfee7 et 0xfee0 (**figure 3**). Le nom de l'appareil est : LSLED. La connexion à celui-ci révèle ses services ainsi que les caractéristiques répertoriées dans le **tableau 1**.

Tableau 1. Caractéristiques du badge BLE

Service	Caractéristique	Propriété
0xfee7	0xfec7	ÉCRIRE
0xfee7	0xfec8	INDIQUER
0xfee7	0xfec9	LIRE
0xfee0	0xfee1	NOTIFIER, LIRE, ÉCRIRE

La lecture de la caractéristique 0xfec9 dans *nRF Connect for Mobile* renvoie la même valeur que les données spécifiques au fabricant. La lecture de 0xfee1 ne renvoie aucune donnée, et l'abonnement à ses notifications ne retourne rien non plus. La caractéristique *User Description* (description de l'utilisateur) de 0xfee1 est « Data ». Cette méthode ne donne rien d'utile.

Décompilation de l'application mobile

Puisque l'application du badge LED Bluetooth pour Android est évidemment capable de communiquer avec le badge LED, essayons de découvrir comment elle fonctionne réellement. Pour ce faire, nous allons la décompiler et fouiller dans les détails de son code source.

Téléchargez le fichier APK de l'application Android en utilisant un site de téléchargement d'APK tiers comme [2]. Il suffit de coller l'URL de l'application à partir du Google Play Store [3] dans le champ de recherche d'APKPure. Ensuite, vous pourrez télécharger le fichier. Ce fichier APK contient le code à octets Dalvik exécutable par Android. Pour comprendre ce que fait l'application, nous avons besoin de son code source. Bien que le développeur ait compilé son code source en Dalvik, nous pouvons le décompiler avec un décompilateur tel que jadx [4], qui convertit le code à octets Dalvik en code source Java.

Sous la version la plus récente de Windows [5], nous pouvons lancer l'interface graphique en double-cliquant sur le fichier **jadx-gui** dans le répertoire **bin**. Sous les autres systèmes d'exploitation (Linux ; MacOS), nous pouvons lancer l'interface graphique en exécutant bin/jadx-gui depuis la ligne de commande

Ouvrez le fichier APK téléchargé. Le programme décompile maintenant l'application et affiche une arborescence de ses paquets et fichiers Java à gauche. À ce stade-là, la recherche commence. Nous avons déjà

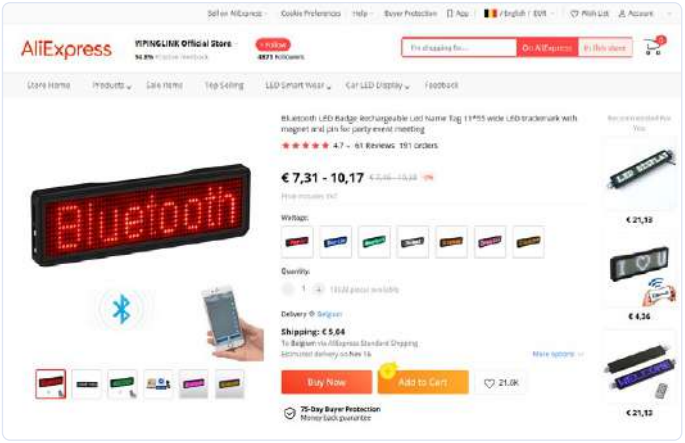


Figure 1. Ce badge LED Bluetooth semble être un appareil intéressant à rétroconcevoir !

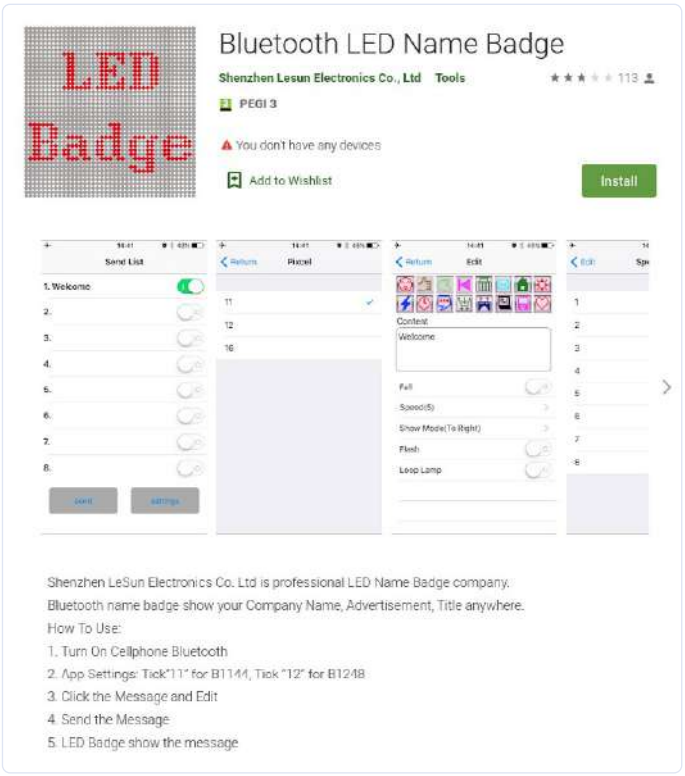


Figure 2. L'application « Bluetooth LED Name Badge app » dans le Google Play Store peut envoyer des commandes au badge LED.

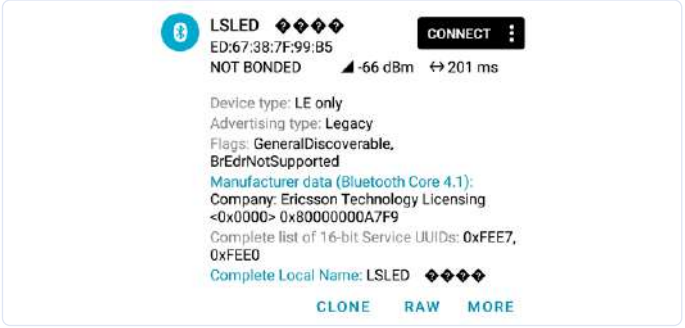


Figure 3. Le badge LED Bluetooth tel que détecté par nRF Connect for Mobile.

Listage 1.

```
public static final String UUID_CHARACTERISTICS_WRITE = "fee1";
public static final String UUID_SERVICE = "fee0";
```

Listage 2.

```
public static byte[] get64(List<SendContent> list, int i) {
    Iterator<SendContent> it = list.iterator();
    while (it.hasNext()) {
        Log.d("abcdef", "get64------------SendContent:" + it.next().toString());
    }
    byte[] bArr = new byte[64];
    bArr[0] = 119;
    bArr[1] = 97;
    bArr[2] = 110;
    bArr[3] = 103;
    bArr[4] = 0;
    bArr[5] = 0;
    bArr[6] = getFlash(list);
    bArr[7] = getMarquee(list);
    byte[] modeAndSpeed = getModeAndSpeed(list);
    for (int i2 = 0; i2 < 8; i2++) {
        bArr[i2 + 8] = modeAndSpeed[i2];
    }
    byte[] msgLength = getMsgLength(list, i);
    for (int i3 = 0; i3 < 16; i3++) {
        bArr[i3 + 16] = msgLength[i3];
    }
    bArr[32] = 0;
    bArr[33] = 0;
    bArr[34] = 0;
    bArr[35] = 0;
    bArr[36] = 0;
    bArr[37] = 0;
    byte[] date = getDate();
    for (int i4 = 0; i4 < 6; i4++) {
        bArr[i4 + 38] = date[i4];
    }
    for (int i5 = 0; i5 < 19; i5++) {
        bArr[i5 + 44] = 0;
    }
    bArr[63] = 0;
    return bArr;
}
```

quelques indices : les UUIDs des services et les caractéristiques trouvés dans *nRF Connect for Mobile*. Dans le menu **Navigation / Text search** (Navigation/Recherche de texte), vous pouvez entrer quelques termes de recherche.

Ceci est plus facile à dire qu'à faire. Le code ne mentionne pas le service 0xfee7 ni ses caractéristiques ! Il fait référence à l'autre service et à ses caractéristiques dans la classe `com.yannis.ledcard.ble.BleDevice`. Les éléments pertinents sont indiqués dans le **listage 1**.

Dans le code, vous pouvez maintenant rechercher ces deux constantes :
`UUID_CHARACTERISTICS_WRITE`
`UUID_SERVICE`
La façon la plus simple de le faire est probablement de cliquer avec le bouton droit de la souris sur le nom de la constante, puis de sélectionner **Find usage**. Cliquez ensuite sur l'une des instances trouvées pour ouvrir le fichier correspondant à cet emplacement.

En cherchant d'autres indices dans le code, je suis tombé sur un code intéressant qui traite les images et les modes d'affichage dans la classe `com.yannis.ledcard.util.LedDataUtil`.

Plus précisément, j'ai trouvé cette méthode Java donnée dans le **listage 2** qui crée une sorte d'en-tête pour les données.

Elle affiche un en-tête fixe (6 octets), un mode et une vitesse, une longueur de message et une date. Elle est appelée par `tSendHeader()` dans la classe mentionnée.

Procéder à l'ingénierie inverse en se plongeant dans le code et en analysant le fonctionnement de l'application n'était pas suffisant. J'ai donc commencé à utiliser l'application avec le badge LED tout en surveillant le trafic entre les deux appareils. Pour mieux comprendre, j'ai pu combiner les résultats obtenus à partir du code source de l'application avec le trafic en temps réel.

Renifler le trafic BLE entre le badge et l'application

Wireshark [6] renifle le trafic Bluetooth de votre téléphone en temps réel avec Android Debug Bridge. Connectez votre téléphone à votre ordinateur avec un câble USB et autorisez la connexion de débogage sur votre téléphone. Lancez Wireshark. Ce dernier devrait afficher **Android Bluetooth Btsnoop Net** comme l'une des interfaces disponibles. Double-cliquez sur l'interface et vous verrez défiler le trafic Bluetooth en direct ! Une meilleure solution consiste à utiliser le renifleur nRF pour le plugin BLE de Wireshark [7] avec un dongle nRF52840 comme renifleur.

Sur le badge BLE, appuyez deux fois sur le bouton inférieur jusqu'à ce qu'il affiche l'icône Bluetooth. Ensuite, sélectionnez le périphérique dans Wireshark pour afficher uniquement les paquets à destination et en provenance de ce périphérique spécifique.

Installez « Bluetooth LED Name Badge » sur votre téléphone Android et lancez l'application. Dans Wireshark, vous voyez que votre téléphone fait une demande de recherche et que le badge BLE affiche le nom de son appareil comme réponse. Sélectionnez le type de dispositif dans l'application. Pour le badge BLE de 11 x 55 pixels, le type est 11. Appuyez sur **yes** (oui).

L'application affiche une liste de messages à envoyer. La configuration par défaut est d'envoyer un message : **Welcome** (Bienvenue). Appuyez sur **Send** (Envoyer). L'application se connecte alors au badge LED et affiche le message de bienvenue sur son écran (**figure 4**).



Figure 4. Le badge LED Bluetooth affiche un message de bienvenue.

Dans Wireshark, nous voyons un paquet CONNECT_REQ suivi d'une demande d'attributs. Viennent ensuite quelques demandes d'écriture et de réponses. Pour obtenir une vision plus claire, cliquez avec le bouton droit de la souris sur *Write Request opcode* dans la section *Bluetooth Attribute Protocol* des détails du paquet, puis choisissez **Apply as Filter / Selected**. Vous pouvez également saisir : btatt.opcode == 0x12 pour filtrer l'affichage.

Dans ce cas, l'application envoie neuf paquets de demande d'écriture, chacun contenant 16 octets de données, comme indiqué ci-dessous :

```
77 61 6e 67 00 00 00 00 30 30 30 30 30 30 30 30
00 07 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 e5 0a 18 0c 37 25 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 c6 c6 c6 c6 d6 fe ee c6 82 00 00 00 00 00 7c
c6 fe c0 c6 7c 00 00 38 18 18 18 18 18 18 18 3c
00 00 00 00 00 7c c6 c0 c0 c6 7c 00 00 00 00 00
7c c6 c6 c6 c6 7c 00 00 00 00 00 ec fe d6 d6 d6
c6 00 00 00 00 00 7c c6 fe c0 c6 7c 00 00 00 00
```

Si vous vérifiez la méthode Java `get64()` de la section précédente, vous reconnaîtrez les parties de l'en-tête : 77 61 6e 67 (équivalent hexadécimal du décimal 119 97 110 103). Ensuite, on obtient 00 deux fois, et encore 00 deux fois, qui sont respectivement les valeurs pour `getFlash()` et `getMarquee()`. Les huit octets suivants sont tous 0x30, qui représentent le mode et la vitesse. Les 16 octets suivants représentent la longueur du message. Puisque l'application affiche une liste de huit messages, cela pourrait être la liste des longueurs de ces messages. Les deux premiers octets sont 00 07, ce qui correspond au nombre de caractères du texte **Welcome**. On peut supposer que les 14 octets suivants représentent les longueurs des sept messages suivants (aucun, dans ce cas).

Ensuite, il y a six 00, et les six octets suivants représentent la date, qui est e5 0a 18 0c 37 25, dans ce cas. Converti en décimal, ça fait 229 10 24 12 55 37. J'ai exécuté cette application le 24 octobre 2021, à 12 h 55 min 37 s. Le mois, le jour et l'heure étaient corrects. L'en-tête se termine par 20 00.

Après cette étape, cinq paquets de 16 octets représentent en quelque sorte les sept caractères du texte **Welcome**. Découvrons comment cela se passe, en commençant par un message plus simple. Ouvrez de nouveau l'application et cliquez sur le premier message. Remplacez le texte par **W** et activez **Flash**. Revenez à l'écran principal, cliquez sur le deuxième message et ajoutez le texte : **e**. Pour ce deuxième message, réglez la vitesse sur 8, le mode sur **Right** et activez **Marquee**. Retournez à l'écran principal. Activez ensuite le curseur à côté du deuxième message et cliquez sur **Send** (envoyer) tout en observant les paquets du *Bluetooth Attribute Protocol* dans Wireshark.

Là, le badge LED affiche la lettre *W* qui se déplace vers la gauche et clignote. Ensuite, il affiche la lettre *e* se déplaçant vers la droite à double vitesse et un cadre de pixels mobiles aux alentours. Dans ce cas, l'application a envoyé six paquets de demande d'écriture, chacun contenant 16 octets de données – voir ci-dessous :

```
77 61 6e 67 00 00 01 02 30 71 30 30 30 30 30 30
00 01 00 01 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 e5 0a 18 10 06 13 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 c6 c6 c6 c6 d6 fe ee c6 82 00 00 00 00 00 7c
c6 fe c0 c6 7c 00 00 00 00 00 00 00 00 00 00 00
```

Le premier paquet commence avec les mêmes 6 octets, mais suivi de la valeur de `getFlash()` qui est 01 et celle de `getMarquee()` qui est 02. C'est parce que nous avons activé **Flash** pour le premier message et **Marquee** pour le deuxième message.

L'octet suivant est toujours 0x30, car nous n'avons rien changé quant au mode et à la vitesse du premier message. Mais l'octet suivant est désormais 0x71. Nous avons changé la vitesse à 8 et le mode à **Right** (droite). Donc, le quartet de gauche de cet octet code apparemment, la vitesse (la vitesse 4 est codée par 3 et la vitesse 8 par 7) et le quartet de droite code le mode. Le deuxième paquet indique 00 01 comme longueur du premier message et la même valeur pour la longueur du deuxième message. Cela confirme notre hypothèse. Ensuite, la date est codée.

Il reste maintenant deux paquets. Convertissez les valeurs hexadécimales en binaire et formatez-les octet par octet et en séquence, puis divisez-les en onze lignes (oui, parce que l'écran fait 11 pixels de haut) comme suit :

```
00000000
11000110
11000110
11000110
11000110
11010110
11111110
11101110
11000110
10000010
00000000

00000000
00000000
00000000
00000000
01111100
11000110
11111110
11000000
11000110
01111100
00000000
```

Listage 3.

```
"""Find BLE LED badges.
Copyright (c) 2022 Koen Vervloesem
SPDX-License-Identifier: MIT
"""
import asyncio
from bleak import BleakScanner
num_devices = 0
def device_found(device, advertisement_data):
    """Show device details if it's a BLE LED badge."""
    global num_devices
    if device.name.startswith("LSLED"):
        num_devices += 1
        print(
            f" () - RSSI: "
        )
async def main():
    """Scan for BLE devices."""
    print("Searching for LED badges...")
    scanner = BleakScanner()
    scanner.register_detection_callback(device_found)
    await scanner.start()
    await asyncio.sleep(5.0)
    await scanner.stop()
    if not num_devices:
        print("No devices found")
if __name__ == "__main__":
    asyncio.run(main())
```

Vers la fin, des 0 de remplissage sont utilisés pour remplir un bloc de 16 octets (non représentés ici). Si vous appliquez le même décodage au premier message avec **Welcome** et que vous placez les bitmaps des lettres les uns à côté des autres, vous obtiendrez un 1 pour chaque pixel activé et un 0 pour chaque pixel désactivé, comme illustré ici :

```
00000000000000000000000000000000000000000000000000000000000000000
11000110000000000011100000000000000000000000000000000000000000000
11000110000000000011000000000000000000000000000000000000000000000
11000110000000000011000000000000000000000000000000000000000000000
11000110011111100001100001111100011111001110110001111100
11010110110001100001100011000110110001101111111011000110
11111110111111100001100011000000110001101101011011111110
11101110110000000001100011000000110001101101011011000000
11000110110001100001100011000110110001101101011011000110
10000010011111000011110001111100011111001100011001111100
00000000000000000000000000000000000000000000000000000000000000000
```

Vous avez maintenant une bonne idée du format des données à envoyer au badge LED afin d'afficher du texte sur l'écran. Rassemblons tout cela et créons un script Python pour envoyer vos propres images au badge LED.

Création d'images aléatoires sur le badge LED avec *Bleak*

Le badge LED à Bluetooth vous permet également d'envoyer des petits bitmaps 11 x 11 prédéfinis. Mais maintenant que vous connaissez le format de données à envoyer, vous pouvez également choisir un bitmap arbitraire, par exemple, pour couvrir la totalité de l'écran 11 x 55. Faisons cela avec *Bleak* [8], une bibliothèque Python multiplateforme pour BLE. Saisissez :

```
pip3 install bleak
```

Créons d'abord un script qui recherche les badges BLE en fonction de leur nom pendant cinq secondes, puis affiche leur adresse Bluetooth et leur RSSI – voir **listage 3**.

Si vous l'exécutez avec deux badges LED qui sont tous deux en mode **Listen**, le script affiche :

```
$ python3 find_led_badge.py
Searching for LED badges...
ED:67:38:80:0D:E2 (LSLED) - RSSI: -74
ED:67:38:7F:99:B5 (LSLED) - RSSI: -83
```

Exploitons maintenant les connaissances acquises dans la section précédente. Examinons ce script qui envoie les commandes appropriées au badge LED pour afficher une image – voir **listage 4**.

Outre *Bleak*, ce script utilise également la bibliothèque *Pillow* [9] :

```
pip3 install Pillow
```

Les quatre premières commandes BLE sont codées en dur. Dans la première commande, 0x34 définit le mode et la vitesse du premier message. 4 signifie un mode statique : le message est affiché sous forme d'image fixe. Dans la deuxième commande, 0x07 définit la longueur du message et le nombre de caractères de 8 pixels. L'écran a une largeur de 55 pixels, et 7 x 8 = 56, nous devons donc coder l'image en 7 caractères. Les deux commandes suivantes devraient encoder la date actuelle, mais ce n'est pas vraiment nécessaire. Il suffit d'envoyer seize 0 pour chacune de ces commandes.

`chunks()` est une fonction auxiliaire pour diviser un tableau d'octets en une liste de tableaux, chacun de 16 octets. Elle est utilisée dans la fonction `commands_for_image()` qui convertit un fichier image en octets représentant des caractères sur l'écran. À la fin de la fonction, ces octets sont divisés en regroupement de 16 octets.

`commands_for_image()` ouvre une image avec *Pillow* et charge ses pixels. Pour chacun des sept caractères de l'écran, elle code le caractère sous forme de 11 octets : un pour chaque ligne. Tous ces octets sont ajoutés à `octets_image`.

À la fin de l'image, le tableau d'octets est complété par des octets supplémentaires afin d'obtenir un multiple de 16 octets. Et, finalement, il est divisé en bloc de 16 octets.

La fonction `main()` concrétise tout cela. Elle se connecte au périphérique et crée une liste de commandes : les quatre commandes de l'en-tête sont étendues avec les commandes pour l'image. Ensuite, chacune de ces commandes est écrite dans la caractéristique avec l'UUID 0xfee1. Le code principal en bas de page vérifie si vous avez

Listage 4.

```
"""Display an image on a BLE LED badge.
Copyright (c) 2022 Koen Vervloesem
SPDX-License-Identifier: MIT
"""
import asyncio
import sys
from PIL import Image
import bleak
WRITE_CHAR_UUID = "0000fee1-0000-1000-8000-00805f9b34fb"
COMMAND1 = bytes([0x77, 0x61, 0x6E, 0x67, 0x00, 0x00, 0x00, 0x00, 0x34, 0x30, 0x30, 0x30, 0x30, 0x30, 0x30, 0x30])
COMMAND2 = bytes([0x00, 0x07, *(14 * [0x00])])
COMMAND3 = bytes(16 * [0x00])
COMMAND4 = bytes(16 * [0x00])
def chunks(lst, n):
    """Yield successive n-sized chunks from lst."""
    for i in range(0, len(lst), n):
        yield lst[i : i + n]
def commands_for_image(image):
    """Return commands to show an image on the BLE LED badge."""
    image_bytes = bytearray()
    with Image.open(image) as im:
        px = im.load()
        for i in range(7):  # 7x8 = 56 -> 7 bytes next to each other
            for row in range(11):
                row_byte = 0
                for column in range(8):
                    try:
                        pixel = int(
                            px[(i * 8) + column, row][3] / 255
                        )
                    except IndexError:
                        pass  # Ignore the 56th pixel in a full row
                    row_byte = row_byte | (pixel << (7 - column))
                image_bytes.append(row_byte)
    # Fill the end with zeroes to have a multiple of 16 bytes
    image_bytes.extend(bytes(16 - len(image_bytes) % 16))
    # Split image bytes into 16-byte chunks
    return list(chunks(image_bytes, 16))
async def main(address, filename):
    """Connect to BLE LED badge and send commands to show an image."""
    try:
        async with bleak.BleakClient(address) as client:
            commands = [COMMAND1, COMMAND2, COMMAND3, COMMAND4]
            commands.extend(commands_for_image(filename))
            for command in commands:
                await client.write_gatt_char(WRITE_CHAR_UUID, command)
    except asyncio.exceptions.TimeoutError:
        print(f"Can't connect to device .")
    except bleak.exc.BleakError as e:
        print(f"Can't write to device : ")
if __name__ == "__main__":
    if len(sys.argv) == 3:
        address = sys.argv[1]
        filename = sys.argv[2]
        asyncio.run(main(address, filename))
    else:
        print(
            "Please specify the BLE MAC address and image filename."
        )
```

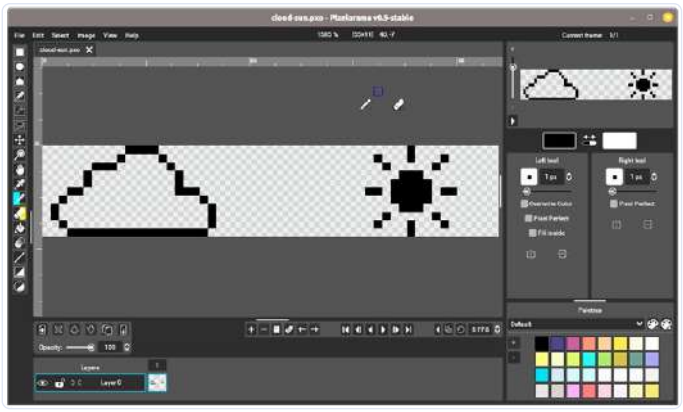


Figure 5. Pixelorama est un éditeur d'images 2D à code source ouvert.



Figure 6. En envoyant les commandes BLE appropriées, vous pouvez afficher vos propres images de 11 x 55 pixels sur le badge LED.

fourni deux attributs sur la ligne de commande. Le premier est attribué à l'adresse Bluetooth, et le second, au nom du fichier.

Avant d'exécuter ce code, vous devez préparer une image d'exactement 11 x 55 pixels, avec un éditeur de pixels tel que Pixelorama [10]. Définissez un pixel pour chaque LED que vous souhaitez allumer sur l'écran. La **figure 5** montre un exemple d'une image de nuages et de soleil que j'ai créée dans Pixelorama.

Exportez l'image sous forme de fichier .PNG. Ensuite, mettez le badge LED en mode **Discovery** en appuyant deux fois sur le bouton du bas jusqu'à ce que l'icône Bluetooth apparaisse. Exécutez le script Python avec deux paramètres : l'adresse Bluetooth du badge et le nom de fichier de l'image :

```
$ python3 display_led_badge.py ED:67:38:7F:99:B5 cloud-sun.png
```

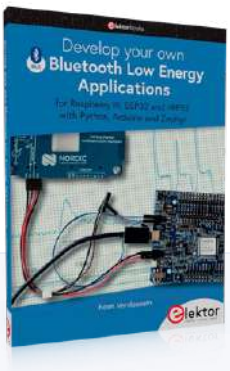
Conclusion

Le badge BLE devrait afficher votre image (**figure 6**). Voilà, vous avez réussi à rétro concevoir le badge BLE et vous pouvez dès maintenant l'utiliser dans votre propre code. Bien entendu, il est possible d'améliorer le script que je vous ai proposé, notamment pour le rendre plus convivial. J'espère que cela vous incitera à appliquer le même principe de rétro-ingénierie à d'autres appareils BLE.

220439-04 — VF : Asma Adhimi

Des questions, des commentaires ?

Envoyez un courriel à l'auteur (koen@vervloesem.eu) ou contactez Elektor (redaction@elektor.fr).



Produits

- **Livre en anglais « Develop your own Bluetooth Low Energy Applications », K. Vervloesem, Elektor 2022 (SKU 20200)**
  www.elektor.fr/20200

Ce livre est livré avec un dongle USB nRF52840 GRATUIT !

- **Version numérique (SKU 20201)**
  www.elektor.fr/20201

LIENS

[1] L'application nRF Connect for Mobile : https://www.nordicsemi.com/Products/Development-tools/nrf-connect-for-mobile
[2] Fichier APK : https://apkpure.com
[3] L'application sur Google Play Store : https://play.google.com/store/apps/details?id=com.yannis.ledcard
[4] Le décompilateur jadx : https://github.com/skylot/jadx
[5] Dernière version du décompilateur jadx : https://github.com/skylot/jadx/releases
[6] Wireshark : https://www.wireshark.org
[7] nRF Sniffer for Bluetooth LE : https://www.nordicsemi.com/Products/Development-tools/nrf-sniffer-for-bluetooth-le
[8] Bleak : https://bleak.readthedocs.io
[9] La bibliothèque Pillow : https://pillow.readthedocs.io
[10] Pixelorama : https://orama-interactive.itch.io/pixelorama
[11] Page de ressources/informations sur le livre : https://www.elektor.fr/develop-your-own-bluetooth-low-energy-applications
[12] Le code sur GitHub : https://github.com/koenvervloesem/bluetooth-low-energy-applications

kit de développement MakePython ESP32

tout dans une boîte

Tam Hanna (Slovaquie)

Les microcontrôleurs modernes, tels que l'ESP32, sont si puissants qu'ils peuvent être programmés en MicroPython. Grâce aux puissantes bibliothèques, cela vous permet de réaliser rapidement un projet. Avec le kit de développement MakePython ESP32, qui est d'une part, un manuel et d'autre part, un kit matériel, Dogan Ibrahim, auteur bien connu d'Elektor, nous présente le MicroPython avec des exemples concrets.



Nous n'avons pas vraiment besoin de discuter du fait que MicroPython n'est pas la voie vers une efficacité logicielle maximale. D'un autre côté, il est vrai que les microcontrôleurs modernes tels que l'ESP32 sont plus que capables de suivre le rythme d'un x486 en termes de performances. En particulier pour réaliser de petites séries, il peut donc être raisonnable d'échanger l'effort de programmation contre la vitesse en utilisant des langages de haut niveau.

Le manuel

Commençons par le manuel, qui est disponible en anglais : il est livré avec un design assez fantaisiste, qui fait croire à une (fausse) reliure en spirale.

La partie didactique commence par une brève explication de l'ESP32 dans son ensemble. Si vous êtes un lecteur d'Elektor et que vous avez déjà une expérience approfondie ou au moins basique des microcontrôleurs, vous aurez un aperçu « rapide » des différents périphériques fournis par Espressif dans le contrôleur. Si vous n'avez pas du tout d'expérience, les explications pourraient parfois être trop courtes pour une compréhension solide.

L'installation des EDI est expliquée en détail pour Windows, tandis que Linux est à peine abordé. Ibrahim présente à la fois uPyCraft et Thonny, mais ne travaille ensuite presque exclusivement qu'avec Thonny. S'ensuit un chapitre qui illustre l'exécution de quelques petits extraits Python à l'aide de Thonny. Si vous n'avez aucune connaissance de la syntaxe Python, vous ne pouvez pas vraiment vous lancer à ce stade. En revanche, les explications sont plus que suffisantes pour comprendre les bases de MicroPython et de l'EDI.

La présentation proprement dite des 46 projets contenus dans le livre se fait ensuite dans le style classique de Dogan Ibrahim : dans un premier temps, le professeur présente toujours la tâche à accomplir, puis il présente le code et les explications de la conception. Les amateurs de codage compact en particulier pourraient être quelque peu agacés par le fait que les listings comportent toujours



Figure 1. Le kit. L'emballage en plastique protège de manière fiable les composants contre les dommages.



Figure 2. Cette carte permet un démarrage rapide avec MicroPython.

un en-tête standardisé d'une bonne dizaine de lignes. D'autre part, les exemples, sans être compliqués, illustrent bien les aspects essentiels du MicroPython.

Après l'avoir lu, vous n'aurez certainement plus à craindre la recherche désespérée de la manière de mettre en service certains périphériques. De l'avis de l'auteur, le fait qu'Ibrahim explique comment restaurer une carte « ratée » avec un nouveau firmware MicroPython est tout à fait louable.

Un regard sur la carte

En parlant de la carte à évaluer, le manuel est disponible en combinaison avec le kit de la **figure 1**, qui convient également pour être emporté en vacances grâce à son emballage plastique assez robuste. Mais voyons maintenant le matériel proprement dit. L'expérience de l'auteur en matière d'enseignement a montré que les développeurs qui ne sont pas familiers avec les systèmes embarqués auront une expérience d'apprentissage plus confortable si la plate-forme de développement fournit un écran (idéalement entièrement graphique).

Elektor contourne astucieusement ce problème en fournissant le circuit imprimé rouge présenté sur la **figure 2**. Le pavé noir est un écran OLED populaire à SSD1306 avec une résolution de 128 par 64 pixels.

C'est ici que se cache le plus gros point de critique : les connecteurs de 2,54 mm ne sont pas soudés. C'est dommage, car une personne qui achète le kit dans un magasin et l'emporte avec elle en vacances est bien embêtée si elle n'a pas de fer à souder à portée de main et que le réparateur de téléphones portables du village ne veut pas l'aider.

Pour le reste, la carte n'a aucune raison d'être critiquée. La variante ESP32 utilisée est l'un des plus grands modèles et dispose d'une mémoire suffisante. L'interface MicroUSB permet d'utiliser le câble de n'importe quel ancien téléphone portable. En outre, le kit comprend également un câble (très court) de qualité décente.

Une démo

Bien que le manuel soit principalement orienté vers les besoins des développeurs travaillant sous Windows, le testeur se référera à Ubuntu 20.04 LTS dans les étapes suivantes, ne serait-ce que pour des raisons de commodité. L'EDI disponible à l'adresse [1] est un choix plutôt mauvais, car son exécution dans toutes les versions d'Ubuntu postérieures à 16.04 échoue avec un message d'erreur du type :

```
ImportError: /tmp/_MEIOhQKhz/libz.so.1: version
`ZLIB_1.2.9' not found (required by /usr/lib/x86_64-
linux-gnu/libpng16.so.16)
```

Une meilleure option est Thonny, qui peut être installé automatiquement en entrant :

```
bash <(wget -O - https://thonny.org/installer-for-linux)
```

Et peut ensuite être lancé en entrant la commande :

```
/home/tamhan/apps/thonny/bin/thonny
```

Pour désinstaller, on utilise la commande :`/home/tamhan/apps/thonny/bin/uninstall`. Après le premier démarrage, il ne reste plus qu'à confirmer le choix de la langue.

L'étape suivante consiste à connecter la carte à une station de travail. Heureusement, Elektor a bien pensé à livrer la carte avec un *runtime* MicroPython préconfiguré. Dans ce contexte, il est particulièrement pratique que la carte allume en même temps l'écran, ce qui facilite la vérification des fonctions. À propos, une puce pont USB-UART CP210x de Silicon Labs est utilisée comme convertisseur.

À ce stade, vous pouvez passer sur Thonny (**figure 3**). Veillez à cliquer sur la version de Python dans le menu de sélection qui apparaît en bas à droite et sélectionnez la version *MicroPython (ESP32)*. Thonny recherchera alors automatiquement les cartes ESP32.

La question de savoir si vous devez exécuter Thonny avec les privilèges de superutilisateur est discutable. Le compte utilisateur de l'auteur étant membre du groupe *plugdev*, il a pu exécuter Thonny avec son compte utilisateur normal et interagir avec l'ESP32.

Mon premier programme en MicroPython

Comme prochaine action, passons à la mise en service de l'écran OLED de l'appareil. Pour ce faire, nous ouvrons l'URL : *www.makerfabs.com/makepython-esp32-starter-kit.html*. Télécharger l'archive *MakePython ESP32 Lessons Source Code*. L'utilisation de RAR est un peu ennuyeuse sous Linux. Dans tous les cas, extrayez l'archive à un endroit facilement accessible. Pour les étapes

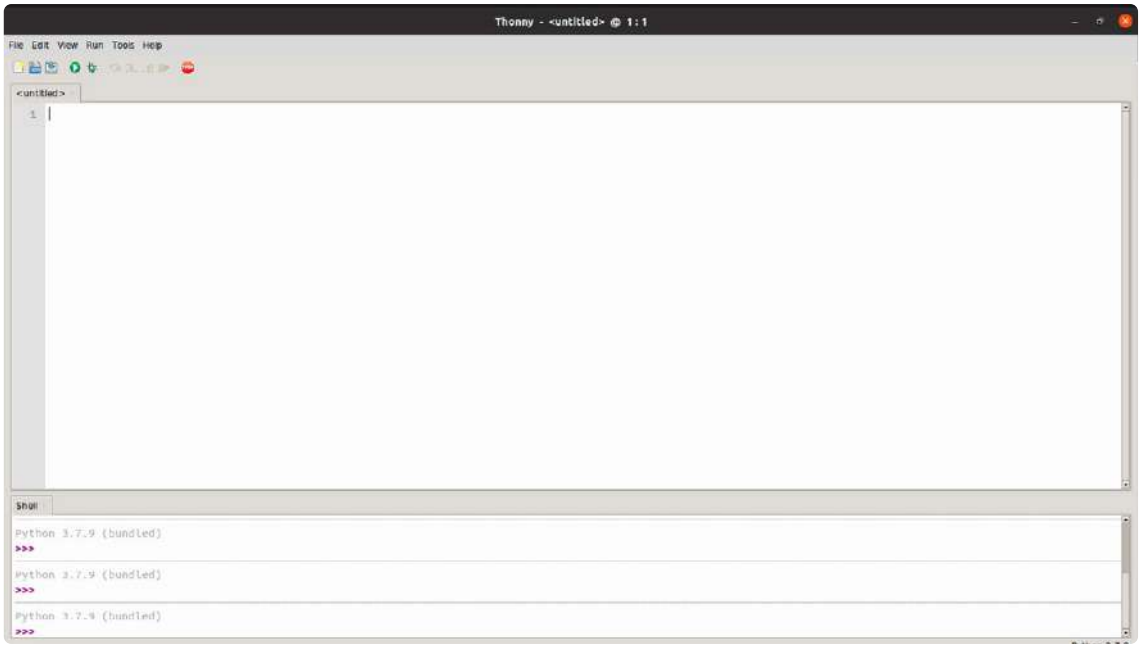


Figure 3. Capture d'écran de Thonny.

suivantes, nous avons particulièrement besoin du fichier *ssd1306.py*, qui fournit le code du pilote OLED.

Dans la première étape, cliquez sur *File -> Open*, puis choisissez l'option *This Computer*. Dans l'étape suivante, on navigue vers le fichier pour le charger dans l'éditeur. Ensuite, nous choisissons *File -> Save as* et enfin l'option *MicroPython Board*.

Si Thonny s'énerve à ce stade parce que le backend est *busy*, vous pouvez cliquer sur l'icône rouge d'arrêt dans la barre d'outils pour ordonner un arrêt. À l'étape suivante, vous enregistrez le fichier sous son nom d'origine (si vous l'avez modifié). Sinon, le fichier est déjà en place.

Comme c'est souvent le cas avec MicroPython, nous devons d'abord installer certaines bibliothèques et définir des constantes contenant des informations supplémentaires sur les spécifications de l'écran :

```
import machine
import ssd1306
import time
WIDTH = const(128)
HEIGHT = const(64)
```

Pour la communication proprement dite avec l'écran connecté via I²C, ce qui n'est pas courant, car on s'attend généralement à ce qu'il s'agisse de SPI, nous utilisons ensuite une instance de la classe I²C du logiciel :

```
pscl = machine.Pin(5, machine.Pin.OUT)
psda = machine.Pin(4, machine.Pin.OUT)
i2c = machine.I2C(scl=pscl, sda=psda)
oled = ssd1306.SSD1306_I2C(WIDTH, HEIGHT, i2c)
```

Enfin, nous devons envoyer du texte à l'écran selon le schéma suivant :

```
while True:
    oled.fill(0)
    oled.text('Hello World!',10,0)
    oled.show()
    time.sleep(1)
```

À ce stade, vous pouvez cliquer sur l'icône de lecture dans l'EDI pour voir le texte apparaître sur l'écran de la carte.

Ça se corse : MQTT

Comme prochaine tâche, nous voulons aller un peu au-delà de la portée du projet de base inclus dans le kit et réaliser une tâche avancée. Cela montre également que le kit est adapté à des tâches plus compliquées. Plus précisément, nous voulons utiliser le protocole MQTT, universellement connu. La tâche devient particulièrement intéressante du fait que nous voulons travailler avec une toute nouvelle version du broker MQTT Mosquitto, disponible. À l'URL [2] vous trouvez une description des problèmes que les *parser* (qui sont configurés de plus en plus strictement pour des raisons de sécurité) peuvent rencontrer en interaction avec les pilotes MicroPython.

En ce qui concerne les pilotes MicroPython, il existe désormais deux implémentations concurrentes de MQTT pour MicroPython. Cependant, nous voulons ici nous appuyer sur la variante la plus simple disponible sur [3]. Télécharger le fichier :

https://github.com/micropython/micropython-lib/blob/master/micropython/umqtt.simple/umqtt/simple.py

Et enregistrez-le sous le nom *simple.py* sur l'ordinateur de traitement.

Dans l'étape suivante, nous devons mettre les émetteurs wifi de la carte en mode station selon le schéma suivant :

```
. . .
wlan=network.WLAN(network.STA_IF)
```

Pour l'établissement effectif de la connexion, nous utilisons ensuite une méthode construite selon le schéma suivant (largement reprise du projet fourni par MakerFabs et implémente un compte à rebours

Listage 1.

```
def connectWiFi():
  i=0
  wlan.active(True)
  wlan.disconnect()
  wlan.connect("ssid","pass")
  while(wlan.ifconfig()[0]=='0.0.0.0'):
    i=i+1
    oled.fill(0)
    oled.text('connecting WiFi',0,16)
    oled.text('Countdown:'+str(20-i)+'s',0,48)
    oled.show()
    time.sleep(1)
    if(i>20):
      break
  oled.fill(0)
  oled.text('Makerfabs',25,0)
  oled.text('MakePython ESP32',0,32)
  if(i<20):
    oled.text('WIFI connected',0,16)
  else:
    oled.text('NOT connected!',0,16)
  oled.show()
  time.sleep(3)
  return True
```

incluant l'établissement de la connexion). Bien sûr, assurez-vous d'adapter les chaînes passées à `wlan.connect` à votre situation de réseau sans fil (voir **listage 1**).

L'établissement de la connexion de test proprement dite ne peut être plus simple :

```
connectWiFi()
while True:
    time.sleep(1)
```

Dans l'étape suivante, nous voulons nous doter d'un serveur Mosquitto. Bien qu'il ne soit pas le broker MQTT le plus rapide, il est open source, gratuit et très strict dans son implémentation MQTT.

Grâce à son utilisation répandue, il existe une image plus ou moins clés en main dans le Docker Hub qui permet le déploiement du serveur sans configurations profondes de l'ordinateur hôte.

Pour une première tentative, il suffit d'entrer la commande suivante :

```
docker run -it -p 1883:1883 -p 9001:9001 -v mosquitto.
conf:/mosquitto/config/mosquitto.conf eclipse-mosquitto
```

Tout d'abord, les paramètres `-p 1883:1883` and `-p 9001:9001` sont importants ici. Après tout, chaque conteneur Docker est doté d'un ensemble complet de ports TCP/IP par le *runtime*. Chacun de ces paramètres connecte ensuite l'un de ces ports à la carte réseau de l'ordinateur hôte. Via le paramètre `-v mosquitto.conf:/mosquitto/config/mosquitto.conf`, nous intégrons également un fichier de configuration.



À ce stade, c'est une bonne idée de faire une première tentative de téléchargement avec une connexion Internet existante. Normalement, Docker se connectera au hub et téléchargera les composants requis, puis quittera avec le message d'erreur suivant :

```
docker: Error response from daemon: source /var/lib/
docker/aufs/mnt/a22b9e557c37d99eb71f17e7bc6d38df6e7677
d09225376d416612adf0977ccd/mosquitto/config/mosquitto.
conf is not directory.
```

La cause de l'erreur est qu'il n'existe pas encore de fichier nommé *mosquitto.conf* sur le poste de travail hôte que nous pourrions mettre à la disposition du conteneur.

Pour résoudre le problème, il suffit de créer un nouveau fichier via la ligne de commande dans le répertoire de travail avec `gedit` :

```
(base) tamhan@tamhan-thinkpad:~$ gedit mosquitto.conf
```

Le passage de la version 1 de Mosquitto à la version 2 s'est accompagné d'un renforcement massif dans le domaine de la configuration de la sécurité. Alors qu'un Mosquitto 1.x (démarré avec les paramètres par défaut) acceptait les requêtes du serveur provenant de n'importe quel client, Mosquitto 2 le refuse. Toutefois, si vous placez les lignes suivantes dans le fichier de configuration, tout fonctionnera à nouveau comme d'habitude (et de manière non sécurisée) :

```
listener 1883
allow_anonymous true
```

L'exécution réelle peut alors être commandée selon le schéma suivant. Notez que le paramètre `-v` nécessite toujours un chemin d'accès complet, c'est pourquoi nous appelons la commande `pwd` et assemblons les résultats par la magie du shell. D'ailleurs, `pwd` signifie *Print Working Directory*, la **figure 4** montre le comportement :

```
(base) tamhan@tamhan-thinkpad:~$ docker run -it -p
1883:1883 -p 9001:9001 -v $(pwd)/mosquitto.conf:/
mosquitto/config/mosquitto.conf eclipse-mosquitto
. . .
```

Si le conteneur Docker confirme le succès de son démarrage en renvoyant `1658673183: mosquitto version 2.0.14 running`, nous sommes prêts à ce stade.

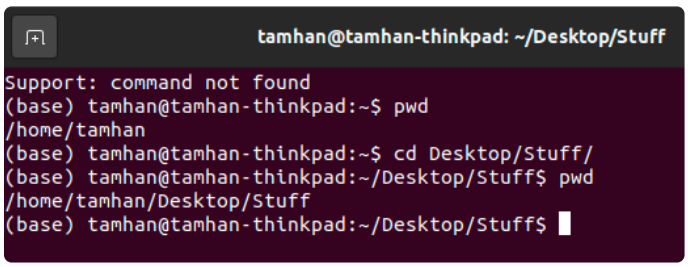


Figure 4. La commande `pwd` renvoie le répertoire de travail actuel du shell.



Utiliser MQTT sur l'ESP32

Tout d'abord, nous avons besoin de quelques constantes pour travailler avec MQTT. Veillez à adapter l'adresse IP transmise dans la chaîne de caractères du serveur à votre situation d'exploitation locale :

```
SERVER = "192.168.178.146"
CLIENT_ID = ubinascii.hexlify(machine.unique_id())
TOPIC = b"led"
state = 0
```

Nous avons également besoin d'une fonction de rappel (*callback*) que le pilote MQTT appellera lorsqu'un message est reçu :

```
def sub_cb(topic, msg):
    global state
    print((topic, msg))
    . . .
```

La configuration réelle de la connexion MQTT est alors relativement simple :

```
connectWiFi()
c = MQTTClient(CLIENT_ID, SERVER, keepalive=30)
c.set_callback(sub_cb)
c.connect()
c.subscribe(TOPIC)
print("Connected to %s, subscribed to %s topic" % (SERVER, TOPIC))

while True:
    c.wait_msg()
```

L'appel de `c.subscribe` garantit que le pilote MQTT informe le serveur des canaux de messages qui l'intéressent actuellement. Il est également important d'appeler périodiquement la méthode `c.wait_msg()` afin de fournir au serveur MQTT du temps CPU pour traiter les informations entrantes ou à envoyer. Dans la ligne de commande ou dans la fenêtre de sortie, vous pouvez alors voir le message `Connected to 192.168.178.146, subscribed to b'led' topic`.

Dans l'étape suivante, nous voulons connecter l'une des LED incluses, ainsi que la résistance incluse, à la broche GPIO 12. L'auteur suppose que le lecteur a des connaissances suffisantes en électronique. L'initialisation du port GPIO ne nécessite alors que du code ESP32 ordinaire :

```
led = Pin(12, Pin.OUT, value=1)
```

Pour traiter les messages entrants, nous devons ensuite étendre la fonction de rappel en combinant la chaîne fournie dans la valeur `topic` avec les constantes fournies pour les différentes commandes :

```
def sub_cb(topic, msg):
    global state
    print((topic, msg))
    if msg == b"on":
        led.value(1)
        state = 1
    elif msg == b"off":
        led.value(0)
        state = 0
    elif msg == b"toggle":
        led.value(state)
```

Pour les tests réels, nous voulons utiliser l'utilitaire de ligne de commande *mosquitto_pub* : c'est un outil disponible sous Linux qui permet d'envoyer des commandes directement à un serveur MQTT. Les commandes suivantes sont ensuite utilisées pour allumer et éteindre la LED :

```
(base) tamhan@tamhan-thinkpad:~$ mosquitto_pub -t led
-m 'on'
(base) tamhan@tamhan-thinkpad:~$ mosquitto_pub -t led
-m 'off'
```

Si l'ESP32 et le serveur sont dans le même réseau, vous pouvez maintenant allumer et éteindre la LED.

Conclusion et perspectives

Le kit de développement MakePython ESP32 est une boîte à outils fascinante et compacte qui permet aux développeurs Python et/ou aux électroniciens de réunir rapidement les deux mondes. En particulier parce que tout est si joliment emballé de manière pratique, c'est un produit que l'auteur est heureux de recommander.

220429-04 — VF : Maxime Valens

Produits

- **Kit de développement MakePython ESP32 (SKU 20137)**
  www.elektor.fr/20137

LIENS

[1] EDI uPyCraft : https://github.com/DFRobot/uPyCraft
[2] umqtt.simple et mosquitto 2.0.12 : https://github.com/micropython/micropython-lib/issues/445
[3] umqtt.simple - client MQTT simple pour MicroPython :
https://github.com/micropython/micropython-lib/tree/master/micropython/umqtt.simple



machines à vision totale

la technologie derrière les systèmes de vision industrielle d'aujourd'hui

Stuart Cording (Elektor)

Avec l'avènement des capteurs d'images CMOS, il est désormais facile de produire en masse des caméras de faible consommation grâce à un processus économique de fabrication de semi-conducteurs. Aujourd'hui, nous avons tous au moins deux appareils de ce type dans nos smartphones. Grâce à des microcontrôleurs toujours plus puissants, la capture d'images et le traitement d'objets sont à la portée de tous, et ils prennent même en charge les mesures 3D dans les capteurs industriels.

De nos jours, on n'est jamais loin d'une caméra. Si vous n'en avez pas dans votre poche, il y a de fortes chances que votre voisin dans le train en ait une. Et, si ce n'est pas le cas, il y en a probablement une dans le wagon, à la gare ou dans le système de sécurité du café devant lequel vous êtes passé en arrivant. Les caméras sont devenues aussi omniprésentes grâce aux capteurs d'images CMOS, une technologie qui a considérablement réduit leur coût, leur consommation d'énergie et leur taille par rapport à l'ancienne technologie des dispositifs à transfert de charge (CCD) qu'ils ont remplacée. Avec les progrès dans la capture d'images et de vidéos, la croissance rapide de la puissance des ordinateurs au cours de la dernière décennie a permis de traiter les images en temps réel. Cela permet d'automatiser l'inspection dans diverses applications industrielles, comme la détection de composants mal positionnés sur des circuits imprimés, la localisation de pièces sur des bandes transporteuses et même l'évaluation de la qualité d'un joint soudé.

CCD – Technologie d'imagerie du passé

Les capteurs d'images utilisent un réseau de photodiodes pour convertir les photons en un courant électrique. Dans les CCD, les photodiodes d'une ligne sont reliées par des interrupteurs. Une fois le capteur illuminé, les charges formées dans chaque pixel sont transférées au pixel voisin par des impulsions de commande successives sur les interrupteurs, jusqu'à arriver au bord du capteur (**figure 1**). Là, les charges sont converties une par une en tension, puis en valeur numérique par un convertisseur analogique-numérique (CA/N). Les capteurs CCD sont fabriqués selon un procédé métal-oxyde-semiconducteur (MOS), qui diffère de la technologie MOS complémentaire (CMOS) utilisée pour la plupart des dispositifs en silicium.

La longévité du CCD en tant que technologie de capteur est liée à son rendement quantique élevé (son taux de conversion des photons en électrons), qui peut atteindre 95 %. Au-delà des applications commerciales et grand public, les CCD ont été utilisés dans des appareils scientifiques, tels que des spectromètres et même le télescope Hubble [1]. Cependant, l'exploitation des images qu'ils fournissent nécessitait des circuits externes.

Les capteurs d'images CMOS sont devenus une technologie viable dans les années 1990. Au départ, la qualité de l'image était médiocre, avec une faible résolution, des images bruitées et une mauvaise définition des couleurs. En raison d'inévitables différences dans la conception, les photodiodes de ces capteurs étaient beaucoup plus petites que celles d'un capteur CCD ayant un nombre équivalent de pixels. Plutôt que de mettre en œuvre un circuit de transfert de charge, chaque pixel nécessitait son propre amplificateur, ce qui réduisait la surface disponible pour chaque pixel (**figure 2**). Cependant, grâce à un gros effort de recherche et de développement pour satisfaire les besoins du marché, la différence de qualité d'image entre les deux technologies est aujourd'hui minime. La facilité de production due au processus de fabrication CMOS a également fait baisser les prix, avec l'intégration dans la puce du capteur d'une partie croissante du traitement de l'image. En outre, la consommation d'énergie n'est que le centième de celle d'un capteur CCD équivalent, ce qui

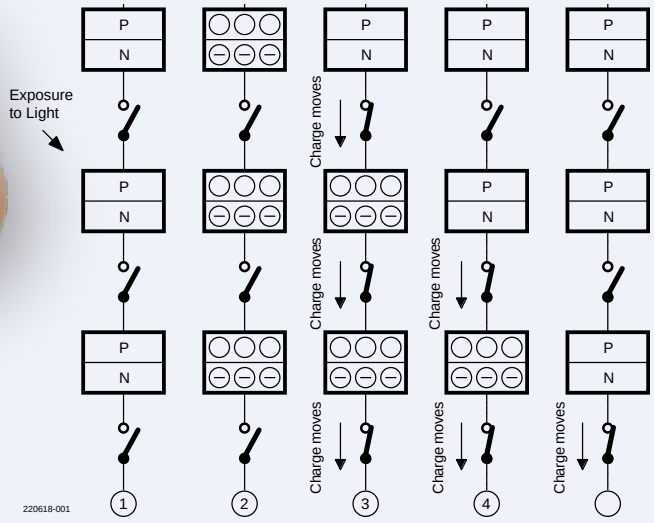


Figure 1. Les capteurs d'images CCD utilisent le transfert de charges entre pixels pour lire leurs valeurs. (Source : Canon Inc.)

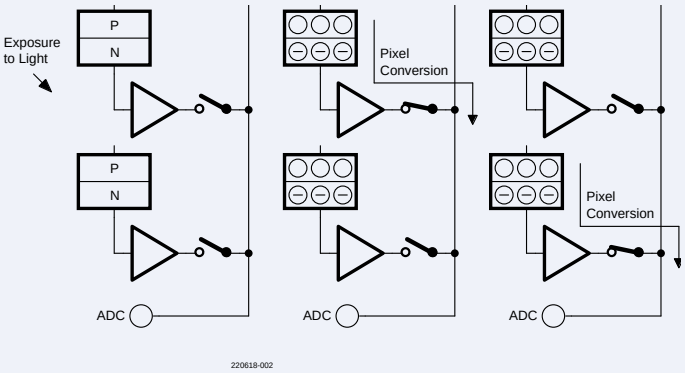
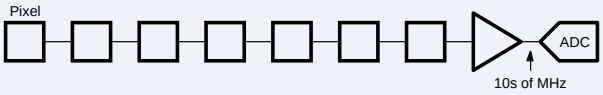


Figure 2. Les capteurs d'images CMOS utilisent un CA/N pour convertir la valeur de chaque pixel. (Source : Canon Inc.)



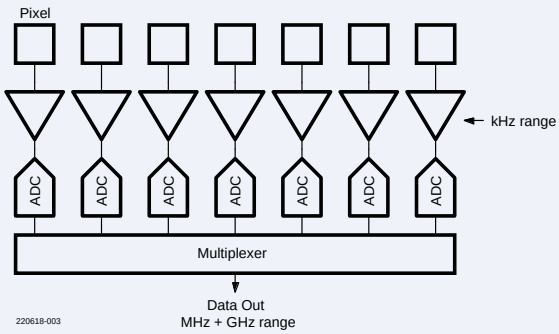


Figure 3. Les capteurs d'images CMOS étant massivement parallèles, le frontal analogique n'a pas besoin d'être aussi rapide que celui utilisé dans les capteurs CCD. (Source : Teledyne Digital Imaging Inc.)

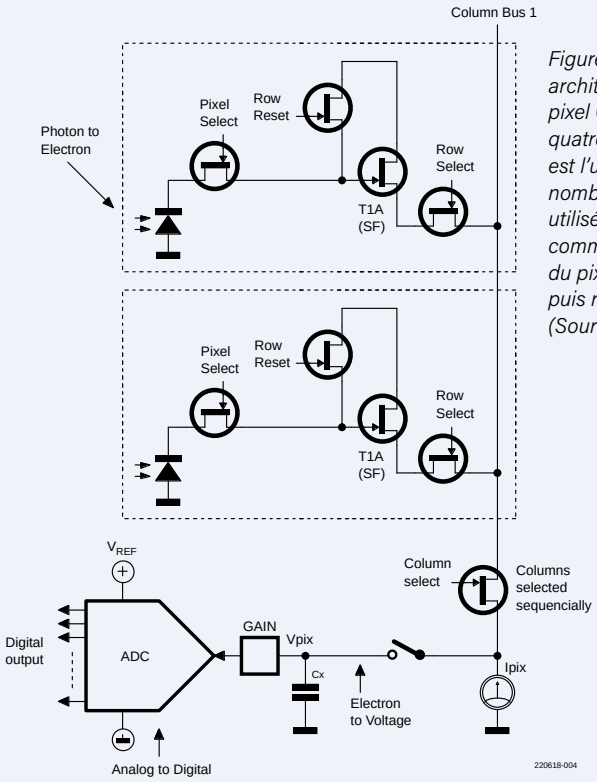


Figure 4. Cette architecture de pixel CMOS à quatre transistors est l'une des nombreuses utilisées et montre comment la charge du pixel est évaluée puis réinitialisée. (Source : OnSemi)

est essentiel pour les appareils portables actuels fonctionnant sur batterie.

La réduction du bruit dans les pixels et l'augmentation de la fréquence d'images sont obtenues par une conversion de données massivement parallèle, de multiples CA/N convertissant simultanément les charges dans une ligne de pixels. Par rapport aux CCD, cela permet d'utiliser des amplificateurs et des CA/N à bande passante plus faible (**figure 3**). La conception des pixels varie mais consiste généralement en un amplificateur à haute impédance connecté à la jonction de la photodiode et en des interrupteurs pour la sélection des lignes et des colonnes. Enfin, un interrupteur est prévu pour réinitialiser le pixel (**figure 4**).

Ajouter de la couleur

Si l'utilisation du populaire procédé de production CMOS a pu réduire le coût des capteurs d'images, ceux-ci sont tout sauf normaux en matière de fabrication. Les photodiodes sont sensibles à toute la lumière visible, de sorte qu'on n'obtiendrait que des niveaux de gris. Pour produire des images en couleur, un réseau de filtres est nécessaire. On utilise généralement un filtre de Bayer, du nom de son inventeur, Bryce Bayer, d'Eastman Kodak. Il utilise des filtres rouges, bleus et deux filtres verts disposés de manière à ce que tout carré de quatre pixels soit composé d'une moitié de vert, d'un quart de rouge et d'un quart de bleu. La part plus élevée du vert est due à la physiologie de l'œil, plus sensible au vert [2].

L'image brute résultante n'est pas adaptée à la consommation humaine immédiate sans prétraitement. Ce traitement est effectué au moyen d'un processus appelé démosaïquage. Dans sa forme la plus simple, chaque pixel est converti en une valeur RVB (rouge/vert/bleu) sur la base des valeurs de pixels spécifiques qui l'entourent. Cependant, certaines approches de démosaïquage, telles que l'interpolation linéaire, peuvent provoquer des artefacts dans l'image finale, notamment autour des bords des objets. Pour lutter contre ce problème, il existe une sélection d'algorithmes qui tentent de minimiser les artefacts tout en fournissant une image finale rapide [3].

Une autre différence par rapport à la fabrication typique d'une puce est la nécessité

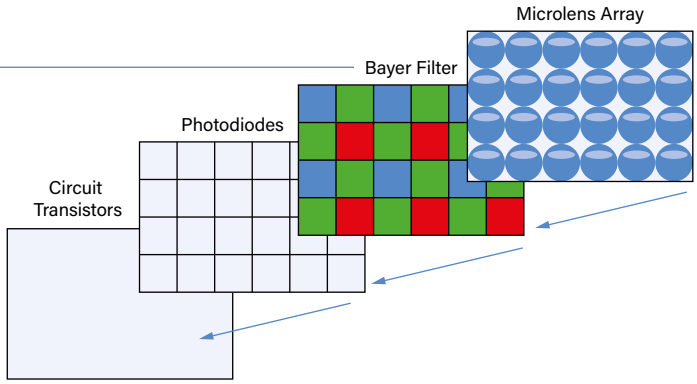


Figure 5. Contrairement aux semi-conducteurs classiques, les capteurs d'images nécessitent un filtre de couleur et un réseau de microlentilles dans leur construction.

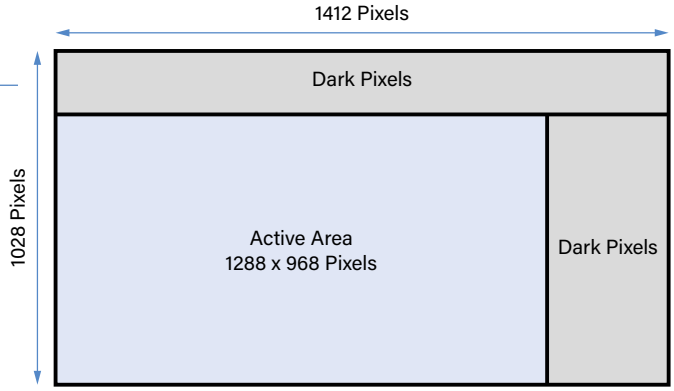


Figure 6. Des pixels supplémentaires peuvent être mis en œuvre dans un capteur d'image CMOS pour assurer la compensation du niveau de noir. (Source : OnSemi)

d'une couche de microlentilles. La surface photosensible de chaque pixel étant très petite, la lentille permet de concentrer la lumière entrante sur le pixel, ce qui augmente l'efficacité quantique (**figure 5**).

Comprendre les obturateurs électroniques

Le fonctionnement de l'obturateur du capteur d'images peut également être difficile à comprendre pour les non-initiés. Un obturateur électronique est utilisé pour exposer les pixels à la lumière pendant la durée correcte. En général, deux options principales sont disponibles, l'obturateur global et l'obturateur roulant, et certains capteurs d'images prennent en charge les deux. Avec un obturateur global, tous les pixels du capteur d'images sont contrôlés électroniquement pour être exposés en même temps. Ensuite, chaque ligne de pixels est convertie en valeurs numériques et transmise au processeur hôte qui gère le traitement de l'image. Avec un obturateur roulant, les rangées de pixels sont exposées à la lumière l'une après l'autre, la conversion s'effectuant immédiatement pour chaque ligne.

La différence dans l'image qui en résulte est partiellement impactée par le type d'obturateur utilisé. Un obturateur global capture l'image instantanément. Par contre, l'obturateur roulant capture chaque ligne avec un léger retard par rapport à la ligne précédente. Lorsque l'objet cible est en mouvement, des artefacts peuvent apparaître dans l'image résultante. Par exemple, si l'on photographie une locomotive se déplaçant de gauche à droite, l'image finale présentera des traînées de gauche à droite en partant du haut de l'image.

Si le choix d'un obturateur global semble s'imposer, il y a, comme toujours, d'autres détails à prendre en compte. La technique de l'obturateur roulant nécessite moins de transistors autour de chaque pixel, ce qui laisse plus de surface pour sa partie photosensible, d'où une meilleure qualité d'image. Cette conception simplifiée permet également de réduire le bruit dans le pixel. Enfin, l'obturateur roulant permet d'atteindre des fréquences d'images plus élevées que l'obturateur global, car les rangées supérieures de l'image suivante peuvent commencer à être exposées pendant que les dernières rangées sont converties.

La décision quant à la meilleure solution dépend de l'application, et les problèmes causés par l'obturateur roulant peuvent être résolus en utilisant un flash synchronisé avec l'obturateur. Certains instruments scientifiques nécessitent une approche globale de l'obturateur, tandis qu'une caméra d'inspection visuelle se contentera d'un obturateur roulant et d'une source de lumière flash synchronisée. Une ressource utile pour comprendre ces questions plus en détail est fournie par PCO AG [4].

Une autre curiosité potentielle avec les capteurs d'images est la différence entre le nombre de pixels sur le capteur et le nombre disponible pour l'application utilisatrice. Par exemple, le OnSemi AR0130CS [5] est un capteur d'image numérique CMOS de ⅓ de pouce et de 1,2 mégapixels avec un obturateur roulant. Sa fiche technique déclare 1280 × 960 pixels actifs alors que le nombre réel est de 1412 × 1028 pixels avec une zone active de 1288 × 968. Certaines des lignes supérieures et un bloc des colonnes du côté droit sont optiquement noirs. Ils sont utilisés en interne pour contrôler et corriger le niveau de noir (**figure 6**).

Un autre défi pour l'acquisition d'une image de bonne qualité réside dans l'alimentation électrique. Le taux de réjection d'alimentation (PSRR) des régulateurs à faible chute (LDO) utilisés doit empêcher tout bruit de se propager dans les broches d'alimentation du capteur d'image CMOS. Dans le cas contraire, le risque de bruit des pixels est accru. Selon une note d'application d'OnSemi [6], lorsque les capteurs d'images dépassent 50 mégapixels, un PSRR supérieur à 90 dB jusqu'à 10 kHz et supérieur à 45 dB entre 1 et 3 MHz est recommandé. La charge dynamique peut également être assez considérable, avec quelques centaines de milliampères de variation de charge pendant les transitions de ligne. Dans ce cas, des gros condensateurs sont essentiels pour obtenir des performances optimales. Enfin, il faut tenir compte du bruit de sortie du LDO (nV/√Hz), car tout ce qui se situe dans la plage de 10 Hz à 1 MHz peut entraîner un bruit supplémentaire dans les pixels, réduisant ainsi leur plage dynamique.

Vision industrielle avec les microcontrôleurs

Avec leurs processeurs 32 bits et un mégaoctet de SRAM ou plus, les grands microcontrôleurs d'aujourd'hui sont bien adaptés aux applications de vision simples et peuvent même intégrer certains algorithmes d'apprentissage machine (AM) utiles pour la détection d'objets. Un projet à prendre en considération est OpenMV [7], une combinaison de matériel et d'environnement de développement logiciel ciblant spécifiquement la vision industrielle. Utilisant la famille de microcontrôleurs STM32, leur matériel donne accès à des capteurs d'images allant jusqu'à 5 mégapixels, remplaçables selon les besoins.

Le développement du logiciel est pris en charge par l'environnement de développement intégré (EDI) OpenMV, qui est adapté aux besoins de la vision industrielle (**figure 7**). Sur le côté gauche se trouve un éditeur pour écrire du code en MicroPython, et un terminal série, affichant la sortie texte du code

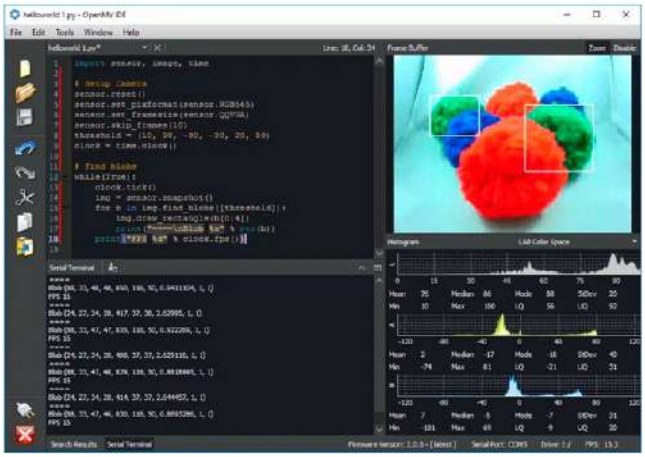


Figure 7. L'environnement de développement OpenMV fournit un éditeur pour le codage et affiche l'image capturée par le capteur associé. (Source : OpenMV)

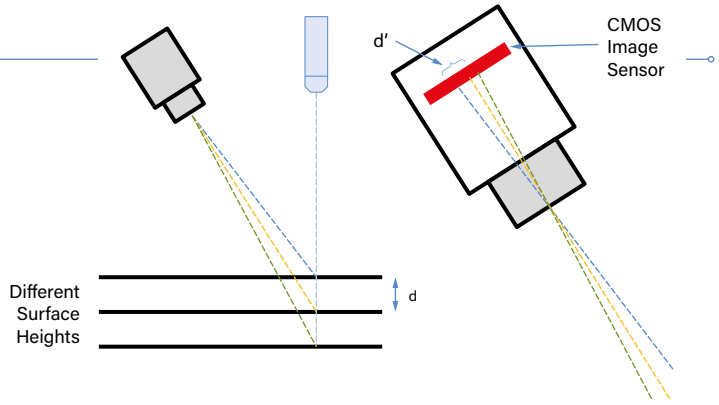


Figure 8. La triangulation laser utilise un capteur d'image CMOS pour détecter la réflexion d'un laser sur une surface afin d'en déterminer la hauteur. (Source : MoviMED)

de l'application. Sur le côté droit se trouve la sortie de la caméra. En haut se trouve le tampon d'image montrant l'image de la caméra combinée à toute sortie injectée dans le flux vidéo par l'application. Il peut s'agir de messages textuels ou de cadres de limitation mettant en évidence les objets détectés en utilisant l'AM. En bas, des histogrammes sont affichés, comme les niveaux de rouge, de bleu et de vert de l'image. Tout comme dans l'environnement Arduino, des exemples de projets sont disponibles. Ils comprennent également des modèles d'AM pour des applications telles que la détection de visages et de personnes.

Imagerie 3D par triangulation laser

Les caméras équipées de capteurs d'images CMOS peuvent être utilisées pour une multitude d'applications de vision, mais qui relèvent généralement de l'imagerie 2D. Cependant, beaucoup d'applications exigent des mesures de surface précises qui requièrent une capacité d'imagerie 3D. Dans le domaine industriel, la triangulation laser est une solution courante utilisée pour mesurer la surface des produits finis tels que les soudures, les cordons de colle et la position des composants sur les circuits imprimés.

Le principe de ces capteurs consiste à projeter un point ou une ligne laser sur l'objet à mesurer. La lumière réfléchie par la surface est reçue par un capteur CMOS qui détermine la distance de l'objet par triangulation (**figure 8**). Le positionnement du laser et du capteur détermine la complexité des calculs nécessaires pour déterminer la distance, la résolution et la probabilité d'erreurs de mesure. Par exemple, le laser peut être perpendiculaire à l'objet (géométrie standard) et la caméra inclinée, ce qui en fait un bon capteur polyvalent. Cependant, pour mesurer des surfaces très réfléchissantes comme le verre et le métal poli, il est préférable d'opter pour une géométrie de type « look-away », avec le capteur d'image et le laser inclinés par rapport à l'objet.

Les fabricants de capteurs tels que SmartRay proposent une large gamme de capteurs 3D basés sur cette technologie. Leurs capteurs ECCO 95.015G [8] sont spécialement conçus pour le verre plat et les surfaces spéculaires et hautement réfléchissantes. Montés à une distance de 23,5 mm de l'objet à mesurer, ils offrent une résolution verticale de 0,42 à 0,54 µm. L'unité fournit ses mesures sous la forme d'une série de points sur une sortie gigabit Ethernet à laquelle peut être connecté un système d'imagerie industrielle standard pour évaluer les données.

Capteurs d'images CMOS pour tous

Grâce à leur faible coût et à leur faible consommation, les capteurs d'images CMOS sont devenus omniprésents dans la société actuelle. Dans les applications de détection, jusqu'ici domaine des capteurs à ultrasons ou à infrarouge, il est aujourd'hui parfaitement concevable de les remplacer par des capteurs d'images CMOS. Cependant, avec leurs millions de pixels et un usage ardu à maîtriser, ils peuvent être difficiles à interfacer avec des microcontrôleurs. Heureusement, avec leur puissance de calcul et leur capacité de mémoire, les microcontrôleurs d'aujourd'hui sont capables de gérer une fonction relativement complexe de détection d'objets par apprentissage machine utilisant des capteurs d'images de taille modeste. Ces dispositifs ne sont pas limités aux images 2D. Les capteurs industriels CMOS utilisant la triangulation laser fournissent des mesures en trois dimensions de précision micrométrique, permettant d'automatiser le contrôle visuel de qualité des pièces.

(220618-04) — VF : Helmut Müller

Des questions, des commentaires ?

Envoyez un courriel à l'auteur (stuart.cording@elektor.com) ou contactez Elektor (redaction@elektor.fr)

LIENS

[1] « Les instruments de Hubble : WFPC2 Wide Field Planetary Camera 2 », Agence spatiale européenne : https://bit.ly/3WBhN1Z

[2] « The Human Eye's Response to Light », Université d'État de l'Iowa : https://bit.ly/3UsmVDU

[3] A. Rajwade, « Color Image Demosaicing «, Indian Institute of Technology Bombay : https://bit.ly/3UneoCg

[4] « Pourquoi toutes ces discussions sur l'obturateur global et l'obturateur roulant ? », PCO AG : https://bit.ly/3h90h4R

[5] Page du produit : AR0130CS Capteur d'image CMOS : https://bit.ly/3hgWq63

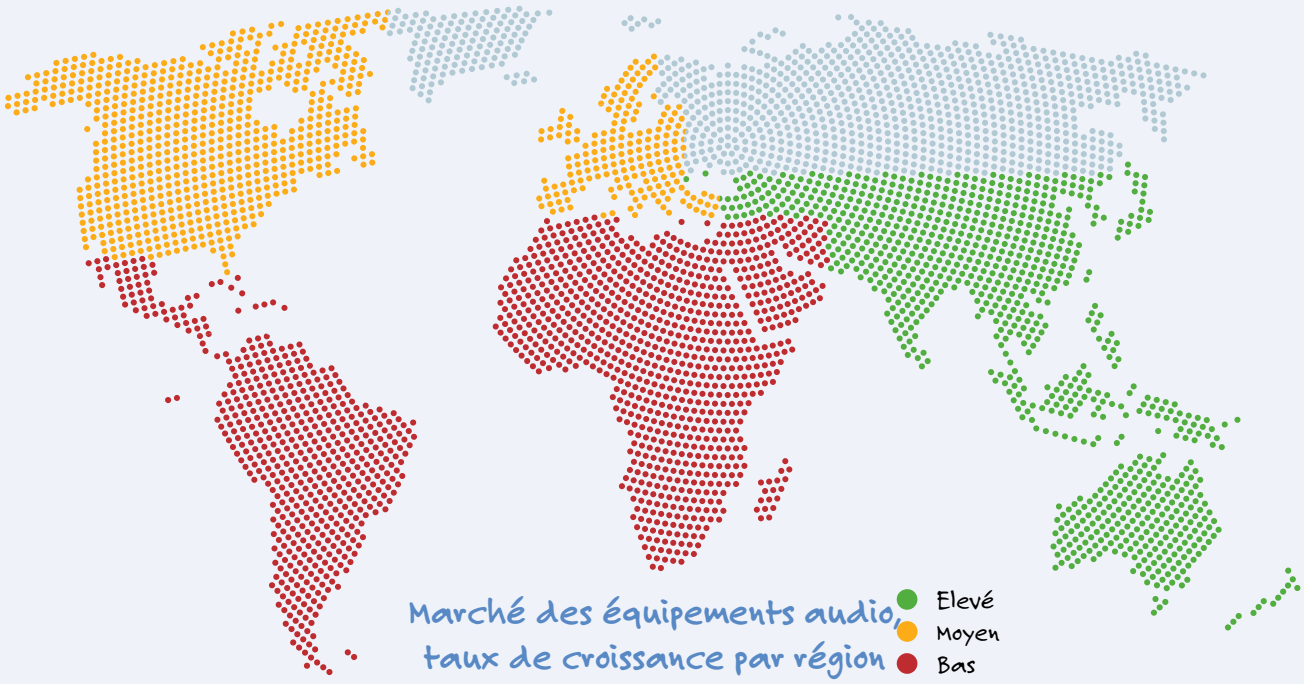
[6] M. Dadafshar, « Understanding Challenges in Powering High Resolution, High Frame Rate CMOS Image Sensors », OnSemi, juillet 2022 : https://bit.ly/3Dz7EtS

[7] Site web : OpenMV : https://openmv.io/

[8] Page produit : ECCO 95.015G : https://bit.ly/3E7HakZ

Elektor Infographies

Robert van der Zwan

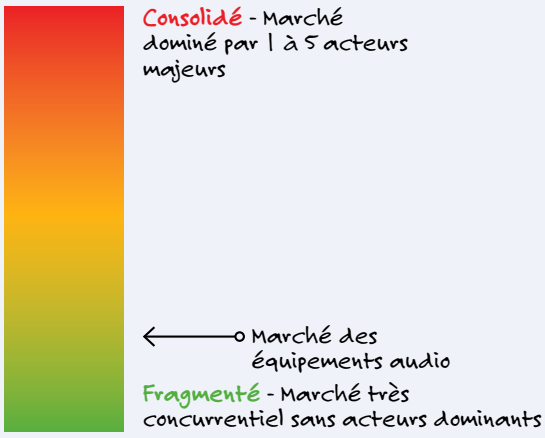


La région Asie-Pacifique prend les devants.

En ce qui concerne les équipements audio dans un environnement professionnel comme les tables de mixage, amplificateurs, microphones et haut-parleurs/moniteurs, les experts prévoient un taux de croissance d'environ 6 % pour les quatre prochaines années. Cela inclut aussi tous les équipements audio dans le secteur automobile.

Le marché passera d'environ 13,5 milliards de dollars US en 2022 à 18 milliards en 2027. L'Asie-Pacifique est la région qui connaît la croissance la plus rapide, suivie par les régions d'Europe et d'Amérique du Nord. La différence de taux de croissance entre l'Asie-Pacifique, l'Europe/Amérique du Nord et l'Afrique/Amérique du Sud s'explique, en grande partie, par un marché automobile en plein essor en Chine et en Inde. Une seule voiture peut contenir jusqu'à 10 haut-parleurs, voire plus.

(Sources : Fortune Business Insights, carte Mordor Intelligence).



Source : Mordor Intelligence (Rapport sur le marché des équipements audio, 2022-2027)

Concentration du marché ? Pas tant que ça

Le marché des équipements audio professionnels est promis à un bel avenir. Par conséquent, les vendeurs ne manquent pas d'intérêt pour explorer ce marché. Les perspectives des acheteurs d'équipements audio s'en trouvent encore améliorées. L'absence relative de concentration du marché signifie que la concurrence par les prix sur le marché de l'équipement audio sera certainement féroce. En outre, de nombreuses innovations sont en cours, qu'elles concernent les amplificateurs, les microphones ou les haut-parleurs. En résumé, le client professionnel peut s'attendre à un niveau d'entrée et de sortie sonores de haute qualité pour des prix raisonnables. De cette manière, l'innovation audio donnera un coup de fouet à la vidéoconférence.

(Sources : Microsoft, Mordor Intelligence)

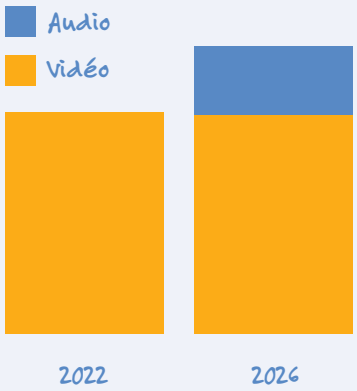
La voiture du futur est...audio en entrée et en sortie

En regroupant quelques chiffres, il devient évident que le marché de l'équipement audio professionnel se compose principalement d'équipements audio dans les automobiles. Veuillez consulter le graphique. Pourquoi la voiture est-elle si répandue ? La première raison est que les conducteurs passent une grande partie de leur temps dans leur voiture, avec des estimations de quelque 45 heures supplémentaires par an à cause des retards dus au trafic. La deuxième raison est que l'utilisation des services d'info-divertissement dans une voiture peut être dangereuse : ils peuvent distraire le conducteur et l'empêcher de veiller à ce qui se passe sur la route. C'est pourquoi la reconnaissance vocale est si importante. La reconnaissance vocale nécessite des processeurs audio, qui font également partie du marché des équipements audio.



(Sources : Fortune Business Insights, Mordor Intelligence).

Une vue d'ensemble : la vidéo l'emporte sur l'audio



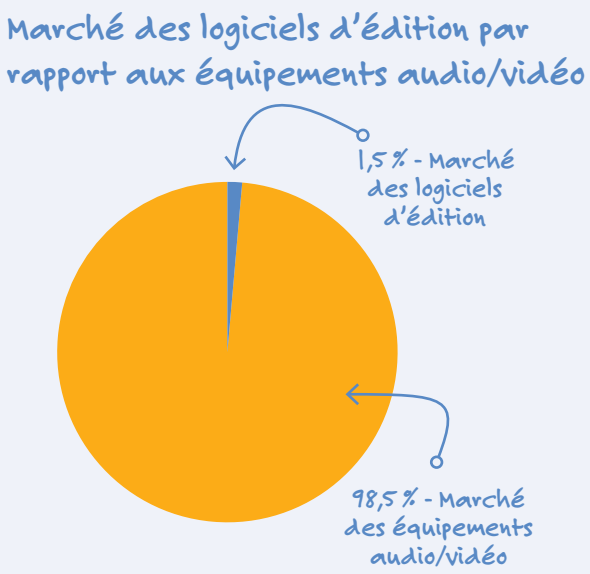
Il y a une différence entre le marché audio professionnel et le marché audio global. Ce dernier comprend les utilisateurs privés. Le marché global de l'audio concerne toutes sortes d'enregistreurs et de lecteurs, et pas seulement les mixeurs professionnels, etc. Il en va de même pour les équipements vidéo : une caméra de télévision est différente d'une caméra de smartphone. Cependant, si nous regroupons l'utilisation professionnelle et l'utilisation privée pour l'audio et la vidéo, que disent les chiffres ? Eh bien, ce marché valait environ 158 milliards de dollars américains l'année dernière, avec une croissance prévue à 196 milliards en 2026. Environ 80 % du marché global est constitué d'équipements vidéo. Les téléviseurs incurvés seront les gagnants au cours des prochaines années.

(Sources : Market Prospects, Research and Markets).

Le matériel ? D'accord, **mais qu'en est-il des logiciels ?**

Jusqu'à présent, nous avons parlé de matériel. Qu'en est-il des logiciels ? Il existe, bien sûr, un marché pour les logiciels de montage audio et vidéo, ne serait-ce qu'à cause des YouTubers. Le marché des logiciels croît à un rythme encore plus élevé que le marché des équipements audio et vidéo : 7 % au lieu d'environ 6 % pour les années à venir. Le marché des logiciels audio et vidéo n'est qu'une goutte d'eau dans l'océan cependant, comparé au matériel audio et vidéo. D'après les chiffres de Transparency Market Research, le marché des logiciels de montage audio et vidéo passera d'environ 2 milliards de dollars US en 2020 à environ 3 milliards en 2030. Cela signifie que le marché des logiciels audio/vidéo représente environ 1,5 % du marché du matériel informatique.

(Sources : Research and Markets, Transparency Market Research).





▲ *Figure 1. Robot assistant à commande vocale. (Source : PaO_STUDIO sur Shutterstock)*

l'évolution de la **commande vocale et audio** des appareils électroniques

Mark Patrick (Mouser Electronics)

La parole est un moyen efficace pour les gens d'exprimer des idées et des désirs. Bien avant l'ère industrielle, les humains avaient déjà découvert que les animaux pouvaient être entraînés à reconnaître et à répondre à des ordres de base pour accomplir une tâche.

Il est donc apparu comme logique de développer un moyen de communiquer par la voix avec nos machines pour les commander. L'utilisation de la voix et de l'audio comme interface de commande des appareils électroniques a gagné en popularité ces dernières années et cette méthode continue d'évoluer afin de répondre aux attentes des utilisateurs et aux exigences des nouvelles applications.

Dans cet article, nous décrirons quels sont les avantages de la commande vocale et audio des équipements et des machines électroniques ainsi que la façon dont cette technologie est mise en œuvre. Nous verrons également comment cette interface de commande peut désormais être intégrée dans des appareils hors ligne et quelles sont les possibilités d'amélioration de l'expérience audio fournie par ces appareils.

Utilisation de la voix pour commander les appareils électroniques

L'interaction voix-machine présente quelques avantages évidents :

- Pour les humains, la parole est une forme de communication intuitive qui permet de transmettre facilement des commandes par voie verbale.
- Il est toujours possible de communiquer verbalement, même lorsque les yeux ou les mains de la personne sont occupés par d'autres choses. La commande vocale simultanée s'avère dès lors très pratique dans certaines circonstances, notamment au volant, où il est illégal de commander un appareil à commande tactile tout en conduisant.
- La voix est un moyen efficace de commander des machines capables d'écouter et de répondre sans nécessiter de commandes complexes.
- L'intégration vocale permet dans de nombreux cas de se dispenser d'un écran tactile. Cela s'avère tout particulièrement utile pour commander des équipements à distance ou dans le cas d'appareils portatifs alimentés par batterie, dont la conception est souvent soumise aux contraintes de la réduction de la taille et de la consommation d'énergie. De plus, la commande vocale est manifestement plus hygiénique que la commande tactile pour les appareils commandés par plusieurs utilisateurs.
- Comme le montre la **figure 1**, cet avantage peut être utilisé dans des applications destinées à assister des personnes handicapées se trouvant dans l'incapacité de toucher un écran. La commande vocale permet dès lors de donner des ordres à la machine – par exemple, ouvrir une porte – ou de communiquer à distance des informations sur l'état de santé d'une personne.

Le système audio frontal (AFE) d'un dispositif à commande vocale comprend un réseau de microphones et des blocs de traitement de signal. L'AFE traite le signal d'un réseau de microphones multicanaux pour annuler les interférences provenant des bruits de fond ou de la lecture de l'appareil lui-même. Ce signal est ensuite envoyé à un moteur de détection de « mot de réveil », lequel a pour fonction de reconnaître les mots préprogrammés qui activent l'appareil (par exemple « Alexa » ou « OK Google » pour les plus connus). Plusieurs algorithmes de traitement du signal sont utilisés pour annuler les signaux d'interférence indésirables. Voici les principaux composants d'une solution à commande vocale.

Un réseau de microphones. Les systèmes à activation vocale nécessitent un ou plusieurs microphones pour capter le signal de commande audio. Les principaux critères dictant le choix d'un réseau de microphones sont sa taille, son prix, ses performances et sa robustesse. Le fait de combiner différents signaux provenant d'un réseau à multiples microphones permet d'améliorer le rapport signal sur bruit (SNR) de la chaîne de signal audio.

Un détecteur de direction d'arrivée (DoA) permet de déterminer la position de l'utilisateur par rapport à l'appareil contrôlé afin que le réseau de microphones oriente son faisceau en direction de la voix.
Un formeur de faisceau (« beamformer ») accepte les sons provenant de la direction d'arrivée et rejette ceux provenant d'autres directions. Son efficacité dépend de la géométrie et du SNR du réseau de microphones, de la largeur du faisceau et du niveau de bruit de fond.

Un compensateur d'écho acoustique (AEC) rejette le signal émis depuis le haut-parleur de l'appareil afin de capter plus clairement la commande vocale de l'utilisateur, par exemple lorsque celui-ci émet une commande vocale pendant que l'appareil diffuse de la musique.

Un éliminateur d'interférences adaptatif (AIC) supprime les bruits de sources externes qu'un formeur de faisceau traditionnel ne parviendrait pas à supprimer entièrement. Il peut s'agir, par exemple, d'un bruit fort produit par un autre équipement.

Un détecteur de mots de réveil (ou mots d'activation). Un algorithme de détection de mots de réveil compare le signal vocal traité par l'AFE aux entrées d'une bibliothèque de mots de réveil comme « Hé Google ». L'algorithme repose sur un modèle d'apprentissage automatique. Plus le modèle est grand, plus il est précis. Par exemple, un modèle entraîné de 1 Mo sera plus précis qu'un modèle de 64 Ko. En revanche, il sollicitera davantage le processeur. Quand il s'agit de mots de réveil, il faut disposer de grands modèles pour assurer une détection plus précise des mots de réveil et réduire le nombre de faux positifs.

Les amplificateurs audio de classe D

L'aspect traitement de la voix de ce type d'interface de commande a connu de nombreuses améliorations, de sorte que même des appareils à faible coût peuvent être dotés d'une fonction de reconnaissance vocale précise. En revanche, l'aspect audio a longtemps été délaissé et la plupart des enceintes intelligentes et autres appareils audio connectés plus anciens offrent une qualité audio médiocre par rapport à des équipements audio haute fidélité.

Il est possible que les constructeurs aient considéré que l'engouement pour la commande vocale allait suffire à

mesure du THD avec un oscilloscope et une FFT

calculer facilement le facteur de distorsion

Sebastian Westerhold (AI5GW) (Allemagne)

La distorsion harmonique totale, ou THD en abrégé, est une mesure importante du contenu harmonique des signaux dans les réseaux d'alimentation et les assemblages électroniques, tels que les amplificateurs audio. En général, des appareils de mesure spéciaux sont nécessaires pour la déterminer. Toutefois, si vous disposez d'un oscilloscope numérique doté de la fonction FFT, vous pouvez la calculer avec un peu de mathématiques.

Tout d'abord, un peu de terminologie : une onde est constituée d'une fréquence fondamentale et des harmoniques. Les harmoniques sont des multiples entiers de la fréquence fondamentale. Le premier harmonique est le double de la fréquence de l'onde fondamentale. La 3ème harmonique = 4 fois la fréquence de l'onde fondamentale.

Il existe deux représentants importants de la famille THD, la THD_R et la THD_F. La règle suivante s'applique aux deux : plus la THD est petite, plus le contenu harmonique du signal mesuré est faible. Plus le contenu harmonique est important, plus la THD est élevée. Les harmoniques étant généralement un effet de la distorsion, il s'agit essentiellement d'une mesure permettant de quantifier la distorsion non linéaire.

La THD_F désigne le rapport entre tous les harmoniques et l'onde fondamentale. Cette quantité est utilisée, par exemple, dans la mesure des harmoniques dans les réseaux d'alimentation électrique. La THD_R, quant à elle, désigne le rapport de tous les harmoniques sur le signal total. Cette quantité est particulièrement populaire dans l'ingénierie audio et est également connue sous le nom de « facteur de distorsion ».

La distinction entre les deux métriques est très importante, car la THD_F et la THD_R, bien qu'étroitement liées, produisent des résultats différents. Dans la littérature et les fiches techniques, les deux valeurs sont souvent confondues l'une avec l'autre [1]. Dans certains endroits, vous pouvez trouver uniquement le terme « THD » sans autre définition de la méthode utilisée, par exemple, dans la fiche technique de Texas Instruments du LM386.

En pratique, la THD_F peut être calculée facilement à partir de la somme des différences de niveau de puissance de tous les harmoniques par rapport à la fondamentale. Avec cela, on peut calculer la THD_R.

$$THD_F = 100 \cdot \sqrt{\sum_{n=2}^{\infty} 10^{\left(\frac{P_n}{10}\right)}} = 100 \cdot \sqrt{10^{\left(\frac{P_2}{10}\right)} + 10^{\left(\frac{P_3}{10}\right)} + 10^{\left(\frac{P_4}{10}\right)} \ldots}$$

Formule 1 : THD_F (en %), n = numéro de l'harmonique de tableau 1, P = différence de niveau de puissance par rapport à l'onde fondamentale en dBc.

La formule décrivant ce principe peut sembler déconcertante à première vue. En termes simplifiés, la THD_F est la racine carrée de la somme de tous les rapports de tension calculés à partir des différences de niveau de puissance relative d'un nombre infini d'harmoniques par rapport à l'onde fondamentale. Cette phrase peut sembler tout aussi déconcertante que la formule, mais elle n'est pas aussi complexe qu'elle n'y paraît.

La valeur de la THD_F calculée de cette manière peut ensuite être convertie en facteur de distorsion à l'aide de la formule suivante :

$$THD_R = \frac{THD_F}{\sqrt{1 - THD_F^2}}$$

Formule 2 : THD_R (en %), THD_F (en %).

La différence entre la THD_F et la THD_R diminue à mesure que les valeurs elles-mêmes sont plus petites. En dessous de 1 %, la différence est presque négligeable.

Dans les réseaux d'alimentation électrique, pour des raisons évidentes, la fréquence du secteur, par exemple 50 Hz, est utilisée comme fréquence fondamentale lors du calcul de la THD. Dans la technologie audio, 1 kHz est une norme courante.

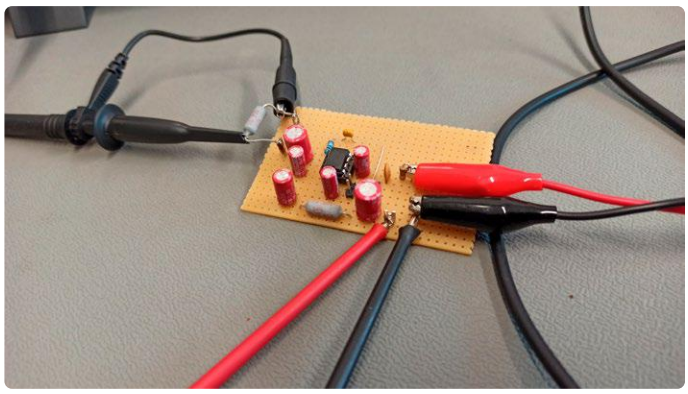


Figure 1. Amplificateur audio avec un LM386N-1 comme cobaye.

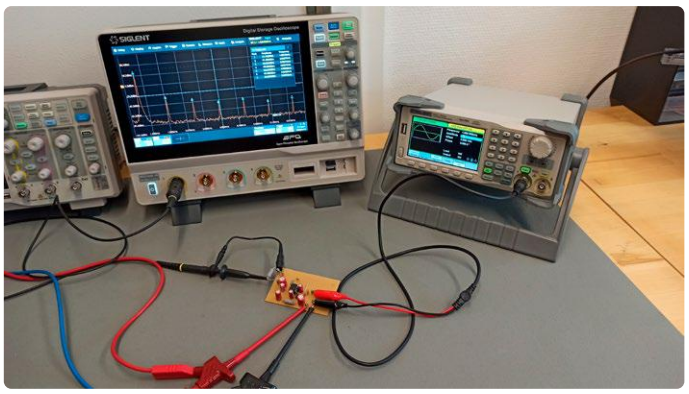


Figure 2. Configuration de mesure complète avec générateur de signaux, oscilloscope et circuit de test.

En pratique, il est bien sûr impossible de prendre en compte un nombre infini d'harmoniques. Par exemple, 50 harmoniques sont le maximum considéré dans les réseaux d'alimentation électrique. En technologie audio, il est logique de négliger ceux non audibles.

Exemple pratique

Pour montrer que la terne théorie mathématique n'est pas si mauvaise en pratique, considérons l'exemple suivant : un signal sinusoïdal de fréquence 1 kHz est envoyé à l'entrée d'un amplificateur (LM386N-1) (**figure 1**). Le signal de sortie de l'amplificateur est connecté à une résistance de 10 Ω (servant de substitut approximatif à l'impédance d'un haut-parleur) et acheminé par une sonde 1:10 à un oscilloscope HD Siglent SDS2104X (**figure 2**).

Vu dans le domaine temporel, le signal de sortie semble assez propre et sinusoïdal (**figure 3**). Cependant, cette première impression est trompeuse, comme le montre un coup d'oeil sur le spectre FFT : les harmoniques, ici extraites jusqu'à 7 kHz, sont clairement visibles à côté de la fondamentale à 1 kHz (**figure 4**).

Pour des raisons de clarté, j'ai écarté mon idée initiale d'approcher l'infini dans le nombre d'harmoniques considérés, au moins jusqu'à la limite de la bande passante de l'oscilloscope. J'ai fini par enregistrer les niveaux de puissance des sept premiers composants, correspondant au niveau de puissance de l'onde fondamentale et des six premiers harmoniques.

La plupart des oscilloscopes offrent la possibilité d'afficher automatiquement les niveaux de puissance mesurés dans un tableau pratique. Ils sont généralement affichés dans l'unité dBm. Il s'agit de dix fois le logarithme décimal du rapport de la valeur mesurée par rapport à une quantité de référence. Dans le cas du dBm, la quantité de référence est 1 mW.

Avant de pouvoir effectuer des calculs, les niveaux de puissance absolue doivent être convertis en une différence de niveau de signal par rapport à l'onde fondamentale. L'unité dBc (dB par rapport à la porteuse) est utilisée ici. Pour la conversion, le niveau de puissance de l'onde fondamentale doit être soustrait des niveaux de puissance de tous les harmoniques. Par exemple, pour le premier harmonique : -31,7 dBm - 22,1 dBm = -53,8 dBc.

Le **tableau 1** résume les harmoniques avec leurs numéros d'ordre, les niveaux de puissance absolue en dBm (arrondis) et les différences



Figure 3. Signal de sortie du circuit de test dans le domaine temporel.

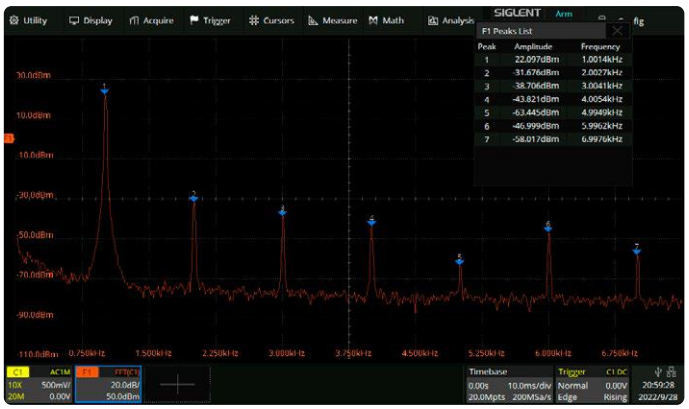


Figure 4. Spectre FFT du signal de sortie.

Tableau 1. Amplitudes absolues et relatives des six premiers harmoniques (arrondies).

Numéro d'ordre de l'harmonique	Niveau de puissance (dBm)	Niveau de puissance relative (dBc)
1. (1 kHz) = fondamentale	22,1	0
2. (2 kHz) = 1er harmonique	-31,7	-53,8
3. (3 kHz) = 2ème harmonique	-38,7	-60,8
4. (4 kHz) = 3ème harmonique	-43,8	-65,9
5. (5 kHz) = 4ème harmonique	-63,5	-85,6
6. (6 kHz) = 5ème harmonique	-47,0	-69,1
7. (7 kHz) = 6ème harmonique	-58,0	-80,1

de niveau de puissance calculées par rapport à la fondamentale. Les niveaux de puissance relatifs du **tableau 1** peuvent être directement insérés dans la formule 1.

$$100 \cdot \sqrt{10^{\left(\frac{-53.8}{10}\right)} + 10^{\left(\frac{-60.8}{10}\right)} + 10^{\left(\frac{-65.9}{10}\right)} + 10^{\left(\frac{-85.6}{10}\right)} + 10^{\left(\frac{-69.1}{10}\right)} + 10^{\left(\frac{-80.1}{10}\right)}} \approx 0.232\,\%$$

La valeur THD_F de l'amplificateur (considérée jusqu'à 7 kHz) est donc d'environ 0,232 %. Avec la deuxième formule, la THD_R (le facteur de distorsion) peut alors également être calculée.

$$THD_R = \frac{0.232}{\sqrt{1-0.232^2}} \approx 0.239\,\%$$

Le facteur de distorsion est donc d'environ 0,239 %. La fiche technique du LM386 spécifie une THD de 0,2 %. Ceci est assez proche des valeurs calculées ici. Malheureusement, Texas Instruments ne révèle pas le type de THD indiqué dans la fiche technique, ni le nombre d'harmoniques inclus dans la mesure.

Limites et notes

La méthode présentée ici trouve ses limites dans la plage dynamique du convertisseur analogique-numérique (CA/N) de l'oscilloscope utilisé. En général, il s'agit d'un 8 bits avec une gamme dynamique théorique d'environ 48 dB. Ils peuvent être utilisés jusqu'à une THD minimum d'environ 3 %.

Les lecteurs attentifs auront peut-être remarqué que la THD calculée ci-dessus à titre d'exemple est nettement inférieure à cette limite. Cela est possible car j'ai utilisé un oscilloscope à 12 bits, qui a une gamme dynamique théorique d'environ 72 dB. Cela signifie que des valeurs de THD d'environ 0,2 % sont dans le domaine du possible.

Toutefois, pour tirer le meilleur parti de ces limites, la prudence est de mise : la plage dynamique réellement disponible dépend également de manière significative de la déviation verticale. Si le CA/N est sur-dirigé, des distorsions se produisent ; s'il est sous-dirigé, une gamme dynamique précieuse est perdue.

Si vous voulez essayer cette méthode directement, vous pouvez vous demander où obtenir des signaux de référence avec des valeurs de THD connues. Un simple générateur de signaux peut vous aider : un signal carré raisonnablement symétrique avec un rapport cyclique de 50 % a un THD_F d'environ 48,3 %. Avec un signal triangulaire également symétrique avec le même rapport cyclique, le THD_F est encore d'environ 12,1 %.

Si vous souhaitez approfondir la question et comprendre la dérivation des formules, vous trouverez de plus amples informations sous [2] et [3].

220398-04 — VF : Maxime Valens

Des questions, des commentaires ?

Envoyez un courriel à l'auteur (sebastian@baltic-lab.com) ou contactez Elektor à (redaction@elektor.fr).

Produits

- **Siglent SDS1204X-E oscilloscope 4 voies (200 MHz) (SKU 18524)**
  www.elektor.fr/18524
- **Siglent SDG1032X générateur de fonctions arbitraires double canal (30 MHz) (SKU 20276)**
  www.elektor.fr/20276
- **Adafruit 2.5 W ampli mono classe D (PAM8302) (SKU 18745)**
  www.elektor.fr/18745

LIENS

[1] Doron Shmilovitz, "On the definition of total harmonic distortion and its effect on measurement interpretation," Power Delivery, IEEE Transactions, S. 526 – 528, 2005: https://doi.org/10.1109/TPWRD.2004.839744
[2] Sebastian Westerhold, "Total Harmonic Distortion (THD) from dBc," 2022: https://baltic-lab.com/thd/
[3] Sebastian Westerhold, "Total Harmonic Distortion (THD) analysis utilizing the FFT capabilities of modern digital storage oscilloscopes," 2022: http://dx.doi.org/10.5281/zenodo.6969825

embeddedworld2023

Exhibition&Conference

... it´s a smarter world

UNITED

JOIN THE EMBEDDED COMMUNITY

14–16.3.2023

Get your free ticket now!

embedded-world.com/voucher

Use the voucher code **GG4ew23**

Media partners

Markt&Technik

Elektronik

SmarterWorld

DESIGN& ELEKTRONIK

Elektronik automotive

medical-design

computer & automation

elektroniknet.de

NÜRNBERG MESSE

faire oublier le défaut de qualité audio. Or, les consommateurs se sont aussi montrés plus exigeants sur le plan de l'expérience audio à mesure que ce type d'appareils intelligents se démocratisait. Les amplificateurs audio traditionnels de classe AB sont trop peu efficaces pour être utilisés dans des appareils connectés de faible puissance. Cependant, plusieurs fabricants de puces ont récemment mis sur le marché une gamme d'amplificateurs audio de classe D de haute technologie qui constituent une amélioration considérable par rapport aux générations d'amplificateurs précédentes. Nombre de ces nouveaux amplificateurs ont été spécialement conçus pour permettre à des appareils intelligents ou connectés de délivrer un son de qualité supérieure.

Le TAS2770 [1] de Texas Instruments [2] est un amplificateur audio à puissance d'entrée de 15 W qui améliore la qualité audio. Il est en outre doté d'une excellente capacité à capturer la voix, ce qui permet de commander plus facilement et plus naturellement les appareils à commande vocale qui en sont équipés. C'est le premier système audio frontal à combiner une entrée de microphone numérique avec un puissant amplificateur de détection I/V, ce qui lui permet de capturer la voix et les bruits ambiants pour l'annulation d'écho ou la réduction du bruit dans les applications vocales. Maxim Integrated [3] (qui fait désormais partie du groupe Analog Devices [4]) propose pour sa part les MAX98357 [5] et MAX98358 [6], des amplificateurs de classe D avec un rendement de 92 % et qui délivrent une puissance de 3,2 W de classe AB. On trouvera un schéma fonctionnel simplifié de ces amplificateurs en **figure 2**. Économe en énergie, le PAM8106 [7] de Diodes Incorporated [8] est particulièrement adapté aux appareils alimentés par des batteries plomb-acide de 1,5 V ou lithium-ion de 3,5 V. Il affiche un rendement de 92 % et peut être intégré dans une architecture compacte en se passant d'un encombrant dissipateur thermique.



Figure 3. Solution de commande vocale hors ligne SLN-LOCAL2-IOT par NXP. (Source : NXP [10])

Commande vocale hors ligne

Des solutions basées sur le cloud comme Alexa d'Amazon et Google Assistant sont simples à mettre en œuvre sur des appareils qui disposent d'une connexion Internet stable, mais lorsque la connexion est faible ou inexistante, on leur préférera une solution de commande vocale hors ligne. Prenons par exemple un produit qui ne doit répondre qu'à des commandes simples d'un seul mot comme « marche », « arrêt », « réinitialisation », etc. – ce qu'on appelle couramment le repérage de mots-clés – il paraît plus logique

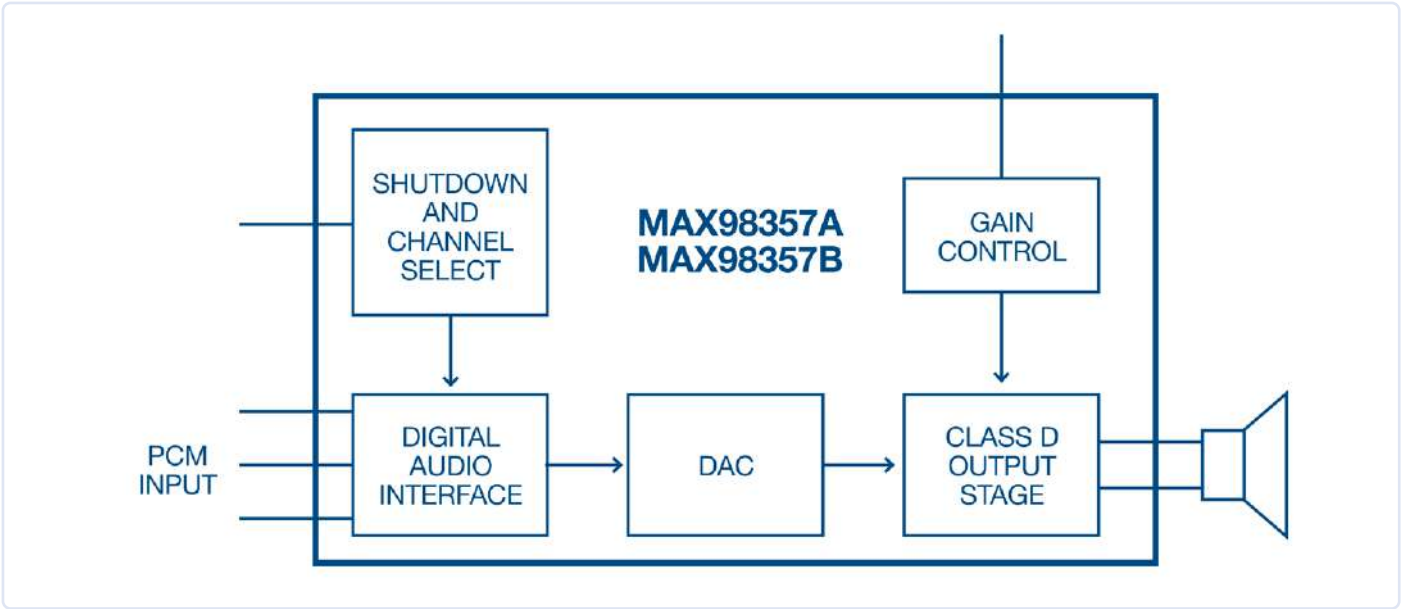


Figure 2. Schéma fonctionnel simplifié des amplificateurs audio de classe D de Maxim Integrated. (Source : Maxim Integrated)

que le traitement s'effectue localement, c'est-à-dire sur l'appareil même. Il est en effet possible de doter l'appareil d'un système de commande par mot-clé de base en y intégrant un microcontrôleur à faible coût. Avec EdgeReady, NXP propose une solution de commande vocale traitée localement et hors ligne reposant sur le microcontrôleur Crossover i.MX RT qui permet aux développeurs d'intégrer rapidement le contrôle vocal à leurs produits. La solution basée sur le microcontrôleur i.MX RT106S inclut le kit de développement SLN-LOCAL2-IOT [9] (voir **figure 3**).

Cette solution s'accompagne d'un logiciel entièrement intégré sous FreeRTOS livré avec un kit de développement logiciel (SDK) afin de réaliser rapidement la preuve de concept. Elle permet aux fabricants d'intégrer facilement de la fonctionnalité de commande vocale à tous leurs produits intelligents (appareils, maisons, bâtiments et produits industriels intelligents) sans qu'il soit besoin de disposer d'une connexion wifi ou au cloud. La commande vocale hors ligne répond ainsi en grande partie aux craintes de nombreux consommateurs concernant le piratage de leurs données accessibles en ligne. Le kit comprend enfin un outil compatible Windows™ servant à créer, à partir de la saisie de texte dans plus de quarante langues, des modèles vocaux pour plus d'une centaine de commandes personnalisées et plusieurs mots de réveil.

Conclusion

La voix et l'audio sont rapidement devenus les interfaces privilégiées dans de nombreux appareils intelligents. Elles remplacent avantageusement les affichages numériques – coûteux et énergivores – pour les appareils connectés portatifs et à faible consommation d'énergie. Les premiers systèmes de ce type souffraient pour la plupart d'une qualité audio médiocre et nécessitaient une connexion au cloud.

Mais une nouvelle génération d'amplificateurs audio de classe D à haut rendement permet désormais aux fabricants de proposer des équipements qui offrent aux consommateurs une expérience audio de haute qualité. Par ailleurs, il existe aujourd'hui des solutions qui permettent de commander des appareils sans connexion Internet ou avec une connexion restreinte. Ces innovations démontrent que cette technologie est capable de s'adapter aux nouvelles attentes des consommateurs qui apparaissent à mesure qu'ils s'habituent à l'utilisation de cette interface de commande. C'est là une tendance qui devrait se poursuivre.

220620-04



À propos de l'auteur

En tant que directeur marketing technique de Mouser Electronics pour la région EMEA, Mark Patrick est responsable de la création et de la diffusion du contenu technique - un contenu essentiel à la stratégie de Mouser visant à soutenir, informer et inspirer son public d'ingénieurs.

Avant de diriger l'équipe de marketing technique, Patrick faisait partie de l'équipe de marketing achat de la région EMEA et jouait un rôle essentiel dans l'établissement et le développement des relations avec les principaux partenaires et fournisseurs. En plus d'avoir occupé divers postes dans les départements techniques et marketing, Patrick a travaillé pendant huit ans chez Texas Instruments, dans les services support et ventes techniques.

Ingénieur expérimenté, passionné de synthétiseurs vintage et de motos, il n'hésite pas à les réparer. Patrick est titulaire d'un diplôme d'ingénieur en électronique avec mention très bien de l'université de Coventry.

LIENS

[1] TAS2770 Audio Amplifier Evaluation Module : http://elektor.link/MouserTAS2770
[2] Texas Instruments @ Mouser : http://elektor.link/MouserTexasInstruments
[3] Produits présentés par Maxim Integrated : http://elektor.link/MouserMaxim
[4] Analog Devices @ Mouser : http://elektor.link/MouserAnalogDevices
[5] MAX98357 : http://elektor.link/MouserMAX98357
[6] MAX98358 : http://elektor.link/MouserMAX98358
[7] PAM8106 amplificateur audio 10 W : http://elektor.link/MouserPAM8106
[8] Diodes Incorporated @ Mouser : http://elektor.link/MouserDiodesIncorporated
[9] SLN-LOCAL2-IOT Solution for Local Voice Control : http://elektor.link/MouserSLNLOCAL2IOTDevkit
[10] SLN-LOCAL2-IOT Product : http://elektor.link/MouserSLNLOCAL2IOT

bilan du WEEF 2022

C. J. Abate (Elektor)

Le mouvement éthique dans l'électronique était au centre des préoccupations lors du deuxième Forum mondial de l'électronique éthique de la Fondation Lenthe, qui s'est tenu à electronica 2022 à Munich, en Allemagne.



Le deuxième Forum mondial de l'électronique éthique s'est déroulé au salon electronica Munich (Source : Messe München)

Le 15 novembre 2022, au salon electronica de Munich, le Forum mondial de l'électronique éthique (WEEF) a rassemblé des ingénieurs, des cadres de l'industrie électronique, des universitaires et des journalistes spécialisés dans les technologies de l'information, pour discuter des enjeux éthiques les plus importants liés à l'électronique de notre époque. Au cours de cet événement d'une journée, la Fondation Lenthe, un producteur d'événements sur l'entrepreneuriat social à but non lucratif basé à Amsterdam, ainsi que ELEKTRONIKPRAXIS et electronica, ses partenaires médiatiques, ont offert une plateforme aux leaders avant-gardistes pour discuter des sujets suivants : les pratiques d'écoconception, l'importance des objectifs de développement durable, la planète et les personnes avant le profit, les pratiques de ressources humaines éthiques, les ressources durables et bien plus encore.

L'événement était l'occasion de discussions stimulantes et des réflexions de la part d'un large éventail de leaders d'opinion : Stefan Heinemann (professeur, Université des sciences appliquées FOM), Reinhard Pfeiffer (PDG, Centre des expositions à Munich). Florian Kurz (KK Ventures), Michel Loriot (AGIGA), Tessa Quandt (VARTA), Dr. Ralf Bodelier (auteur, journaliste), Hendrik-Jan Overmeer (membre de l'équipe fondatrice, Deedmob), Andrea Barrett (VP responsabilité sociale et développement durable, RS Group), Robert van Kats (Co-Fonda-



Le prix du Forum mondial de l'électronique éthique 2022 (Source : Messe München)



Milda Pladaitė, lauréate du prix WEEF, décerné par Beatriz Sousa (Elektor). (Source : Messe München)



Le lauréat Frank Stührenberg (PDG de Phoenix Contact), avec le Dr Reinhard Pfeiffer (PDG de Messe München) et Don Akkermans (Fondation Lenthe/Président du comité de programme). (Source : Messe München)

teur, bkvv architects), Milda Pladaitė (Initiateur de Engineers Enterprise), Kelly Heaton (artiste, É.-U.), Johann Wiesböck (Rédacteur en chef, Elektronikpraxis), C.J. Abate (directeur du contenu, Elektor), Udo Bormann (directeur du projet *electronica fast forward 2022 Startup Award*), Stuart Cording (journaliste, Elektor) et Don Akkermans (président, WEEF). En plus des conférences, le jury du WEEF a honoré quatre lauréats du prix WEEF :

- **Sven Krumpel** : le propriétaire de CODICO, qui met à la disposition de ses employés des zones de culture pour les fruits et légumes.
- **Gopal Kumar Mohoto :** un ingénieur travaillant sur l'électrification des transports au Bangladesh, il est le cofondateur et le directeur technique d'Advanced Dynamics.
- **Frank Stührenberg :** PDG de Phoenix Contact, qui est membre du conseil d'administration de la Fondation KlimaWirtschaft et l'un des principaux contributeurs de l'All Electric Society.
- **Milda Pladaitė :** une ingénieure civile lituanienne, faisant partie du comité des futurs leaders des jeunes ingénieurs de la Fédération mondiale des organisations d'ingénieurs (FMOI).

L'intégralité de l'événement a été filmée par Elektor, à suivre sur la chaîne Elektor TV à *elektor.tv* !



Vous êtes curieux de connaître l'avenir du WEEF ? Visitez le site *worldethicalelectronicsforum.com* pour obtenir des informations actualisées sur le WEEF, l'index WEEF 2023, ainsi que des articles et des nouvelles sur l'électronique éthique. Au début ou à la mi-2023, la Fondation Lenthe (le producteur du WEEF) lancera le GUIDE annuel du WEEF. Il sera distribué en ligne et en version imprimée à des milliers de partenaires et de parties prenantes du WEEF. Le GUIDE WEEF contiendra les profils des leaders d'opinion figurant dans l'INDEX 100 WEEF, des interviews de gagnants, des commentaires controversés provenant de l'extérieur de l'industrie, des interviews de leaders éthiques, et bien plus encore. Si vous voulez y participer, veuillez contacter Shenja Panik (shenja.panik@elektor.com).

220642-04 — VF : Laurent Rauber



Les animateurs Johann Wiesböck et Don Akkermans interviewant le Dr Heinemann, professeur d'éthique des affaires (FOM).



Stuart Cording était l'un des deux animateurs qui ont mené une journée riche en débats éthiques. (Source : Messe München)



Don Akkermans et C. J. Abate discutent des résultats de recherches sur les exposants d'electronica et leur représentation éthique.

FFWD

bilan du salon electronica 2022

les innovateurs n'ont pas manqué d'impressionner



Brian Tristam Williams

Après des mois de préparation et d'enthousiasme, les prix *Fast Forward* 2022 soutenus par Elektor, pour les jeunes entreprises et les scale-up, ont été décernés lors du salon electronica 2022 à Munich.

Pendant la compétition *electronica Fast Forward* 2022 en novembre, organisé par Elektor, nous cherchions des choses hors du commun, et nous n'avons pas été déçus. Nous avons enfin pu voir les jeunes entreprises, et celles en pleine croissance attendues avec impatience au cours des mois précédents. Sur le stand d'Elektor, nous avons accueilli tous ces amis dans une zone d'exposition et de présentation spécialement conçue pour la compétition *Fast Forward* 2022.

Participants

Les jeunes entreprises ont présenté leurs innovations, chacune dans son propre stand :

- **Opulo**, avec leur machine « pick-and-place » pour circuits imprimés, innovante et open-source.
- **AMSEL** nous a donné un aperçu approfondi de sa solution révolutionnaire de prétraitement des surfaces.
- **AirHood** nous a présenté sa hotte de cuisine qui n'est pas seulement une hotte, mais aussi une solution pour plan de travail portable et esthétique pour les fumées de cuisine.
- **V-Juice** nous a montré ses solutions de recharge sans fil élégantes et minces, avec plus d'applications et de possibilités que l'effet marketing, ludique et de mode.
- **Treesense** a mis en avant des capteurs environnementaux, permettant aux arbres de communiquer et de nous dire ce qu'ils veulent, ce qui permet d'économiser de nombreuses ressources manuelles dans la gestion et l'entretien de la flore.



L'espace d'exposition et de présentation Fast Forward 2022 animé par Elektor.

En pleine croissance :

- Déjà plébiscitée par les participants, la société **wheel.me** a présenté ses solutions de roues autonomes, qui devraient mettre un terme aux opérations pénibles de levage et de transport industriel et commercial.

Sur la route de l'innovation :

- **Voltera** a présenté ses produits existants, ainsi qu'un nouveau produit passionnant qui offre la possibilité d'imprimer (presque) tout sur tout.
- **Quadruped** était le favori des visiteurs du salon, attirant les foules avec ses robots à quatre pattes vantant les solutions logicielles de l'entreprise.

Et un projet universitaire :

- **L'équipe Ecurie Aix** (RWTH Aachen University) était fière de présenter son projet universitaire , renouvelé chaque année, sur les voitures électriques Formula Student entièrement autonomes.



La gagnante du premier prix est Kai Kreos-Nemcock de wheel.me.



Une roue autonome wheel.me soulève des charges lourdes intelligemment.



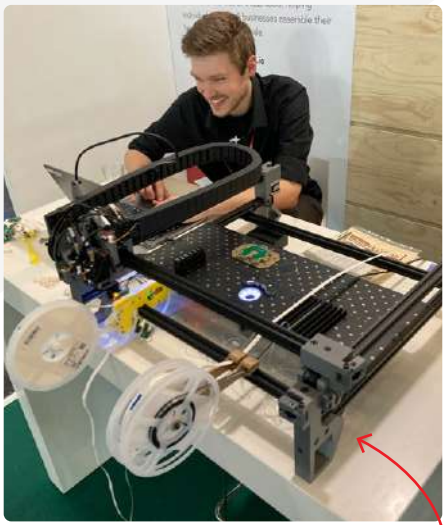
La gagnante du troisième prix est Alba Dervishi de Treesense.

Les neuf participants ont pu présenter leurs projets en profondeur le mardi et le mercredi, les start-up et les scale-up nous ont épatés avec leurs présentations rapides officielles le jeudi.

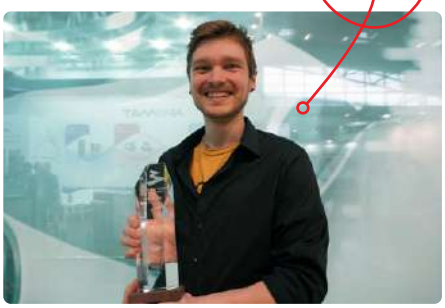
La tâche n'était pas aisée pour les membres du jury. Les délibérations se sont déroulées jusqu'à jeudi tard dans la nuit, et la liste des gagnants a été annoncée vendredi.

Premier prix

La société **wheel.me**, qui a remporté un prix média de 75 000 euros auprès d'Elektor, a fait de grands progrès dans l'élimination de la pénibilité industrielle, avec des applications comme la logistique et le stockage dans les hôpitaux [1]. Leurs roues ne sont pas seulement performantes, elles soulèvent aussi une grosse charge là où c'est nécessaire, mais elles sont aussi connectées, sensibles et autonomes. Nous sommes impatients de les voir se développer et d'observer l'impact qu'elles auront dans l'industrie !



Stephen Hawes met en place sa machine de placement de composants Opulo.



Le gagnant du deuxième prix est Stephen Hawes d'Opulo .

Deuxième prix

Stephen Hawes a fait preuve d'un enthousiasme contagieux tout au long de la semaine, et sa machine de placement de composants **Opulo**, qu'il a montée étape par étape, en la documentant minutieusement et en la filmant sur YouTube, a remporté le 2e prix. Cela signifie qu'**Opulo** pourra promouvoir son offre avec 50 000 euros de budget marketing à utiliser à partir du kit média Elektor. [2].

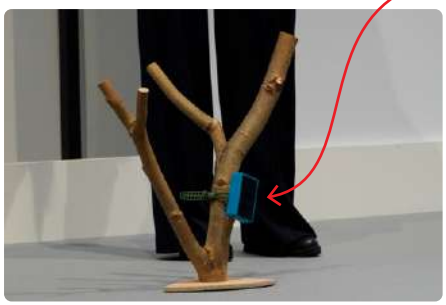
Troisième prix

Treesense était le favori parmi les participants à cause de sa fibre environnementale, apportant une innovation électronique sous la forme de capteurs intégrés pour prendre le pouls des arbres et faire savoir aux humains qui s'en occupent ce dont ils ont besoin. [3]. Avec le budget marketing de 25 000 euros alloué par Elektor, **Treesense** a pu faire la lumière sur un sujet qui ne reçoit pas autant d'attention qu'il le mérite : l'entretien des arbres de notre planète.

Voilà, c'est à peu près tout. Nous n'avons jamais parlé d'un quatrième prix. Mais finalement, avec un jury de fin de soirée passionné et dans l'impasse, nous ne pouvions pas passer à autre chose sans reconnaître un participant et son potentiel, en admettant « Il y a quelque-chose là ».

Prix d'encouragement

AMSEL a remporté un budget marketing de 10 000 € pour faire décoller son incroyable technologie eBlaze. Oui, nous avons été obligés d'inventer un prix supplémentaire cette année, car cette technologie de pointe de prétraitement des surfaces, bien qu'encore à ses débuts, ne pouvait pas passer inaperçue. C'est à regarder de près, alors que nous avons invité leur PDG, le Dr Salman Asad [4], à revenir pour Fast Forward 2024, où il pourra prouver qu'il y avait vraiment quelque chose à développer.



Grâce aux capteurs Treesense, les arbres peuvent nous dire ce dont ils ont besoin.



Le Dr. Salman Asad (AMSEL) est le lauréat du prix d'encouragement.

Félicitations à tous les gagnants. Nous suivrons vos progrès avec grand intérêt au cours des deux prochaines années !

220651-04 — VF : Laurent Rauber

LIENS

[1] wheel.me : https://wheel.me
[2] Opulo : https://opulo.io
[3] Treesense : https://treesense.net
[4] Profil de Salman Asad (AMSEL) sur LinkedIn : https://linkedin.com/in/syed-salman-asad-86337216

The Tube

un surprenant amplificateur à tubes



Gerd Reime (Allemagne)

Cet amplificateur original se distingue par de très bonnes performances et une esthétique hors du commun. Construction artisanale, circuits d'audiophile et tubes à vide conjuguent aspects nostalgiques et apparence moderne.

J'ai construit mon premier amplificateur à tubes en 1971, je n'avais alors que 15 ans. Il était animé par deux EL84 fonctionnant en push-pull et utilisait un schéma trouvé dans Elektor. Le reste des composants provenait de vieilles radios. Peu de temps après, j'ai commencé un apprentissage de technicien radio et télévision chez Grundig. Plus tard, j'ai travaillé en R&D. Ayant pris ma retraite, j'avais à cœur de travailler sur un projet original. Je pensais à un modeste amplificateur de puissance de 10 à 15 W utilisant deux EL84, comme autrefois, mais amélioré.

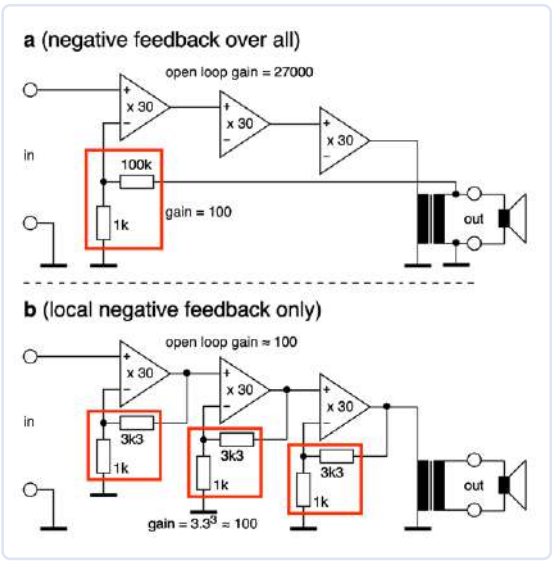


Figure 1. Différentes approches de la contreréaction : (a) contreréaction globale classique et (b) contreréaction locale à chaque étage de l'amplificateur.

Au départ, je voulais que l'esthétique du boîtier sorte des sentiers battus, mais pris de passion pour la technique des lampes, je tentai de mettre au point le meilleur circuit possible à l'aide de nombreux essais et mesures. Ce qui n'était qu'un projet amusant devint très sérieux. Ceux qui souhaitent s'inspirer de mes idées trouveront l'essentiel dans la description qui suit. Les instructions complètes sont plutôt élaborées et trop vastes pour un article d'Elektor. Pour le détail, vous pouvez consulter le fichier PDF disponible en téléchargement [1].

Considérations préliminaires

De nombreux audiophiles ne jurent que par les amplificateurs à tubes. Pourquoi ? La chaleur du son et la douceur de la réponse à la saturation sont souvent cités. Presque tous les circuits que je connais fonctionnent avec une rétroaction négative (ou contreréaction), mais quels sont ses effets ?

Comme le suggère son nom, la tâche d'un amplificateur audio est d'amplifier le faible signal d'une source audio afin qu'un haut-parleur puisse le reproduire. Pour cela, la tension et le courant du signal audio doivent être amplifiés afin que la bobine et le cône du haut-parleur *dynamique* (le plus utilisé) puissent être entraînés avec une amplitude suffisante pour que l'oreille les entende convenablement. C'est facile avec les amplificateurs à semi-conducteurs, mais les amplificateurs à tubes fonctionnent avec des tensions plutôt élevées et donc des courants plutôt faibles, qui ne peuvent pas attaquer directement un haut-parleur. Un transformateur de sortie est généralement nécessaire pour adapter l'impédance. Depuis toujours les amateurs passionnés de hi-fi tentent de répondre à la question de savoir quel est le meilleur transformateur.

En outre, la caractéristique de fonctionnement des tubes n'est pas linéaire sur une plage aussi étendue qu'on le souhaiterait. La contreréaction parvient à réduire la distorsion du signal audio qui en résulte. Une partie du signal de

sortie est renvoyée en opposition de phase vers l'entrée, et le gain élevé en boucle ouverte (très « excédentaire » par rapport au gain final de l'amplificateur) réduit les non-linéarités qui se produiraient autrement. Sur la **figure 1a**, la contreréaction est fournie par le diviseur de tension entouré en rouge. Toutes les non-linéarités et tous les signaux de distorsion générés dans l'amplificateur sont réduits d'un facteur égal au rapport *gain en boucle « ouverte » / gain défini par la contreréaction*. Dans l'exemple de la figure 1a, ce facteur vaut *27 000/100 = 270*. Comme la contreréaction est censée réduire toute la distorsion de ce facteur, les défauts de conception ont moins d'effet.

Une alternative à une forte contreréaction globale (de la sortie du haut-parleur à l'entrée du 1er étage de l'amplificateur) consiste à limiter le gain de chaque étage d'amplification par une contreréaction locale. Sur la **figure 1b**, c'est le rôle des trois diviseurs de tension entourés en rouge et correspondant fonctionnellement aux résistances d'anode et de cathode des tubes). Cela linéarise chacun des étages d'amplification, et donc l'ampli tout entier qui ne produit qu'une faible distorsion et n'a pas besoin de contreréaction globale. La question est de savoir si cela fait une différence et si c'est perceptible à l'écoute.

La différence la plus frappante avec un amplificateur à semi-conducteurs est la présence d'un transformateur de sortie en amont du haut-parleur. Le transformateur de sortie est soit à l'intérieur de la boucle de rétroaction (**figure 1a**), soit à l'extérieur (**figure 1b**).

On peut s'attendre à ce que le schéma de la **figure 1a** ait une caractéristique tension/fréquence assez droite. Mes mesures sur ce type d'amplificateur le confirment (**figure 2a**). La courbe en tension est très plate, mais la courbe en courant montre une forte baisse autour d'1 kHz car l'impédance du haut-parleur est maximale à cette fréquence. Par conséquent, une chute de puissance peut également être observée à 1 kHz. Avec le transfo à l'extérieur (**figure 2 b**) l'impédance de charge est plus élevée à 1 kHz : la tension augmente et la chute de courant n'est pas aussi forte. La variation d'impédance du haut-parleur influe donc moins sur la puissance de sortie, ce qui devrait se traduire par un son plus fidèle. Un effet similaire se produit en basse fréquence, à la résonance du woofer autour de 60 Hz (**figure 3**). Là encore, la variation d'impédance du haut-parleur influe moins si le transfo de sortie est hors de la boucle de contreréaction. J'ai répété ces mesures avec divers haut-parleurs, elles donnent en général un résultat analogue.

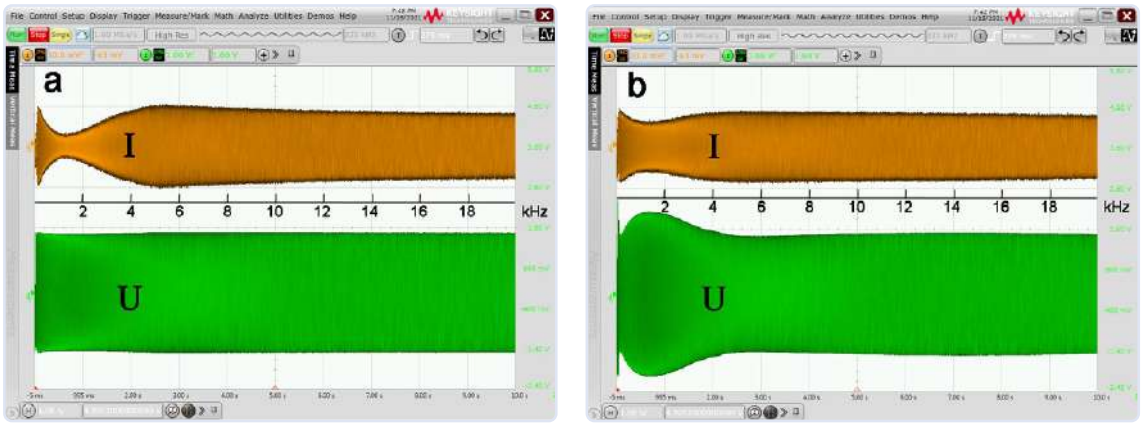


Figure 2. Courant (ocre) et tension (vert) sur la gamme de fréquences (20 Hz à 20 kHz) : (a) contre-réaction globale classique et (b) contre-réaction locale.

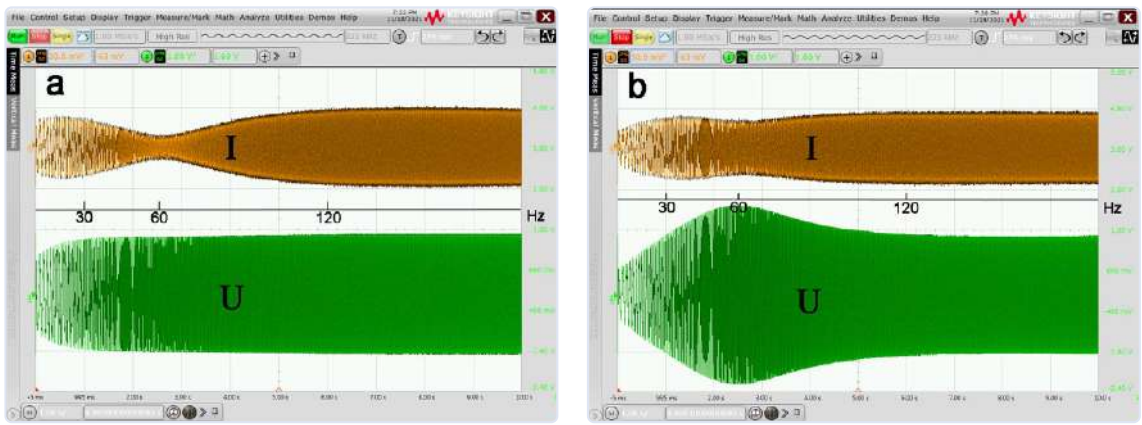


Figure 3. Courant (ocre) et tension (vert) dans la région des basses : (a) contre-réaction globale classique et (b) contre-réaction locale.

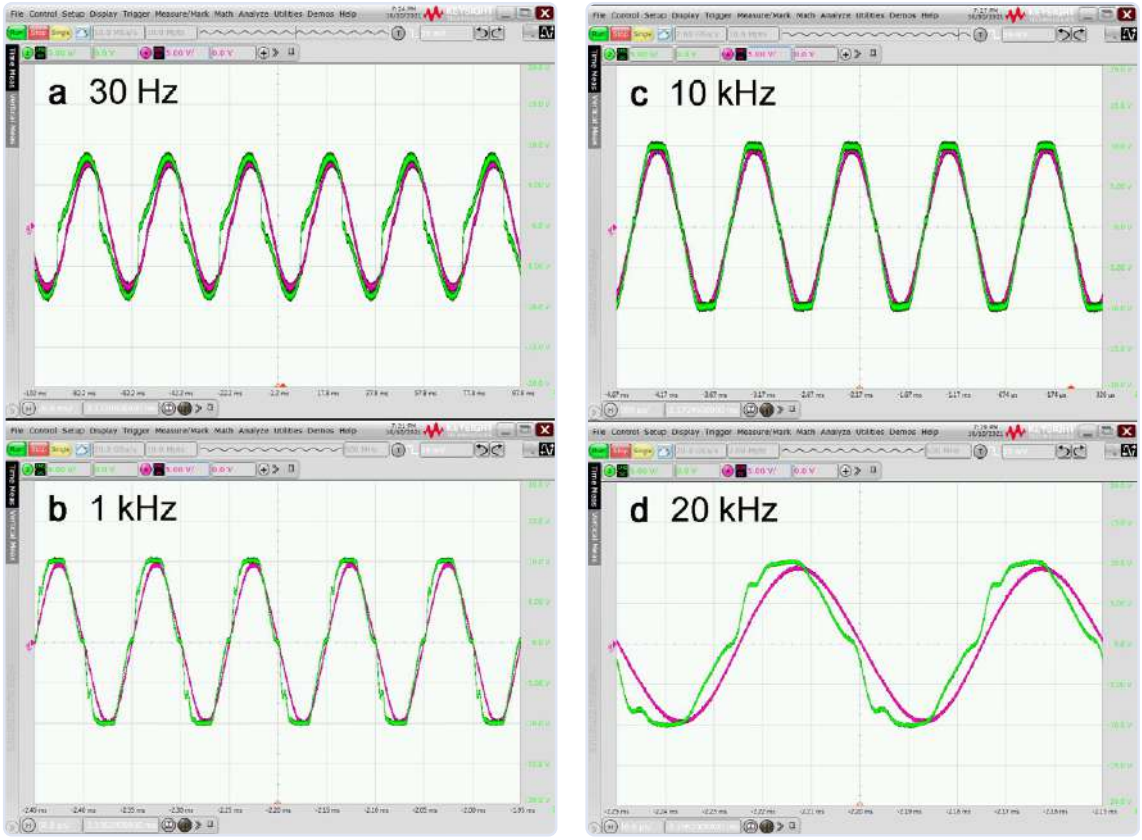


*Figure 4. Comportement en saturation à différentes fréquences (**vert** = contreréaction globale classique ; **rouge** = contreréaction locale).*

Pour conclure, un ampli à tubes sans contreréaction du HP vers l'un des premiers étages (version b) a un comportement sensiblement différent d'un ampli classique (version a). Bien sûr, je ne me suis pas limité à ces simples mesures sur les deux versions ; j'ai également demandé à des collègues de participer à des essais d'écoute. Les résultats sont clairs : la version b, sans contreréaction globale, est toujours arrivée en tête dans ces comparaisons. Le son était ressenti comme nettement plus chaud et plus équilibré. Selon le genre musical concerné, la présence était ressentie comme plus intense. Ces deux résultats corroborent sans surprise ceux des mesures.

Autre différence : à faible volume ou puissance, un étage de sortie push-pull de classe A/B fonctionne en classe A et présente donc naturellement moins de distorsion. C'est pourquoi à faible volume, avec ou sans contreréaction et un bon transformateur de sortie, la distorsion harmonique devrait être similaire. La surprise vint du circuit sans contreréaction globale qui avait un contenu harmonique nettement inférieur, en particulier en classe A. Par ailleurs, les meilleurs transfos de sortie sont à mon avis ceux de Menno Van der Veen [2].

Écrêtage et saturation

C'est notoire, si on pousse le volume, un ampli à tubes écrête le signal en douceur, lorsque les tensions de pointe du signal approchent les tensions d'alimentation. Ce sont en général, les tubes de sortie et le transfo de sortie (qui entre en saturation dans ces moments-là) qui limitent l'amplitude. C'est aussi là que la contreréaction globale devient problématique car lorsque l'écrêtage se produit, la rétroaction force les étages amont non limités en gain à piloter l'étage de sortie encore plus fort, ce qui entraîne directement une forte augmentation de la distorsion harmonique. Sans cette forme de contreréaction, l'augmentation de la distorsion harmonique est nettement plus douce en cas de saturation des étages de sortie.

Les transformateurs de sortie sont souvent incapables de transférer de fortes puissances, en particulier à basse fréquence de l'ordre de 20 à 30 Hz. C'est pourquoi les effets négatifs de la contreréaction deviennent audibles dans la partie TBF du spectre, même à faible puissance. À cela s'ajoutent les effets de la courbe de phase du transfo qui produisent aussi des effets sonores indésirables dus à la contreréaction.

Mes mesures à différentes fréquences avec 10 % de saturation confirment ces considérations. Les signaux de l'ampli avec contreréaction locale (en rouge) sont systématiquement meilleurs que ceux (en vert) plus ou moins déformés de l'ampli avec contreréaction globale (**figure 4**). Avec la contreréaction globale, à 20 kHz les différences sont spectaculaires et à 30 Hz des dégradations notables sont visibles. Il est intéressant de noter qu'à cette fréquence de 30 Hz, 10 % de saturation représentent 85 % du signal nominal. En résumé, on peut affirmer à juste titre qu'un écrêtage vraiment doux et sans effet perturbateur notable n'est possible qu'avec une contreréaction locale.

Contrôle du volume à correction physiologique

On lit souvent que seule une réponse linéaire est correcte. En corollaire, tout bon amplificateur doit avoir une

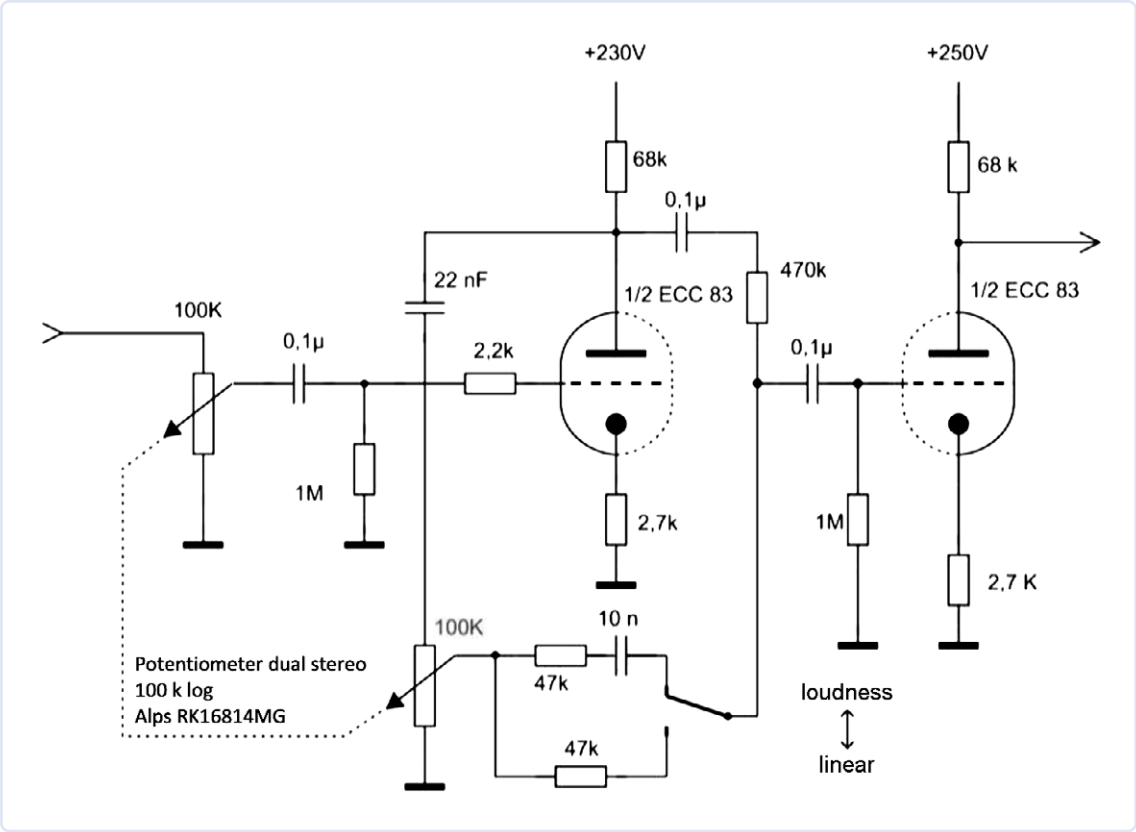


Figure 5. Commande de volume à correction physiologique avec potentiomètre Alps quad.

réponse en fréquence plate. Je soupçonne en revanche que l'exigence d'une réponse en fréquence plate soit en fait basée sur le fait que les potentiomètres avec prises ne sont plus disponibles aujourd'hui. Les circuits équivalents (que l'on trouve uniquement en ligne) restent très moyens. La raison à tout cela, c'est qu'à faible volume, l'ouïe humaine est moins sensible aux hautes et basses fréquences, ou (question de point de vue) elle privilégie les fréquences moyennes.

Autrefois, la compensation de cette non-linéarité auditive, la correction physiologique, était standard. Le principe est d'accentuer les hautes et basses fréquences à faible volume, et de diminuer progressivement cet effet quand le volume augmente. Mais combien d'amplificateurs modernes offrent cette fonction ? Pratiquement aucun.

À côté des potentiomètres vintage à deux ou trois prises, de toute façon non idéals en raison des pas, il y a d'autres solutions plus élégantes. Prenez un potentiomètre stéréo double (en fait un potentiomètre quadruple) comme on en trouve encore chez Alps et ailleurs. Avec cela et un circuit simple, on peut réaliser une excellente commande de volume à correction physiologique. En outre, un simple commutateur permet de passer de la correction à la linéarité, sans modifier l'ampli.

J'ai conçu un circuit avec le potentiomètre Alps *RK16814MG* (**figure 5**). Ce potentiomètre est motorisé, mais au besoin, le moteur peut être retiré pour actionner le potentiomètre manuellement. Pour quantifier l'effet du contrôle de compensation auditive, j'ai mesuré et normalisé au même niveau de signal à 1 kHz la réponse en fréquence pour différents réglages, et j'ai superposé les courbes. L'effet sur les basses est visible sur la **figure 6**. On voit clairement que la correction de la réponse diminue progressivement quand l'intensité sonore augmente.

Pilotage des tubes de sortie

Les étages push-pull ont bien sûr besoin de signaux déphasés de 180°. Par rapport à un circuit simple à un seul tube, le schéma d'une voie de *The Tube* ajoute un brin de complexité et améliore la qualité audio (**figure 7**).

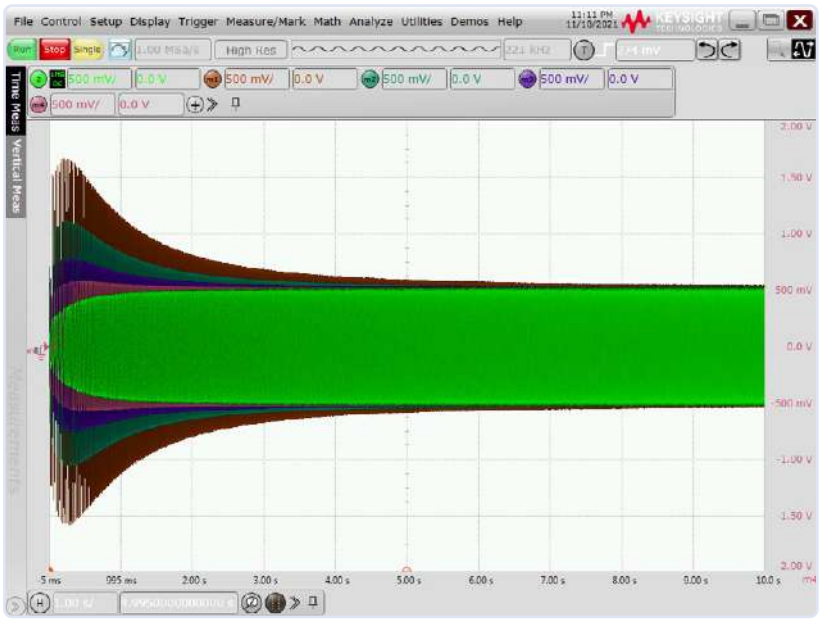


Figure 6. Effet de la commande physiologique : marron = faible niveau sonore ; vert foncé = calme (musique de fond) ; bleu = niveau sonore moyen ; rouge = assez fort (réponse en fréquence linéaire) ; vert = intensité sonore maximale juste avant saturation.

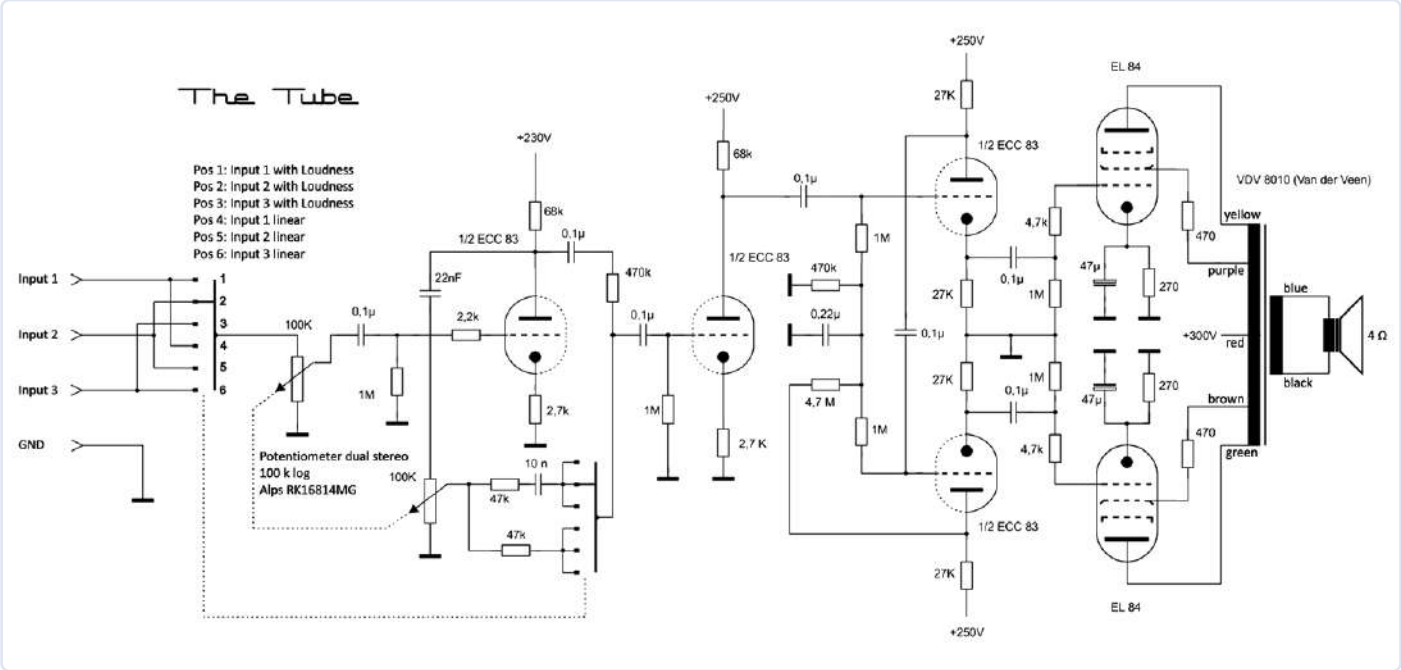


Figure 7. Schéma électronique d'une voie de ***The Tube****. Caractéristiques originales : commande de volume à correcteur physiologique et commande symétrique des tubes de sortie push-pull.*

Chaque tube de sortie EL84 est piloté par son propre cathode follower (1/2 ECC83). Grâce à des résistances d'anode et de cathode identiques, la triode du haut pilote avec précision la grille de celle du bas pour produire un déphasage symétrique presque parfait. Ce schéma réduit la distorsion et les erreurs de phase. Ainsi, on obtient un amplificateur correct avec un son « tube » typique, mais sans distorsion harmonique, malgré l'absence de contreréaction globale. Divers essais d'écoute comparatifs confirment que ce schéma est (subjectivement) meilleur qu'un amplificateur à tubes haut de gamme très coûteux et gourmand en énergie (classe A oblige) et qu'il fait même mordre la poussière à de bons amplificateurs à semi-conducteurs.

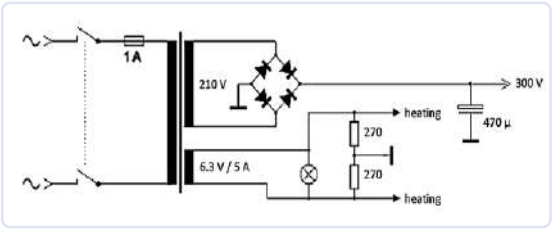


Figure 8. Le schéma de l'alimentation électrique est assez simple.

La schéma de base de l'alimentation est relativement simple (**figure 8**). Pour les détails, voir le lien [1].

Construction

L'idée a germé quand mes fils ont trouvé quelque part un lot de gros tubes métalliques. J'ai été convaincu sur le champ que des « tubes » seraient idéals pour abriter un amplificateur à… tubes.

Le recto de la plaque supérieure en aluminium massif accueille les jolis supports des tubes. Comme on peut le voir (**figure 9**), les tubes de l'étage intermédiaire sont voisins des tubes de sortie. En raison du dégagement de chaleur attendu, il n'est pas question d'utiliser du plastique ici.

Le verso de la plaque d'aluminium (**figure 10**) est équipé d'entretoises sur lesquelles la « carte à circuit imprimé » est vissée. Les guillemets sont de rigueur pour deux

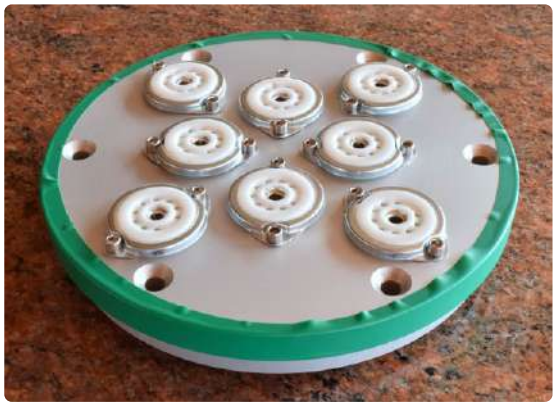


Figure 9. Construction mécanique attrayante pour les composants électroniques : les douilles de tous les tubes de l'amplificateur stéréo.



Figure 10. Les entretoises s'accordent très bien à la plaque d'aluminium massif.



Figure 11. Conçu spécifiquement pour les tubes : un « circuit imprimé » en laiton massif.

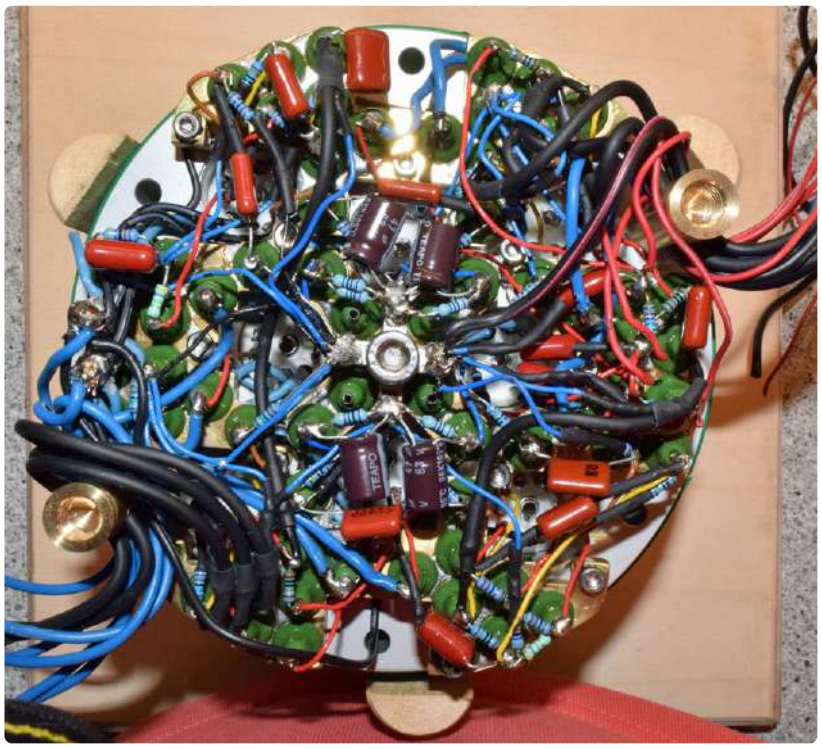


Figure 12. Plots de soudage avec isolateurs en verre vert presque vintage.

raisons : d'une part un circuit imprimé ce n'est pas courant dans un amplificateur à tubes et d'autre part, il est composé de supports de composants (**figure 11**) fraisés dans des plaques de laiton massif et il n'y a pas d'époxy. Après tout, il n'y a aucune raison de négliger l'esthétique des composants internes d'un tel amplificateur à tubes. Le verso de la plaque d'aluminium avec les parties en laiton usiné et les jolis plots de soudage à isolateurs en verre vert le prouve (**figure 12**).



Figure 14. Les gros condensateurs électrolytiques ont leur propre plaque de laiton.



▲

Figure 13. Vue de dessous (côté soudure) de la « carte » entièrement assemblée. Un peu de fouillis, mais c'est parfaitement robuste.

La plaque entièrement assemblée est très propre (**figure 13**). Une autre « carte » accueille des composants plutôt grands, tels que des condensateurs électrolytiques (**figure 14**). D'autres plaques-support en laiton ont dû être usinées pour le transformateur d'alimentation, circulaire puisque toroïdal, admirez ! (**figure 15**). Bien sûr, cela ne clôt pas le chapitre mécanique. Assez logiquement, le panneau avant est un anneau en aluminium sur lequel sont montés le sélecteur d'entrée, l'interrupteur

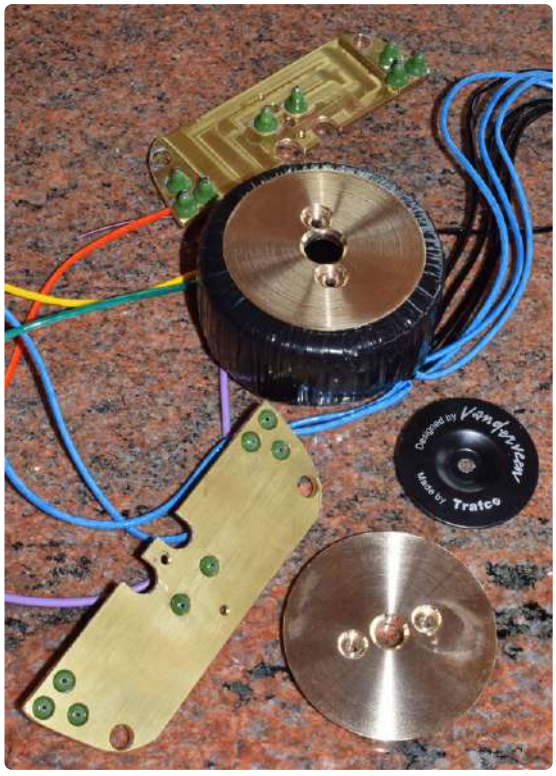


Figure 15. Tout sera rond, même le transfo, ou cela ne sera pas. Ici aussi, des pièces mécaniques de fixation sont nécessaires.

◀



Figure 16. Les commandes (commutateur d'entrée, interrupteur d'alimentation et potentiomètre de volume) sont montées sur un anneau d'aluminium usiné distinct.

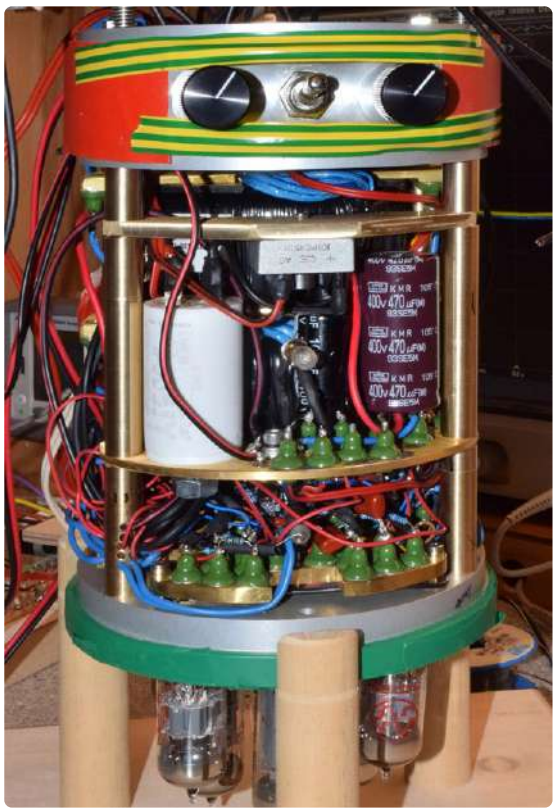


Figure 17. Assemblage provisoire de l'amplificateur complet, sans le boîtier tubulaire de protection.



Figure 18. L'amplificateur terminé dans toute sa gloire. Un vrai bijou — plus que compatible avec un salon.

d'alimentation et le potentiomètre de réglage du volume (**figure 16**). L'assemblage provisoire présenté (**figure 17**) donne une bonne impression de cette disposition. Et maintenant - roulement de tambour ! - l'amplificateur terminé dans son boîtier (**figure 18**).

Si on soigne bien les détails, un tel amplificateur à tubes affirme toujours une forte esthétique qui souligne sa technologie. Mais ne manque-t-il pas quelque chose d'important ? Comme tout amplificateur, *The Tube* nécessite des connecteurs d'entrée des signaux audio faibles et de sortie des signaux puissants vers les haut-parleurs.



Figure 19. Une cavité à l'arrière abrite tous les connecteurs : entrées, sorties et alimentation secteur.

Par ailleurs, le secteur doit atteindre le transformateur d'alimentation. Un cavité dans le boîtier tubulaire a résolu la question (**figure 19**). Elle abrite les fiches Cinch plaquées or pour les entrées, les prises XLR des haut-parleurs et un connecteur d'alimentation pour le câble secteur.

Vous vous sentez inspiré ?

Comme le montre ce projet, pour un amplificateur à tubes, concevoir un bon circuit n'est que la moitié de l'histoire. Pour paraphraser Thomas A. Edison, à un pincée d'inspiration (électronique) il faut ajouter pas mal de transpiration (mécanique). Si vous voulez vous inspirer de mon projet, sachez qu'il faudra mettre beaucoup d'efforts dans la réalisation afin d'obtenir un amplificateur à tubes qui non seulement sonne bien, mais aussi attire les regards. Le vieil adage « *no pain, no gain* » prend tout son sens avec la technologie des tubes.

220089-04 — VF : Yves Georges

À propos de l'auteur

Après un apprentissage de technicien en radio et télévision, Gerd Reime a travaillé successivement dans les services de développement de Grundig et Nokia. Il est titulaire de plusieurs brevets.

Des questions, des commentaires ?

Si vous avez des questions techniques ou des commentaires sur l'article, envoyez un courriel à l'équipe de rédaction d'Elektor (redaction@elektor.fr).

Produits

- *Elektor Audio Collection* **(clé USB) (SKU 19892)**
  https://www.elektor.fr/19892
- **Livre en anglais « The State of Hollow State Audio », R. A. Honeycutt (Elektor 2020)**
  www.elektor.fr/19170

LIENS

[1] Description complète du projet (PDF) : https://www.elektormagazine.fr/220089-04
[2] Transformateurs Van der Veen : https://www.mennovanderveen.nl

à quand des **biomatériaux** pour **l'électronique ?**



Source : CHUTTERSNAP, UnSplash

Priscilla Haring-Kuipers (Pays-Bas)

Une grande partie des dommages environnementaux imputables à l'électronique provient des matériaux utilisés. Sommes-nous prêts à adopter des biomatériaux plus respectueux de l'environnement ?

Recyclage

En électronique, le terme durabilité fait généralement référence au recyclage, et il y a effectivement beaucoup à faire de ce côté-là. Moins de 20 % des appareils électroniques sont recyclés de par le monde, et la plupart d'entre eux sont traités par des recycleurs de déchets électroniques peu transparents, voire par des ferrailleurs et des mineurs urbains [1]. Le processus de recyclage pourrait être amélioré dès l'amont, notamment en recourant à des biomatériaux adaptés à l'économie circulaire. Ce sont des matériaux propres, non toxiques, et plus faciles à traiter.

Les inconvénients du FR-4

La plupart des produits électroniques reposent sur des circuits imprimés composés de FR-4 et de feuilles de cuivre. Le FR-4 (de l'anglais *Flame Retardant class 4*) est une résine époxy renforcée par des fibres de verre. La résine elle-même est un polymère thermodurci dont la structure moléculaire contient des « anneaux époxydes » [2]. Elle est souvent obtenue à partir de dérivés du pétrole et possède une empreinte carbone conséquente (5,7 à 7,6 kg de CO_2 par kg) [3]. Sa fabrication peut provoquer de l'asthme professionnel et des lésions pulmonaires irréversibles. Sa grande viscosité à haute température ambiante requiert par ailleurs l'emploi de diluants à base d'hydrocarbures.

De la même façon qu'un gâteau cuit ne peut être ramené à ses ingrédients de départ, une résine époxy ne peut être décomposée en ses constituants initiaux qui, donc, ne peuvent être réutilisés. Le recyclage de la résine est très difficile en raison même de la réaction chimique qui lui a donné ses exceptionnelles propriétés physico-chimiques. « Il faudra davantage de recherches pour trouver des polymères biosourcés performants et recyclables à même de remplacer les polymères synthétiques des circuits imprimés » [4].

La fabrication, le traitement et l'élimination du FR-4 créent des problèmes que l'emploi d'autres matériaux pourrait éviter. Le goudron de houille, qui était un des ingrédients de base du FR-4, a ainsi déjà été remplacé par d'autres dérivés de combustibles fossiles. D'autres ingrédients sont à l'étude.

Des chercheurs ont par exemple conçu des matrices de résine biosourcées à base de papier et de lin, et contenant moins de retardateurs de flamme toxiques ; d'autres utilisent des substituts à base d'huile de soja et de chanvre. Contrairement au FR-4, ces bio-cartes ont l'inconvénient d'être sensibles à l'humidité.

Vers plus de bio

Le PLA, ou acide polylactique, peut être synthétisé à partir d'amidon de maïs et est biodégradable. Aussi séduisant soit-il, il résiste toutefois mal au stress thermique, une faiblesse incompatible avec les dépôts de métaux en phase vapeur, fours à refusion et autre soudage à la vague propres à l'électronique. Un traitement chimique est à l'étude pour le rendre plus résistant à la chaleur. Comme me l'a dit un chimiste : « Aujourd'hui nous savons faire croître toutes sortes de bioplastiques. Il vous suffit d'énoncer vos exigences. »

Les matériaux conducteurs à base de carbone sont plus résistants à la chaleur et ont une meilleure conductivité électrique. Un grand nombre peuvent être synthétisés à partir de sources renouvelables et pourraient donc remplacer divers métaux. Les nanotubes de carbone peuvent être obtenus à partir de pratiquement n'importe quel composé organique, et le graphène peut être fabriqué par exfoliation du graphite. Plusieurs universités américaines tentent d'utiliser de l'encre de graphène biosourcée pour imprimer des circuits sur des substrats eux aussi biosourcés,



Source : Michael Schiffer, UnSplash

« des outils qui pour l'instant n'existent qu'en laboratoire ou dans l'imagination des chercheurs ».

Des universités finlandaises cherchent quant à elles à fabriquer des capteurs piézoélectriques flexibles à partir de fibres de bois et de bactéries, des supercondensateurs à partir de pissenlit, et divers biocapteurs, électrodes et cartes à partir de « nouveaux matériaux verts » comme la cellulose.

La définition du terme « biosourcé » a évolué. Selon la FDA américaine (*Food and Drug Administration*), les bioplastiques comprennent désormais les produits fabriqués à partir de polymères dérivés du pétrole et mélangés à des fibres naturelles. N'étant pas biodégradables, ces matériaux ne correspondent sans doute pas à l'idée que vous vous faites d'un « bioplastique ».

Quelle durée pour le durable ?

Deux caractéristiques opposent les plastiques classiques et les biomatériaux : les uns durent, les autres sont biodégradables. Les premiers résistent même si bien aux agents de vieillissement naturels (comme l'humidité et la température) qu'ils finissent par former des montagnes de déchets électroniques que nous ne parvenons pas à recycler correctement. Les biomatériaux, eux, se dégradent avec le temps. C'est appréciable pour un produit en fin de vie, ça l'est un peu moins lorsque vous n'en êtes qu'au stade de la conception dudit produit.

La plupart des produits électroniques devraient en fait être dégradables. Certains sont à usage unique, les étiquettes RFID p. ex., et la majorité des autres n'a pas vocation à être utilisé durant des décennies. Alors peut-être la valeur par défaut devrait-elle être la biodégradabilité plutôt qu'une durée de vie illimitée. Si vraiment il vous fallait fabriquer un produit à partir de plastiques ou de résines non-dégradables, alors vous devriez aussi prouver que vous saurez offrir un service d'assistance technique couvrant au moins les 50 prochaines années.

Des initiatives comme la Loi sur le climat et le Pacte vert pour l'Europe permettront un usage plus responsable des matériaux. Les progrès de la science des matériaux laissent eux aussi espérer une « accélération de la mise en œuvre d'une économie biosourcée ». Si aucun de ces matériaux n'est à ma connaissance utilisable à grande échelle, la plupart des chercheurs semblent viser une mise sur le marché d'ici cinq ans. Je m'attends à de grandes bio-choses.

(220556-04) — VF : Hervé Moreau



Source : Fukayamamo, UnSplash

WORLD ETHICAL ELECTRONICS FORUM

Forum mondial de l'électronique éthique 2023

En novembre 2021, Elektor a lancé à Munich le **World Ethical Electronics Forum** (WEEF) avec pour objectif d'ouvrir un débat public sur l'éthique et les objectifs de développement durable. Parmi les intervenants figuraient, outre les ingénieurs et rédacteurs d'Elektor : Stefan Heinemann (Professeur d'éthique commerciale à l'école de commerce FOM), Paula Palade (PhD, Jaguar Land Rover), Margot Cooijmans (Directrice, Philips Foundation), ainsi que plusieurs autres spécialistes des questions d'éthique, dont Priscilla Haring-Kuipers. Pour en savoir plus sur le WEEF, visitez la page elektormagazine.com/weef.

LIENS

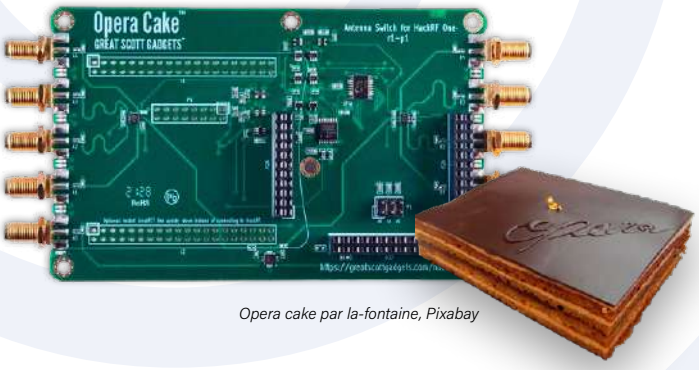
[1] eWaste Ben : https://www.youtube.com/c/eWasteBen
[2] Custom Materials, Inc., Is Epoxy Resin Plastic?, janvier, 2022 : https://www.youtube.com/watch?v=i8nEQQ9S0V0
[3] NIH, Bioplastics and Carbon-Based Sustainable Materials, Components, and Devices: Toward Green Electronics, 20 ocotbre 2021 : https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8532127/
[4] O. Ogunseitan, et al, Biobased materials for sustainable printed circuit boards, Nature Reviews, 12 septembre 2022 : https://www.nature.com/articles/s41578-022-00485-2#citeas

commutateur d'antenne Opera Cake pour HackRF One

connectez jusqu'à huit antennes à votre SDR

Clemens Valens (Elektor)

L'Opera Cake de Great Scott Gadgets est un commutateur d'antenne destiné au HackRF One SDR. Avec lui, vous pouvez connecter jusqu'à huit antennes à votre système ou insérer une banque de filtres commutables.



Opera cake par la-fontaine, Pixabay

Suivant la tradition des gadgets électroniques portant le nom de gâteaux, de pâtisseries et de sucreries (pensez à Raspberry et Banana Pi, Snickerdoodle [1] et aux premières versions d'Android), le commutateur d'antenne Opera Cake est un appareil qui n'a aucun rapport avec son nom (au moins Pi fait référence à Python).

Le gâteau Opéra que je connais (et que j'ai même fait quelques fois) est un gâteau multicouches qui a surtout le goût du café. Et c'est peut-être le rapport avec l'appareil examiné ci-dessous ? Vous aurez besoin de beaucoup de café pour rester éveillé pendant les longues heures de plaisir qu'il vous procurera.

L'Opera Cake commutateur d'antenne…

Comme mentionné ci-dessus, l'Opera Cake est un commutateur d'antenne pour HackRF One [2], comme le montre la **figure 1**. En fait, il s'agit d'un double commutateur d'antenne à quatre voies, c'est-à-dire une paire de commutateurs unipolaires à quatre positions (SP4T). Vous pouvez utiliser les deux commutateurs, également appelés banques, en parallèle comme un commutateur DP4T ou en série (en quelque sorte) comme un commutateur SP8T. Les commutateurs sont bidirectionnels, ils fonctionnent donc en mode RX et TX.

Commençons par décrire à quoi ressemble l'Opera Cake. En gros, il s'agit d'une carte de 120 mm sur 75 mm à quatre couches (attendez une minute, quatre couches, gâteau en couches, hmm…) avec beaucoup de connecteurs. Il y a cinq SMA femelles à chaque extrémité de la carte et trois connecteurs empilables montés sur celle-ci, avec des empreintes pour trois autres.

… est une carte d'extension

La carte comporte relativement peu de pièces électroniques. Cela s'explique par le fait que l'Opera Cake est une carte d'extension. Elle ne nécessite aucune intelligence, puisque la carte sur laquelle elle se branche la contrôle. En termes de taille, il est destiné à être branché sur une carte HackRF One. Lorsque vous y soudez les bons connecteurs, vous pouvez également y brancher un GreatFET One [3] (à l'envers). Notez que si vous avez un HackRF One installé dans un boîtier, vous devrez l'ouvrir (avec soin !) avant de pouvoir l'y brancher.

Configuration

La première application du commutateur d'antenne qui vient à l'esprit est, bien sûr, la connexion d'une variété d'antennes à un seul HackRF One afin que vous n'ayez pas à recâbler votre installation chaque fois que vous voulez changer de bande RF.

Le changement d'antenne peut être fait manuellement avec un petit logiciel, ou automatiquement par le firmware du HackRF One basé sur la fréquence ou le temps. La commutation basée sur la fréquence permet, par exemple, des applications d'analyseur de spectre à large bande, tandis que la commutation basée sur le temps vous permet de faire des choses cool comme la goniométrie pseudo-Doppler [4].

En utilisant le commutateur en mode DP4T, vous pouvez l'utiliser pour insérer, par exemple, des atténuateurs ou des filtres dans le circuit de l'antenne. Ainsi, l'Opera Cake peut fonctionner comme un banc de filtres commutés.

L'Opera Cake est empilable

Sachez également que jusqu'à huit Opera Cakes peuvent être empilés les uns sur les autres (une SDR en couches, hmm…), permettant différentes configurations de commutateurs. En mode manuel, chacun peut être contrôlé séparément, mais en mode de commutation automatique de fréquence et de temps, toutes les cartes commutent en même temps et de la même manière. Ainsi, une pile de deux cartes permettrait une configuration DP8T, commutant, par exemple, huit filtres.

Pourquoi en voudrais-je un ?

Opera Cake n'est pas une nouvelle conception, il date de 2016 et peut-être même d'avant. Ce qui est nouveau, c'est que vous pouvez désormais l'acheter en tant que module entièrement assemblé et testé. Jusqu'à présent, si vous en vouliez un, vous deviez le construire vous-même à partir des fichiers de conception publiés sur GitHub [5].

Le commutateur d'antenne Opera Cake, joliment assemblé, est, sans surprise, un compagnon parfait pour HackRF One. Au lieu de vous obliger à recâbler votre matériel ou à ajuster la longueur de votre antenne chaque fois que vous voulez travailler dans une bande différente, il vous permet de connecter (et de garder connectées) jusqu'à huit de vos antennes préférées en même temps.

Quelques remarques

Selon l'âge de votre HackRF One, il se peut que vous deviez mettre à jour son firmware à la dernière version pour qu'il fonctionne avec le commutateur d'antenne. Les instructions sur la manière de procéder sont disponibles sur le site Web de Great Scott Gadgets [6].

Sachez que, comme sur HackRF One, les connecteurs SMA de l'Opera Cake sont de type femelle. Cela signifie qu'il ne fonctionnera qu'avec des antennes équipées d'un connecteur mâle (c'est-à-dire avec une broche centrale, voir la **figure 2**). Par conséquent, soyez prudent et choisissez judicieusement.

Un HackRF One avec une ou plusieurs cartes Opera Cake empilées sur lui ne rentre plus dans son boîtier. Ceci, bien sûr, rend le système vulnérable à la poussière et aux tournevis et autres objets métalliques qui tombent accidentellement sur lui. De plus, selon la manière dont vous avez câblé vos antennes et vos filtres, les boutons poussoirs *Reset* et *Power* peuvent devenir quelque peu inaccessibles (**figure 3**). Comme les connecteurs empilables P20, P22 et P28 ne transportent que l'alimentation et quelques signaux numériques, et aucun signal RF, il est possible d'insérer des connecteurs pour augmenter la distance entre les cartes. Enfin, un (court) câble SMA mâle-mâle pour connecter l'Opera Cake à l'entrée d'antenne du HackRF One n'est malheureusement pas inclus, apportez le vôtre.

220602-04 — VF : Maxime Valens

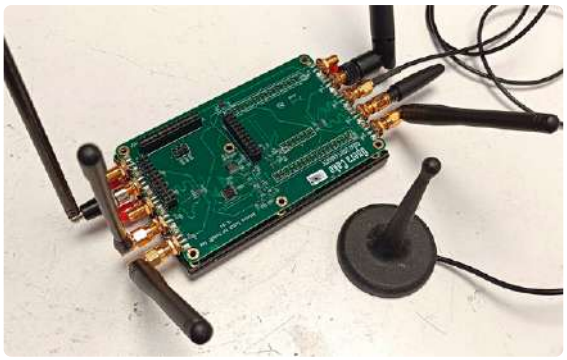


Figure 1. L'Opera Cake de Great Scott Gadgets est un commutateur d'antenne pour HackRF One.



Figure 2. Antennes avec connecteurs SMA. La gauche est adaptée pour Opera Cake et HackRF One, pas celle de droite.

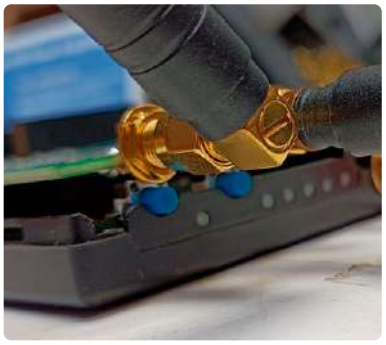


Figure 3. Vous pouvez insérer des connecteurs empilables pour augmenter l'espace entre les deux cartes.

Des questions, des commentaires ?

Vous avez des questions techniques ou des commentaires sur cet article ? Contactez Clemens Valens à l'adresse clemens.valens@elektor.com.

Produits

- **Opera Cake – Commutateur d'antenne pour HackRF One (SKU 20083)** www.elektor.fr/20083
- **Radio logicielle HackRF One (1 MHz to 6 GHz) (SKU 18306)** www.elektor.fr/18306
- **Antenne télescopique ANT500 (75 MHz to 1 GHz) (SKU 18481)** www.elektor.fr/18481
- **Antenne télescopique ANT700 (300 MHz to 1100 MHz) (SKU 18480)** www.elektor.fr/18480



LIENS

[1] Snickerdoodle sur Elektor Labs : https://www.elektormagazine.fr/labs/snickerdoodles-with-zynq
[2] D. Meyer, « Émetteur récepteur SDR HackRF One : radio logicielle de 1 MHz à 6 GHz », Elektormagazine.fr : https://www.elektormagazine.fr/news/emetteur-recepteur-sdr-hackrf-one
[3] C. Valens, « Carte à microcontrôleur GreatFET One Dual-Core pour Python », Elektormagazine.fr : https://www.elektormagazine.fr/news/carte-a-microcontroleur-greatfet-one-dual-core-pour-python
[4] Pseudo-Doppler direction finding : https://www.rtl-sdr.com/pseudo-doppler-direction-finding-with-a-hackrf-and-opera-cake/
[5] Opera Cake sur GitHub : https://github.com/greatscottgadgets/hackrf/tree/master/hardware/operacake
[6] HackRF One firmware : https://github.com/greatscottgadgets/hackrf/releases/latest

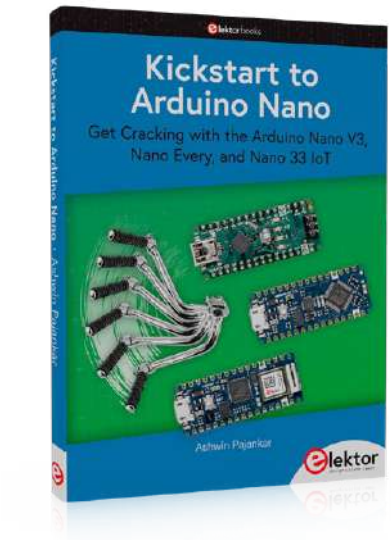


concevoir avec Arduino et plus

entretien avec l'auteur Ashwin Pajankar

Par Alina Neacsu (Elektor)

La communauté Elektor regorge d'ingénieurs qui aiment partager leurs connaissances avec d'autres passionnés d'électronique. Prenez l'exemple d'Ashwin Pajankar, un ingénieur, éducateur, auteur d'Elektor et YouTubeur travaillant à Nashik, en Inde. Lorsqu'il ne s'attaque pas à un nouveau projet, vous le trouverez en train d'aider ses pairs, par le biais de livres, de cours et de vidéos, à apprendre à travailler avec Arduino et d'autres technologies de pointe.



Lisez Kickstart to Arduino Nano pour commencer votre voyage avec l'Arduino Nano V3, Nano Every, et Nano 33 IoT.

Alina Neacsu : tout d'abord, merci de prendre le temps de répondre à ces questions. Parlez-nous de vous. Quelle est votre profession actuelle ?

Ashwin Pajankar : je vis dans la banlieue de la ville de Nashik en Inde. J'ai obtenu une licence d'ingénieur en informatique et ingénierie de l'Institut d'ingénierie et de technologie Shri Guru Gobind Singhji, à Nanded. J'ai également obtenu une maîtrise en technologie (toujours en informatique et ingénierie) de l'International Institute of Information Technology, à Hyderabad. Je parle cinq langues et j'ai vécu et travaillé dans trois états de l'Inde.

Je travaille actuellement en tant que rédacteur technique indépendant, YouTuber et professeur chez Udemy. J'organise également des ateliers de formation à la programmation et à l'électronique pour les professionnels en activité et les étudiants. J'aime les mathématiques, la physique, les ordinateurs et l'électronique.

Alina : vous souvenez-vous de votre premier projet électronique avec des microcontrôleurs ? Pouvez-vous nous le décrire et nous dire quelles technologies étaient utilisées ?

Ashwin : mon tout premier projet de microcontrôleur était très modeste. J'ai commencé à utiliser les microcontrôleurs (avec le 8085 et le 8086) en terminale. On m'a enseigné le 8051 (c'est le microcontrôleur standard utilisé dans toute l'Inde pour enseigner le sujet). Cependant, nous n'avons pas eu de travaux pratiques avec en 12ème année. Lorsque j'ai étudié l'ingénierie (un diplôme de premier cycle de quatre ans au niveau de la licence), j'ai eu l'occasion de travailler avec le kit 8051 en utilisant l'assembleur pendant la troisième année du cours. Mon tout premier projet était de faire clignoter une LED en utilisant le code en assembleur 8051.

Alina : pouvez-vous nous dire comment fonctionne l'écosystème Arduino ? Et quels en sont les avantages et les inconvénients ?

Ashwin : Arduino est un écosystème matériel et logiciel open-source. Comme la conception est open source, les gens et les entreprises peuvent fabriquer leurs propres cartes et ajouter des idées. Le logiciel est également open source et le code source se trouve sur *github.com/arduino*. Les gens peuvent contribuer à cette branche principale ou détourner le projet selon leurs envies. De nombreuses ressources pédagogiques sont disponibles sur Internet. Il est possible de s'initier à l'électronique



et à la programmation en C tout en utilisant Arduino sans dépenser une somme astronomique. Ce sont les avantages. Le seul inconvénient auquel je pense est que, parfois, des produits de contrefaçon présentés comme des produits Arduino originaux sont vendus à des amateurs non avertis.

Alina : qu'est-ce qui a suscité votre intérêt pour la carte Arduino Nano 33 IoT ?

Ashwin : eh bien, auparavant, lorsque nous devions connecter une carte Arduino au wifi, nous devions utiliser des *shield* spécialisés (qui sont un peu difficiles à obtenir en Inde) ou utiliser l'ESP-01. Le travail avec l'ESP-01 est un peu compliqué, et les débutants le trouvent souvent fastidieux. L'Arduino Nano 33 IoT est une solution IdO parfaite, prête à l'emploi. Il est équipé d'un module wifi NINA-W102. Je vérifie régulièrement les nouveaux produits sur *arduino.cc*. J'étais très enthousiaste à l'idée de voir la nouvelle carte avec wifi intégré. Ainsi, lorsqu'elle est devenue disponible en Inde, je l'ai achetée à la toute première occasion pour essayer.

Alina : avez-vous des conseils à donner aux personnes qui souhaitent utiliser l'Arduino Nano ?

Ashwin : oui. Consultez la documentation en ligne sur la page d'accueil d'Arduino. Mon livre, *Kickstart to Arduino Nano* (Elektor, 2022), contient des instructions progressives détaillées.

Alina : parlez-nous du processus d'écriture de votre premier livre. Était-ce un défi ?

Ashwin : je travaillais comme ingénieur informatique à Bangalore, en Inde, lorsque j'ai écrit et publié mon tout premier livre. Il portait sur Raspberry Pi, sur la vision par ordinateur. Bien que c'était ma toute première expérience, je n'ai pas trouvé cela très difficile, car j'aime la programmation, l'électronique et l'écriture. La rédaction de livres me permet de combiner toutes mes passions en quelque chose de très concret.



Ashwin Pandakar présent un démarrage accéléré complet pour l'Arduino Nano dans son livre.

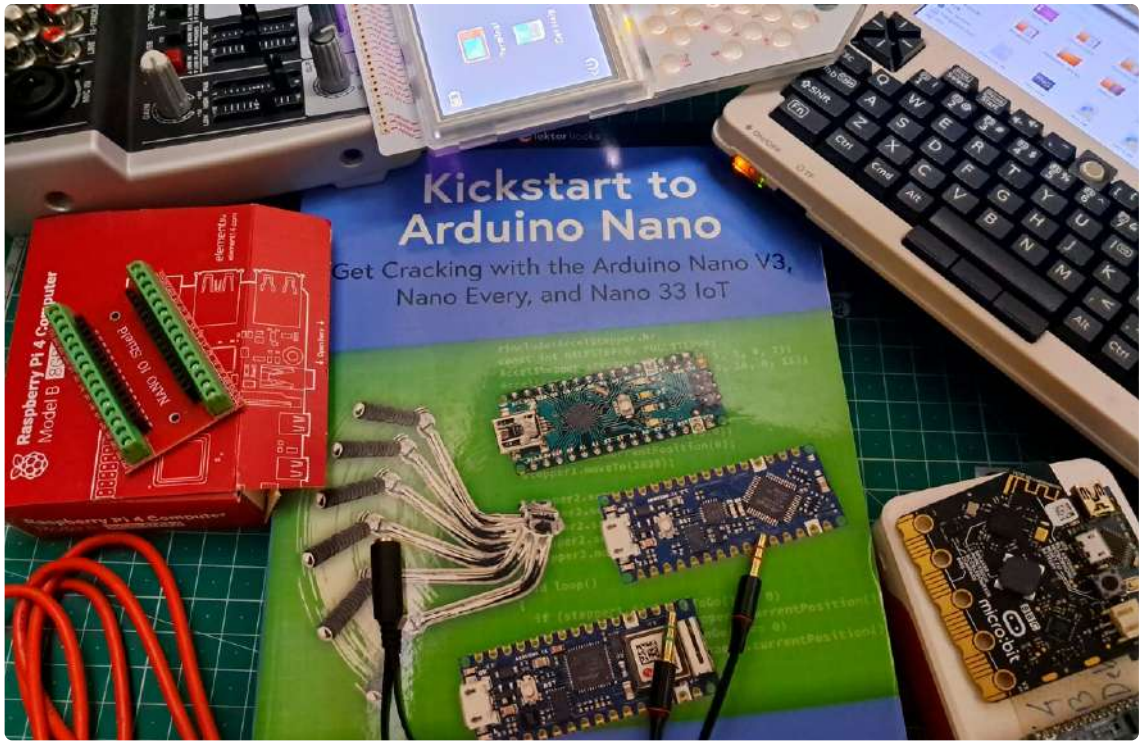
Alina : vous mentionnez dans votre livre que vous étiez en Inde, où vous résidez actuellement, pendant l'écriture. Comment le livre a-t-il pris vie pendant la pandémie de COVID-19 ?

Ashwin : la pandémie de COVID-19 a été une période de test pour toute notre civilisation. L'Inde a été l'une des nations les plus touchées, avec un taux d'hospitalisation et un nombre de décès parmi les plus élevés. En bref, Nashik, ma ville de résidence actuelle, a connu le plus grand nombre de décès en Inde. C'était très perturbant, car personne n'avait jamais connu une telle catastrophe auparavant. Il y avait constamment des fermetures, des couvre-feux et des restrictions. Il y avait toujours une pénurie de nourriture, de médicaments et d'autres fournitures. Il n'y avait pas de lits disponibles dans les hôpitaux. Les services d'ambulance et de corbillard étaient débordés. Même moi, j'ai contracté la maladie deux fois. La première fois, j'ai été admis dans un hôpital public et la seconde, j'ai été soigné à la maison. Je me souviens très bien avoir couru partout pour trouver des médicaments anti-grippe pour moi et mes voisins, car il y en avait très peu.

L'écriture du livre m'a donné un but pendant cette période sombre. En raison de l'isolement et de l'arrêt presque total de tous les autres aspects de la vie, j'ai eu l'occasion de me concentrer là-dessus. De plus, l'équipe d'Elektor m'a beaucoup aidé et guidé à chaque étape. Franchement, je ne trouve pas les mots pour exprimer ma sincère gratitude envers eux pour le grand soutien qu'ils ont toujours apporté.

Alina : vous êtes très impliqué dans l'enseignement en ligne, notamment via Udemy. Pouvez-vous nous décrire ce type d'expérience ? Comment les étudiants interagissent-ils avec vous par rapport à une classe traditionnelle ?

Ashwin : je préfère enseigner en ligne, car cela augmente ma portée. J'ai également organisé des ateliers de formation à la programmation pour des audiences en direct dans le passé. L'enseignement en ligne me permet de toucher des centaines de milliers d'étudiants. Les gens peuvent interagir avec moi en posant des questions sur le portail. Et comme tous les cours sont autodidactes, la participation des étudiants est beaucoup plus élevée.





L'espace de travail d'Ashwin Pajankar comporte divers éléments.

Alina : quand avez-vous réalisé que vous aimiez enseigner à d'autres personnes vos sujets préférés ?

Ashwin : pendant les vacances d'été de mes études d'ingénieur, j'avais l'habitude d'enseigner aux élèves de terminale et de les aider à se préparer aux examens d'entrée en ingénierie. J'enseignais les mathématiques, la physique, l'informatique et l'anglais. C'est à ce moment-là que j'ai réalisé que j'aimais enseigner et rendre les connaissances plus accessibles.

Alina : en fin de compte, quelle est la chose la plus importante que vous aimeriez que vos lecteurs apprennent ?

Ashwin : la chose la plus importante à retenir pour mes lecteurs est d'apprendre à explorer le monde de la technologie par eux-mêmes. C'est très facile de naviguer à travers une fois que vous savez où chercher les connaissances pertinentes. Pour en savoir plus, les lecteurs doivent parcourir eux-mêmes toute la documentation en ligne, les exemples de code et les divers forums de discussion.

Alina : travaillez-vous actuellement sur quelque chose ? De nouvelles idées pour un livre ?

Ashwin : oui. J'ai travaillé avec le Raspberry Pi 4. Pendant le confinement, il y avait une pénurie d'approvisionnement. Cependant, la situation s'est améliorée récemment, et j'ai pu me procurer un Raspberry Pi 4 avec 8 GB de RAM. J'ai fait des expériences avec lui. Une fois que j'aurai terminé le projet sur lequel je travaille, je serai prêt à écrire un livre sur Raspberry Pi pour Elektor. J'aime leur format de livre, et leur équipe est la meilleure en matière de conception. Je prévois d'écrire de nombreux autres livres techniques sur des sujets divers avec Elektor.

Alina : pouvez-vous nous parler des projets personnels que vous avez actuellement en cours ?

Ashwin : actuellement, je me suis lancé dans le grand projet de la refonte de ma chaîne YouTube et de l'élargissement de son audience. Je vais bientôt mettre en ligne de nombreuses vidéos enseignant les mathématiques de niveau scolaire en hindi (une langue parlée par un milliard de personnes dans le monde). Après cela, je couvrirai la physique de base et le génie électronique (encore une fois, en hindi).

Alina : y a-t-il une réalisation ou une contribution dont vous êtes le plus fier ?

Ashwin : les mathématiques, la programmation et l'électronique sont les choses que je fais pour gagner ma vie. Cependant, en dehors de mon travail, j'ai toujours participé activement à apporter à la communauté par le biais des programmes de sensibilisation sociale sur mes lieux de travail et dans les universités où j'ai étudié. Depuis que j'ai commencé à travailler en tant que freelance, je le fais à titre personnel. Grâce à ma participation à des activités éducatives pour les enfants défavorisés dans le cadre du programme d'action sociale de mon université, IIIT-H, mon interview a été publiée dans l'un des journaux les plus importants d'Hyderabad. J'ai reçu plusieurs prix pour avoir servi la communauté par le biais de l'action sociale. C'est l'aspect de ma vie dont je suis le plus fier.

220626-04 — VF : Maxime Valens

Questions ou commentaires ?

Si vous avez des questions sur cet entretien, n'hésitez pas à envoyer un courriel à Alina Neacsu à l'adresse alina.neacsu@elektor.com ou à l'équipe de rédaction d'Elektor à l'adresse editor@elektor.com.

Produits

- **Livre en anglais « *Kickstart to Arduino Nano* », A. Pajankar, (Elektor 2022)**
  Version papier
  www.elektor.fr/20241
  Version numérique
  www.elektor.fr/20242
- **Arduino Nano Every avec des connecteurs (SKU 19939)**
  www.elektor.fr/19939

jauge de longueur à **lidar**

mesure jusqu'à 12 m

Somnath Bera (Inde)

Les lidars sont d'excellents appareils pour détecter les obstacles et mesurer les distances. Dans cet article, nous en utilisons un pour prendre des mesures de distance précises jusqu'à 12 m avec une résolution de 1 cm.

Mon idée initiale en achetant un lidar TFMini-S était de construire une sorte de radar avec. Mais, comme souvent, je n'ai rien fait avec ce module jusqu'à ce que je me sois retrouvé face à une situation éclairante dans mon boulot.

Les tubes soudés par résistance électrique (ERW) sont de longs tubes d'un diamètre de 300 mm ou plus et d'une longueur pouvant atteindre douze mètres. Ils sont utilisés dans notre centrale électrique pour le transport de bouillie de cendres. Un jour, alors que je manipulais ces longs tubes pour l'assemblage des conduites d'évacuation de bouillie de cendres, j'ai constaté que mon équipe devait précisément mesurer chaque tube avant de les faire sortir par la porte. Chaque jour, quelqu'un doit consacrer du temps à effectuer ces mesures au moins 30 à 50 fois. Tenir le mètre à ruban à une extrémité avec un superviseur à l'autre, puis noter la longueur, en plein air, sous un soleil brûlant de 47 °C est un travail fastidieux. Ainsi, mon idée de créer une jauge de mesure de précision est née.

Modules lidar

Le capteur de mesure lidar à point unique TFMini-S (**figure 1**) permet d'effectuer des mesures de distance très rapidement (100 Hz) et très précisément (résolution de 1 cm, précision de 1 %) jusqu'à 12 m, ce qui convient parfaitement à ce type de tâche. Il existe également une version Plus, plus rapide (1 kHz) et dotée d'un boîtier conforme au standard IP65. Les deux versions sont équipées d'une interface série.

En plaçant le lidar à une extrémité du tube et en le dirigeant ensuite vers la plaque de bord à l'autre extrémité, nous obtiendrons une longueur précise du tube. Les deux versions du lidar ont un champ de vision (FOV) de 3,4 degrés ; il est donc très important de le pointer correctement pour obtenir des lectures correctes. Pour faciliter cette tâche, nous avons ajouté à notre système un pointeur laser actionné par un bouton-poussoir.

J'ai utilisé un module ESP32 pour contrôler le lidar et pour faire les calculs. J'ai ajouté un petit écran OLED pour afficher les mesures de distance. Intégrer une moyenne mobile de quelques mesures augmente la précision. Ensuite, il est possible d'imprimer le résultat sous la forme



Figure 1. Le lidar TFMini-S à faible coût a une portée de 12 m avec une résolution de 1 cm.

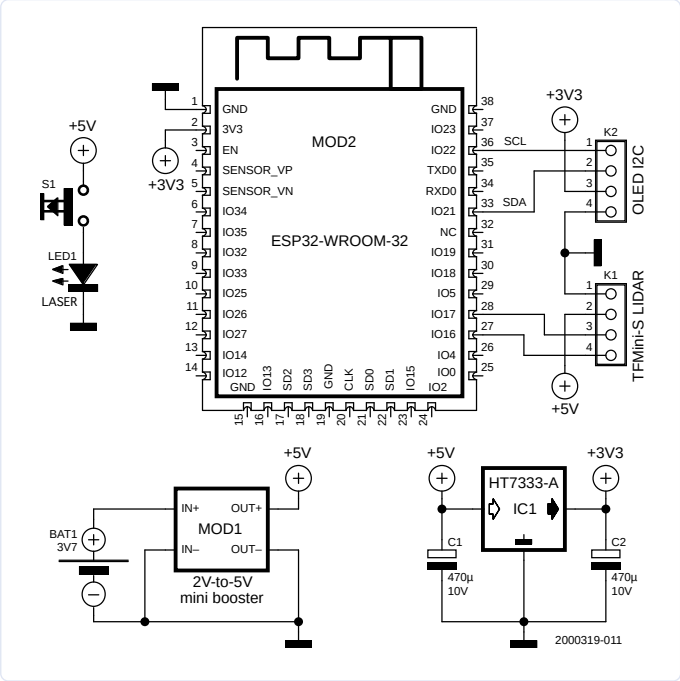


Figure 2. Le prototype utilise un module ESP32-WROOM qui ne dispose pas d'un régulateur 5 V intégré. Si vous utilisez un module DevMod-C ou Pico-Kit, vous pouvez omettre IC1. Dans ce cas, connectez l'alimentation 5 V à l'entrée 5 V du module, et non à sa broche 3.3 V !

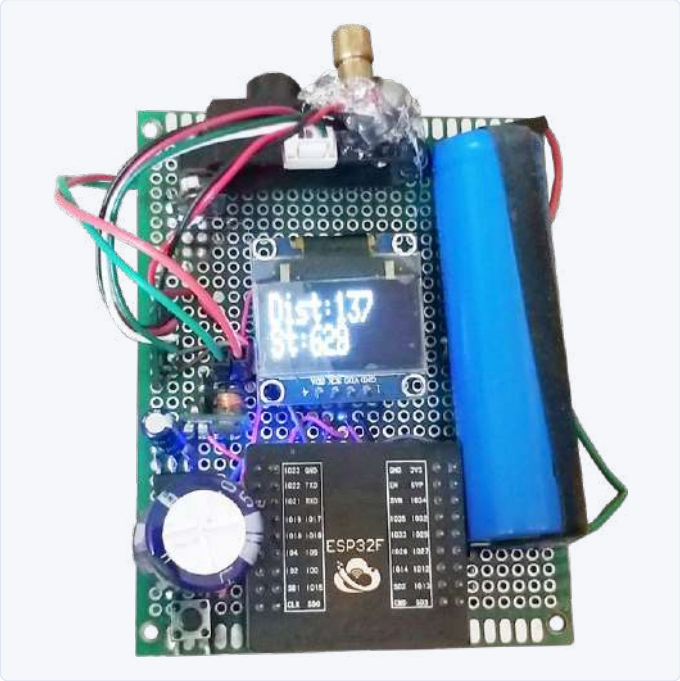


Figure 3. Le prototype de la jauge de précision basée sur le lidar a été construit sur une plaque de prototypage à usage général. Notez comment le pointeur laser est collé à chaud sur le module lidar.

d'une valeur à virgule flottante. Mais, pour plus de rapidité et de simplicité, je l'ai gardé comme un simple nombre entier.

Les lidars TFMini utilisent la réflexion de la lumière laser infrarouge invisible sur la surface de l'objet qu'ils ciblent. Si l'objet ne reflète pas, absorbe complètement ou défléchit la lumière laser entrante, la lecture sera erronée. Par exemple, évitez l'eau et les fenêtres en verre inclinées et très réfléchissantes. La plupart des autres objets réfléchissent suffisamment de lumière et les relevés sont pris facilement, même lorsque l'objet est en mouvement.

Construction de la jauge de distance

Lors de la construction de l'appareil, je devais veiller à ce qu'il soit portable pour permettre à mon équipe de l'utiliser sur le terrain. Le dispositif doit donc être alimenté par des piles. Le circuit réalisé est illustré à la **figure 2**. Le lidar et la diode laser que j'ai utilisée pour le pointage fonctionnent à 5 V. Le module ESP32 WROOM ne nécessite que 3,3 V. J'ai donc ajouté un mini convertisseur élévateur qui génère une alimentation de 5 V à partir d'une tension d'entrée aussi faible que 2 V. J'ai connecté un régulateur de tension HT7333-A à faible chute à la sortie de 5 V pour fournir 3,3 V à l'ESP32. Si vous utilisez un module ESP32 avec un régulateur de tension intégré, vous n'avez pas besoin du HT7333-A. Le circuit complet fonctionne avec deux piles de 1,5 V ou avec une pile LiPo de 3,7 V. Notez que le HT7333-A est un composant monté en surface (CMS) ; soyez donc prudent lors de son montage.

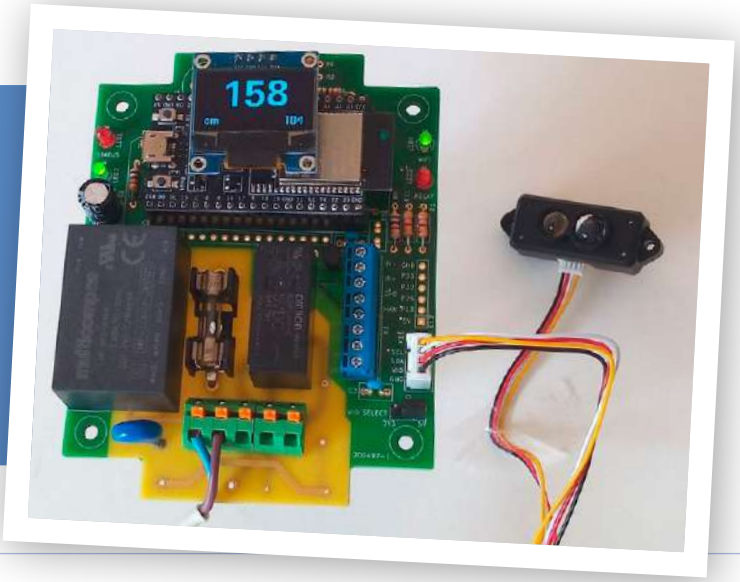
À propos du programme

Le programme que j'ai écrit pour le projet est un croquis Arduino que vous pouvez télécharger à partir de [1]. Il fait appel à la bibliothèque TFMini et aux bibliothèques GFX SSD1306 d'Adafruit disponibles dans le gestionnaire de bibliothèques de l'EDI Arduino.

Le lidar et l'ESP32 communiquent via un port série (Serial1). Tous les 25 ms, une nouvelle mesure est demandée. Toute la tâche est effectuée par la bibliothèque TFMini tandis que le croquis offre simplement une interface utilisateur. Les valeurs de distance et d'intensité du signal sont affichées sur l'écran OLED et sont également transmises sur le port série d'Arduino (Serial0).

Utilisez l'Automator !

L'Automator d'Elektor [2] est la plateforme parfaite pour expérimenter avec le lidar TFMini-S. Basé sur un module ESP32, il est doté d'un connecteur Grove compatible avec le module lidar, il dispose d'un écran OLED et des alimentations nécessaires. Nous (chez Elektor Labs) avons donc créé un croquis Arduino pour ce projet qui fonctionne sur l'Automator. Il utilise les bibliothèques TFLidar et U8g2, disponibles dans le gestionnaire de bibliothèques de l'EDI Arduino. Le programme peut facilement être développé pour ajouter la connectivité wifi ou Bluetooth et pour contrôler le relais et les LED de l'Automator. Il peut être téléchargé à partir de [2].



Utilisation de l'appareil

Après avoir monté tous les composants sur un circuit imprimé (**figure 3**), positionnez le lidar sur le bord de la carte pour ce qu'il ait une vue précise de l'objet. Le pointeur laser est une diode laser rouge ordinaire commandée par un bouton poussoir. Si possible, fixez-le sur le lidar avec de la colle chaude (ou une substance similaire) de façon à ce qu'il pointe dans la même direction.

La distance minimale que le lidar TFMini-S peut atteindre est de 30 cm. Par conséquent, pointez-le vers un objet ou un mur dans une plage de 30 cm à 12 m. Mesurez la distance et vérifiez-la avec un mètre à ruban. Vous finirez par constater que la lecture est précise à un centimètre près.

La deuxième ligne de l'écran indique l'intensité du signal reçu. Au fur et à mesure que l'objet s'éloigne, l'intensité diminue.

200316-04 – VF : Asma Adhimi

Des questions, des commentaires ?

Contactez Elektor (redaction@elektor.fr).

PRODUITS

- **Lidar TFMini-S (SKU 19691)**
  www.elektor.fr/19691
- **Lidar TFmini Plus IP65(SKU 19690)**
  www.elektor.fr/19690
- **ESP32-WROOM-32 (SKU 18421)**
  www.elektor.fr/18421
- **Écran OLED de 0,96 pouce (Bleu, I²C, 4-Pin) (SKU 18747)**
  www.elektor.fr/18747

LIENS

[1] Ce projet sur Elektor Labs : www.elektormagazine.fr/labs/tfmini-lidar-precision-gauge-30-cm-to-1200-cm
[2] L'Automator d'Elektor : www.elektormagazine.fr/labs/automator

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

design > share ; earn



les signaux audios et l'ESP32

l'environnement ESP-ADF en pratique

Tam Hanna (Hongrie)

Le développement de tout nouvel appareil électronique grand public constitue incontestablement un défi. Ces produits ont en général un cycle de vie extrêmement court, de sorte que le temps consacré au développement est primordial pour le succès. Ceci est particulièrement vrai pour les applications audios où on utilise des algorithmes pour mettre en œuvre diverses fonctions de codecs standard. L'*Espressif Audio Development Framework* (ESP-ADF) est un outil puissant avec des ressources qui permettront aux développeurs de telles applications de gagner du temps et de l'énergie.

L'Espressif Audio Development Framework fournit une collection logicielle d'algorithmes et de codecs qui utilisent un format standardisé. Pour développer une application, le concepteur n'a qu'à relier en séquence les « circuits logiciels » appropriés sans avoir à s'occuper de leurs détails internes individuels.

Cet article a pour but d'expliquer les bases du fonctionnement pratique de l'ADF. J'ai beaucoup utilisé l'outil ADF d'Espressif pour m'aider à mettre en œuvre des solutions conformes aux exigences de conception de mes clients. Je peux maintenant partager avec vous beaucoup de choses que j'aurais aimé connaître lorsque j'ai commencé à utiliser l'ADF.

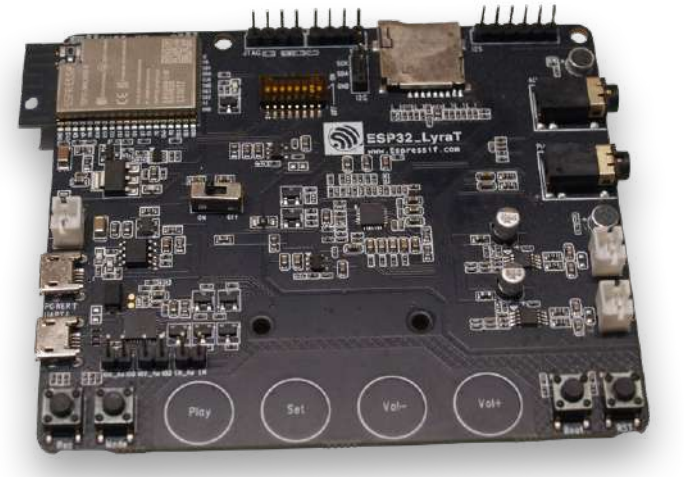


Figure 1. Le petit circuit imprimé noir comprend tout...

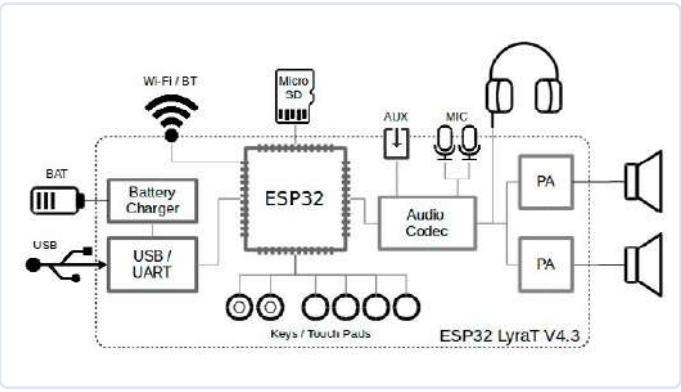


Figure 2. ... ce dont vous avez besoin pour prototyper des applications audio (source : [6]).

Matériel nécessaire

Soyons clairs : l'ESP-ADF est compatible avec tout système à base d'ESP32. Pour la « partie audio », la bibliothèque offre une interface de pilote standardisée qui prend en charge les données. Avant de finaliser la conception matérielle, une grande partie du logiciel peut être écrite à l'avance et testée sur une carte de développement



ordinaire. Espressif propose une gamme de cartes de ce type et pour mes développements maison j'apprécie la carte LyraT présentée dans les **figures 1** et **2**.

Le plus important est le signal analogique en « bande de base » numérisé, fourni par la puce ES8388 (avec deux microphones analogiques). La communication se fait à la fois via I2S et via un groupe de broches dédiées – le schéma de circuit et les interconnexions sont fournis par Espressif et peuvent être facilement reproduits pour la conception du dispositif final. La puce se trouve sans peine auprès de UTSource et LCSC.

Lorsqu'on travaille avec la LyraT, il ne reste pas beaucoup de GPIO disponibles sur l'ESP32 embarqué. Lorsque la conception du matériel de l'appareil devient plus complexe, vous pouvez rapidement manquer d'options d'interface. Dans ce cas, il peut être nécessaire d'envisager l'utilisation d'un second processeur dédié à la prise en charge de la communication matérielle.

Installation de l'ADF

Espressif fournit l'ADF comme un greffon pour son environnement IDF existant. L'ADF ne peut pas être utilisé dans Arduino, et n'est même pas supporté par certaines versions de l'IDF d'Espressif.

La façon la plus pratique d'installer l'environnement de travail est de télécharger le dépôt *ESP-ADF* complet. Je commence par les commandes suivantes pour créer un sous-dossier, *esp4*, où le dépôt sera stocké localement :

```
(base)tamhan@tamhan-thinkpad:~$ mkdir esp4
(base)tamhan@tamhan-thinkpad:~$ cd esp4/
```

Le dépôt complet peut maintenant être téléchargé en utilisant le client en ligne de commande de Git :

```
(base)tamhan@tamhan-thinkpad:~/esp4$ git clone
--recursive https://github.com/espressif/esp-adf.git
Cloning into 'esp-adf'...
```

Une inspection minutieuse de la sortie (non montrée ici) révèle que le dépôt ADF comporte plusieurs références externes – pour cette raison, un téléchargement via la fonction de téléchargement du navigateur intégrée à GitHub n'est pas possible.

Une version de l'IDF d'Espressif est incluse avec l'ADF. Certains des composants nécessitent la présence de la variable d'environnement `ADF_PATH`. Si vous souhaitez utiliser une fenêtre de ligne de commande pour traiter les applications basées sur l'ADF, vous devez définir la variable d'environnement à l'aide des commandes suivantes :

```
(base)tamhan@tamhan-thinkpad:~/esp4$ cd esp-adf
(base)tamhan@tamhan-thinkpad:~/esp4/esp-adf$ export
ADF_PATH=$PWD
```

La présence de la variable `ADF_PATH` peut affecter les projets IDF normaux : j'évite généralement cela en utilisant une fenêtre de terminal différente pour le travail « normal » sur ESP32.

Pour la version IDF intégrée, il faut exécuter l'installation habituelle à l'étape suivante pour obtenir les compilateurs et autres dépendances :

```
(base)tamhan@tamhan-thinkpad:~/esp4/esp-adf$ cd
$ADF_PATH/esp-idf
(base)tamhan@tamhan-thinkpad:~/esp4/esp-adf/esp-idf$ ./
install.sh
Detecting the Python interpreter
. . .All done! You can now run:

  . ./export.sh
```

Il faut alors appeler un script pour la configuration de la chaîne d'outils (les deux points consécutifs ne sont pas une faute de frappe) :

```
(base)tamhan@tamhan-thinkpad:~/esp4/esp-adf/esp-idf$ .
./export.sh
```

En pratique, il est conseillé à ce stade de créer un script shell pour le paramétrage – avec Bash, il peut alors se configurer automatiquement à la demande.

Architecture logicielle : le pipeline

Après avoir téléchargé et installé l'ADF, il est temps de commencer à se pencher sur sa structure théorique. Fidèle au concept de « circuit logiciel » développé à l'origine par The Stepstone Corporation, l'ADF consiste aussi pour le développeur à essentiellement mettre en œuvre un pipeline constitué d'une séquence d'étapes de traitement. À titre d'illustration, la **figure 3** présente un modèle de flux de travail pour l'implémentation d'un lecteur MP3.

Espressif ne semble pas trop s'étendre sur les éléments concrets du pipeline – la **figure 4** montre les rôles et exemples de mise en œuvre proposés.

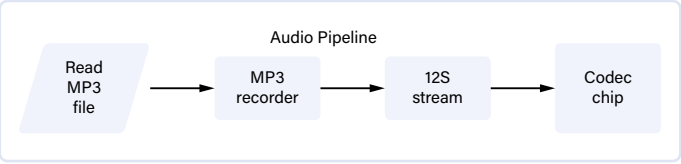


Figure 3. Ce pipeline décode les MP3 (source : [3]).

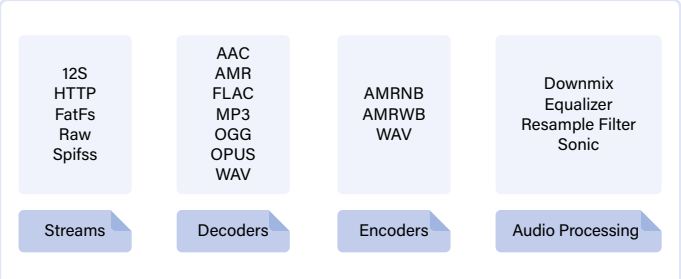


Figure 4. Les éléments des pipelines peuvent être divisés en plusieurs types (source : [3]).

La seule particularité du système actuel est que le concept Unix selon lequel « tout est fichier » s'applique aussi aux périphériques. L'API, documentée en détail sous [1] et dont la compréhension est facilitée par les fonctions de bouton fournies sous [2], permet la création d'« enveloppes » qu'on peut directement intégrer dans le pipeline. Pour les développeurs d'applications, ce n'est toutefois pertinent que dans la mesure où l'idée de considérer également les périphériques comme des éléments du pipeline peut nécessiter une réflexion approfondie.

La documentation de l'API pour les différents éléments du pipeline se trouve sous [3].

Architecture logicielle : construire un lecteur MP3

Pour mieux comprendre, examinons un des exemples fournis par Espressif : Si vous souhaitez mettre en œuvre un projet quelconque à l'aide de l'ADF, vous avez tout intérêt à commencer par rechercher un modèle similaire ou du moins apparenté parmi les exemples. Naviguez vers le répertoire des exemples en utilisant la commande suivante :

```
cd $ADF_PATH/examples/
```

Nous allons voir dans les prochaines étapes comment construire un lecteur MP3 classique. Allez dans le répertoire *~/esp4/esp-adf/examples/get-started/play_mp3_control/main* et ouvrez le fichier *play_mp3_control_example.c* avec votre éditeur préféré.

Le plus important est le point d'entrée, qui montre la création de certaines variables membres. Notre exemple a besoin d'un objet pipeline et deux descripteurs sont nécessaires pour les éléments individuels :

```
void app_main(void)
{
   audio_pipeline_handle_t pipeline;
   audio_element_handle_t i2s_stream_writer, mp3_decoder;
...
```

Le matériel est pris en compte dans l'ADF sous forme d'une abstraction intitulée *Board*. Votre sélection ou paramétrage s'effectue dans le cadre de *Menuconfig*. Les commandes à créer dans le code associé évaluent principalement les constantes de compilation :

```
   audio_board_handle_t board_handle = audio_board_init();
   audio_hal_ctrl_codec(board_handle->audio_hal,
   AUDIO_HAL_CODEC_MODE_BOTH, AUDIO_HAL_CTRL_START);
. . .
   int player_volume;
   audio_hal_get_volume(board_handle->audio_hal,
    &player_volume);
```

Il est intéressant de noter que le HAL contient aussi les paramètres de commande du volume. L'étape suivante est la configuration de l'objet principal du pipeline :

```
   audio_pipeline_cfg_t pipeline_cfg =
    DEFAULT_AUDIO_PIPELINE_CONFIG();
   pipeline = audio_pipeline_init(&pipeline_cfg);
   mem_assert(pipeline);
```

Les développeurs ayant une expérience antérieure d'ESP-IDF trouveront une grande partie de ces éléments familiers. L'objet pipeline est créé en passant une structure de configuration, qui peut ensuite être transmise d'un utilisateur à l'autre.

Vient ensuite la génération des éléments réels du pipeline. Le premier élément est le décodeur MP3 :

```
   mp3_decoder_cfg_t mp3_cfg =
    DEFAULT_MP3_DECODER_CONFIG();
   mp3_decoder = mp3_decoder_init(&mp3_cfg);
   audio_element_set_read_cb(mp3_decoder,
    mp3_music_read_cb, NULL);
```

La question qui se pose maintenant (surtout au vu du pipeline présenté à la **figure 3**) est de savoir comment les données seront fournies. La réponse à cette question est la méthode `audio_element_set_read_cb`, qui reçoit en paramètre une fonction (structurée selon le schéma suivant) :

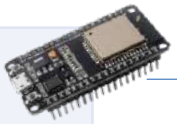
```
int mp3_music_read_cb(audio_element_handle_t el,
        char *buf, int len,
        TickType_t wait_time, void *ctx) {
   int read_size = file_marker.end
      - file_marker.start - file_marker.pos;
   if (read_size == 0) {
       return AEL_IO_DONE;
   } else if (len < read_size) {
       read_size = len;
   }
   memcpy(buf, file_marker.start +
    file_marker.pos, read_size);
   file_marker.pos += read_size;
   return read_size;
}
```

La fonction de rappel fournit les informations à traiter par le codec via le tampon. Le flux I2S est alors simplifié ; il provient de `AUDIO_STREAM_WRITER` et paramétré en sortie afin que les informations entrantes soient dirigées vers le matériel de sonorisation I2S :

```
i2s_stream_cfg_t i2s_cfg = I2S_STREAM_CFG_DEFAULT();
i2s_cfg.type = AUDIO_STREAM_WRITER;
i2s_stream_writer = i2s_stream_init(&i2s_cfg);
```

La création d'éléments de pipeline ne les déclare pas automatiquement : Il s'agit d'une facilité de l'ESP-ADF pour les développeurs qui veulent utiliser plusieurs pipelines en même temps. La configuration du pipeline se fait à la place par la déclaration des différents éléments, chacun d'entre eux possédant aussi une chaîne qui sert d'identifiant (ID) :

```
audio_pipeline_register(pipeline, mp3_decoder, "mp3");
```



Listage 1. Traitement des événements du lecteur MP3.

```
while (1) {
    audio_event_iface_msg_t msg;
    esp_err_t ret = audio_event_iface_listen(evt, &msg, portMAX_DELAY);
    if (ret != ESP_OK) {
        continue;
    }
    if (msg.source_type == AUDIO_ELEMENT_TYPE_ELEMENT && msg.source == (void *) mp3_decoder
                && msg.cmd == AEL_MSG_CMD_REPORT_MUSIC_INFO) {
        audio_element_info_t music_info = ;
        audio_element_getinfo(mp3_decoder, &music_info);
        ESP_LOGI(TAG, "[ * ] Receive music info from mp3 decoder, sample_rates=%d, bits=%d, ch=%d",
                music_info.sample_rates, music_info.bits, music_info.channels);
        audio_element_setinfo(i2s_stream_writer, &music_info);
        i2s_stream_set_clk(i2s_stream_writer, music_info.sample_rates,
                music_info.bits, music_info.channels);
        continue;
    }
}
```

```
audio_pipeline_register(pipeline, i2s_stream_writer,
"i2s");
const char *link_tag[2] = {"mp3", "i2s"};
audio_pipeline_link(pipeline, &link_tag[0], 2);
```

Dans le deuxième acte, le pipeline est alors constitué à l'aide d'un tableau qui fournit les ID des chaînes individuelles dans le bon ordre. Pour le lecteur MP3, seule l'activation du pipeline manque à ce stade :

```
ESP_LOGI(TAG, "[ 5.1 ] Start audio_pipeline");
. . .
audio_pipeline_run(pipeline);
```

Dans le cas du lecteur MP3, il reste un traitement relativement important pour réagir à divers événements (**listage 1**). À ce stade, le traitement exact des événements n'est pas pertinent – il est plus intéressant d'exécuter le processus de compilation à titre de test :

```
(base)tamhan@tamhan-thinkpad:~/esp4/esp-adf/examples/
get-started/play_mp3_control$ make build
```

Une remarque à ce sujet : Pour garder les données de mes clients professionnels séparées et sécurisées, j'écris cet article sur mon ordinateur portable de voyage. La version Ubuntu utilisée ici utilise la version 3.4.3 de CMake, c'est pourquoi il n'est pas possible d'exécuter idf.py. En pratique, idf.py est bien sûr toujours préférable à l'utilisation de `make` et sera également obligatoire dans les futures versions de l'IDF.

Avec un squelette de projet impeccable, la récompense de l'effort est l'écran *menuconfig*, qui offre quelques options supplémentaires par rapport à un projet IDF normal. La plus importante est l'option *(Top) Audio HAL Audio board*, où vous pouvez sélectionner la configuration de carte appropriée, comme le montre la **figure 5**.

Après la sauvegarde, le processus de construction commence de la même manière que pour tout projet IDF. Il est particulièrement important que la carte dispose de deux ports Micro-USB : le port POWER est utilisé pour alimenter la carte, tandis que le port UART est uniquement utilisé pour la communication de données. Normalement, j'utilise simplement deux câbles USB séparés pour connecter la station de travail à la carte.

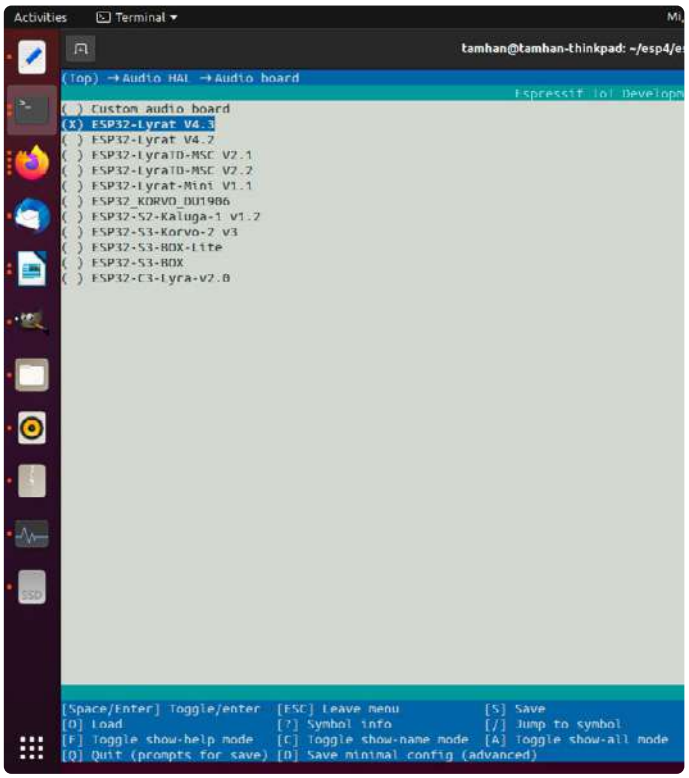


Figure 5. ADF étend Menuconfig avec plusieurs paramètres.

Une chose à noter à propos de la carte LyraT est que (contrairement à de nombreuses autres cartes de développement) elle ne peut pas être reprogrammée automatiquement ; avant même qu'une connexion puisse être établie, vous devez maintenir le bouton poussoir *Boot* enfoncé tout en appuyant brièvement sur RST. Vous devriez alors voir :

```
Serial port /dev/ttyUSB0
Connecting.....................................
```

Si vous vous êtes trompé dans la séquence de boutons-poussoirs, le système interprète qu'il s'agit d'un message de communication corrompu… essayez à nouveau :

```
A fatal error occurred: Failed to connect to ESP32:
Invalid head of packet (0x1B): Possible serial noise or
corruption.
```

Une fois le processus de flashage terminé avec succès, vous pouvez brancher un casque dans la prise phono et appuyer à nouveau sur *Reset*. Si tout est en ordre, vous devriez entendre une mélodie.

Enfin, il faut rappeler que l'exécution du script de configuration export.sh ne suffit pas à paramétrer complètement l'environnement. Avant de l'activer, il est toujours nécessaire de paramétrer correctement la variable d'environnement *ADF_PATH* :

```
(base)tamhan@tamhan-thinkpad:~/esp4/esp-adf$ export
ADF_PATH=$PWD
(base)tamhan@tamhan-thinkpad:~/esp4/esp-adf/esp-idf$ .
./export.sh
Setting IDF_PATH to '/home/tamhan/esp4/esp-adf/esp-idf'
. . .
```

Traitement des données d'entrée

L'utilisation du pipeline ESP-ADF n'a pas de sens que dans le contexte de la lecture de médias : il est tout aussi légitime d'y intégrer des calculs. Par exemple, l'un de mes récents travaux pour un client nécessitait le traitement de données microphoniques par un algorithme. Le calcul proprement dit a été placé dans un élément du pipeline, dont je voudrais vous montrer brièvement la structure générale.

Nous commençons par initialiser le pipeline, et puisque les informations audios arrivent via le bus I2S, un élément de flux I2S est à nouveau requis dans la première étape. Ici, cependant, sa configuration inclut maintenant l'indicateur AUDIO_STREAM_READER, qui le marque comme une entrée ou une source de données :

```
i2s_stream_cfg_t i2s_cfg = I2S_STREAM_CFG_DEFAULT();
i2s_cfg.type = AUDIO_STREAM_READER;
#if defined CONFIG_ESP_LYRAT_MINI_V1_1_BOARD
    i2s_cfg.i2s_port = 1;
#endif
i2s_stream_reader = i2s_stream_init(&i2s_cfg);
```

Outre le flux I2S, li nous faut aussi son compagnon, qui encapsule l'algorithme propriétaire. Voici son initialisation :

```
tams_stream_cfg_t fatfs_cfg = TAMS_STREAM_CFG_DEFAULT();
fatfs_cfg.type = AUDIO_STREAM_WRITER;
tams_stream = tams_stream_init(&fatfs_cfg);
```

La construction proprement dite du pipeline se fait ensuite en assignant des chaînes et en les passant à `audio_pipeline_link()`:

```
audio_pipeline_register(pipeline, i2s_stream_reader,
"i2s");
audio_pipeline_register(pipeline, tams_stream, "tam");
audio_pipeline_link(pipeline, (const char *[]) {"i2s",
"tam"}, 2);
```

Avec cela, nous pouvons passer au fichier d'en-tête qui fournit les éléments nécessaires à la mise en service de la tâche « Tam ». Le plus important est la structure de configuration, qui contient, entre autres, les méta-informations requises pour FreeRTOS (**listage 2**).

`TAMS_STREAM_CFG_DEFAULT` est une macro de commodité qui permet la création des structures avec les paramètres par défaut. Pour des raisons de place, nous ne pouvons pas imprimer les constantes ici :

Listage 2. Configuration des éléments du Pipeline.

```
typedef struct {
   audio_stream_type_t     type;          /*!< Stream type */
   int                     buf_sz;        /*!< Audio Element Buffer size */
   int                     out_rb_size;   /*!< Size of output ring buffer */
   int                     task_stack;    /*!< Task stack size */
   int                     task_core;     /*!< Task running in core (0 or 1) */
   int                     task_prio;     /*!< Task priority (based on freeRTOS priority) */
} tams_stream_cfg_t;
```

Listage 3. Initialisation.

```
audio_element_handle_t tams_stream_init(tams_stream_cfg_t *config)
{
    audio_element_handle_t el;
    tams_stream_t *fatfs = audio_calloc(1, sizeof(tams_stream_t));

    AUDIO_MEM_CHECK(TAG, fatfs, return NULL);

    audio_element_cfg_t cfg = DEFAULT_AUDIO_ELEMENT_CONFIG();
    cfg.open = _tams_open;
    cfg.close = _tams_close;
    cfg.process = _tams_process;
    cfg.destroy = _tams_destroy;
    cfg.task_stack = config->task_stack;
    cfg.task_prio = config->task_prio;
    cfg.task_core = config->task_core;
    cfg.out_rb_size = config->out_rb_size;
    cfg.buffer_len = config->buf_sz;
    if (cfg.buffer_len == 0) {
       cfg.buffer_len = TAMS_STREAM_BUF_SIZE;
    }
    cfg.tag = "file";
...
}
```

```
#define TAMS_STREAM_CFG_DEFAULT() {\
    .task_prio = TAMS_STREAM_TASK_PRIO, \
    . . .
}
```

La seule exception concerne `BUF_SIZE`. Les données fournies ont une largeur de 16 bits, c'est pourquoi nous avons besoin de la valeur constante suivante pour traiter les blocs d'une longueur de 1024 slots :

```
#define TAMS_STREAM_BUF_SIZE   (1024*2)
```

Avec cela, nous pouvons passer à l'initialisation, voir le **listage 3**.

La valeur de `cfg.buffer_len` détermine le nombre de mots de données à fournir par cycle. À part ça, il n'y a essentiellement que du code de gestion banal :

```
    if (config->type == AUDIO_STREAM_WRITER) {
        cfg.write = _tams_write;
    } else {
        cfg.read = _tams_read;
    }
    el = audio_element_init(&cfg);

    AUDIO_MEM_CHECK(TAG, el, goto _tams_init_exit);
    audio_element_setdata(el, fatfs);
    return el;
_tams_init_exit:
    audio_free(fatfs);
    return NULL;
```



Les éléments audios créés par le développeur doivent s'intégrer dans un pipeline, comme le montre la **figure 3**. Pour cela `audio_element_init()` a besoin de certaines informations : outre les paramètres liés au thread, la routine définit également un ensemble de fonctions de rappel que le superviseur audio peut utiliser pour informer l'élément des événements qui se produisent.

À titre d'exemple, examinons la méthode `tams_open` qui se charge de l'initialisation de l'élément selon le schéma suivant :

```
static esp_err_t _tams_open(audio_element_
handle_t self) {
    audio_element_info_t info;
    audio_element_getinfo(self, &info);

    initTamsWorkerAlgo();

     return audio_element_setinfo(self,
&info);
}
```

Comme le présent algorithme n'a pas besoin des informations contenues dans `audio_element_info_t`, l'implémentation se limite à transmettre au pipeline les informations sur les paramètres.

Les événements `read` et `write` sont responsables de l'échange de données. Notre flux n'est utilisé qu'en tant que « point d'écriture » et ne peut pas être lu, il ne fait donc qu'émettre un message d'erreur dans le journal du système lorsqu'une tentative de lecture est effectuée :

```
static int _tams_read(audio_element_handle_t self,
char *buffer, int len,
TickType_t ticks_to_wait, void *context)
{
    ESP_LOGE(TAG, "CENSORED");
    return 0;
}
```

Les opérations d'écriture sont aussi implémentées en « transparent » et n'effectuent aucune opération réelle :

```
static int _tams_write(audio_element_handle_t self,
char *buffer, int len,
TickType_t ticks_to_wait, void *context) {
    return len;
}
```

L'étape suivante nécessite un tampon d'entrée structuré selon le format suivant :

```
int16_t DspBuf[4096];
```

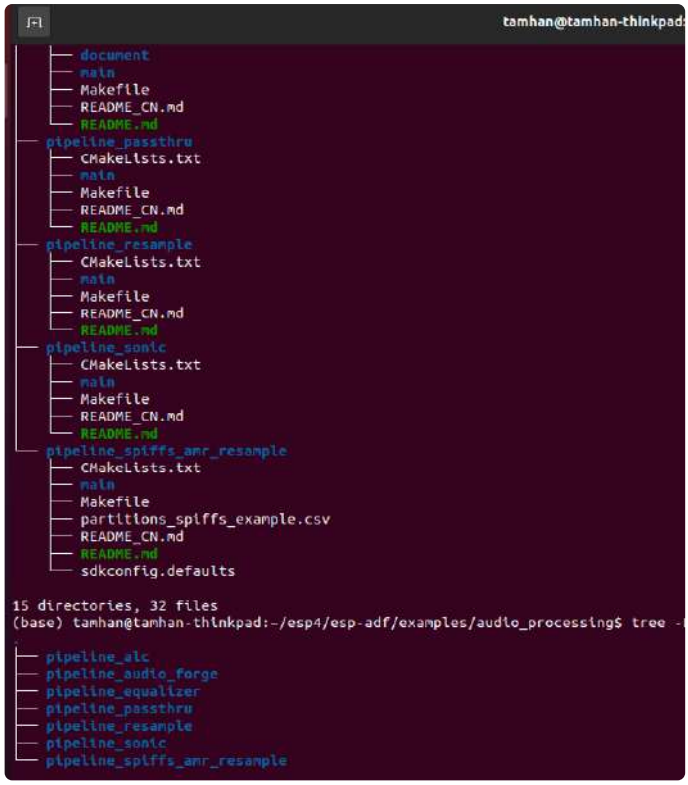


Figure 6. Tout un tas de pipelines à explorer !

`_tams_process()` se charge ensuite du traitement réel des données. La chose la plus importante ici est l'appel à `audio_element_input`, qui copie les informations dans le tampon de travail :

```
static int _tams_process(audio_element_handle_t self,
char *in_buffer, int in_len) {
    audio_element_input(self, (char *)DspBuf, in_len);
```

Comme exemple simple de traitement de données, j'ai inclus une simple opération de normalisation sur une plage de valeurs :

```
    for (int i=0 ; i< in_len ; i++) {
        y_cf[i] = ((float)DspBuf[i]) /  (float)32768;;
    }
```

Enfin, n'oubliez pas les utilités :

```
    int r_size = audio_element_input(self,
          in_buffer, in_len);
    int w_size = 0;
    if (r_size > 0) {
        w_size = audio_element_output(self,
          in_buffer, r_size);
    } else {
        w_size = r_size;
    }
    return w_size;
}
```

Opérations visant à libérer la mémoire précédemment affectée aux flux :

```
static esp_err_t _tams_close(audio_element_handle_t self)
{
    return ESP_OK;
}
static esp_err_t _tams_destroy(audio_element_handle_t
self) {
    tams_stream_t *fatfs =
    (tams_stream_t *)audio_element_getdata(self);
    audio_free(fatfs);
    return ESP_OK;
}
```

Le reste du code fonctionnel de la solution doit seulement arrêter le noyau en utilisant les instructions suivantes :

```
audio_pipeline_run(pipeline);
printf("halting.\n");
for(;;);
```

Notez que la fonction `worker` appelée périodiquement dans le pipeline peut faire ce qu'elle veut – dans ce projet, selon la configuration, elle communique avec des pairs I²C ou SPI en réponse aux données audios entrantes.

Plus d'éléments de pipelines

De nombreux autres exemples liés à l'audio existent dans l'ESP-ADF (**figure 6**). Allez dans *~/esp4/esp-adf/examples/audio_processing* – les algorithmes implémentent un grand nombre d'autres opérations en pipeline.

J'ai trouvé l'exemple *examples/audio_processing/pipeline_equalizer* particulièrement intéressant. Il implémente une fonction complète d'égaliseur graphique et est une classe incluse dans l'environnement ESP-ADF, donc ne nécessite pas de mathématiques avancées. Sa mise en œuvre s'effectue selon le schéma suivant, bien connu :

```
equalizer_cfg_t eq_cfg = DEFAULT_EQUALIZER_CONFIG();
int set_gain[] = {-13, -13, -13, -13, -13, -13, -13,
-13, -13, -13, -13, -13, -13, -13, -13, -13, -13, -13,
-13, -13};
eq_cfg.set_gain = set_gain;
equalizer = equalizer_init(&eq_cfg);
```

Pour en savoir plus sur les champs de données, consultez la rubrique [4].

L'intégration dans le flux de lecture s'effectue ensuite comme d'habitude avec `audio_pipeline_register()` :

```
audio_pipeline_register(pipeline, fatfs_stream_reader,
"file_read");
audio_pipeline_register(pipeline, wav_decoder, "wavdec");
audio_pipeline_register(pipeline,  equalizer,
"equalizer");
audio_pipeline_register(pipeline, i2s_stream_writer,
"i2s");
```

Lorsqu'une application atteint un certain degré de complexité, on arrive à un point où les traitements du signal faits maison ne suffisent plus. Espressif fournit ici de l'aide avec la bibliothèque



ESP-DSP disponible sous [5] : c'est une sorte de canevas qui fournit divers algorithmes DSP avec une grande stabilité algorithmique et diverses optimisations pour les différentes puces communes d'Espressif.

Grâce à l'organisation de sa structure de dossiers, chaque projet ESP32 est « directement » équipé pour accepter l'intégration de composants ESP-IDF supplémentaires. L'installation du DSP est donc facile. Dans un premier temps, créez une copie d'un projet :

```
(base)tamhan@tamhan-thinkpad:~/esp4/esp-adf/examples$
cp -r  audio_processing/pipeline_equalizer/ tamsdsptest1
(base)tamhan@tamhan-thinkpad:~/esp4/esp-adf/examples$
cd tamsdsptest1/
```

Créez-y un dossier portant le nom de *components*. Celui-ci accepte alors la version complète de la bibliothèque, qui peut être téléchargée sur GitHub :

```
(base)tamhan@tamhan-thinkpad:~/esp4/esp-adf/examples/
tamsdsptest1$ mkdir components
(base)tamhan@tamhan-thinkpad:~/esp4/esp-adf/examples/
tamsdsptest1$ cd components/
(base)tamhan@tamhan-thinkpad:~/esp4/esp-adf/examples/
tamsdsptest1/components$ git clone https://github.com/
espressif/esp-dsp.git
```

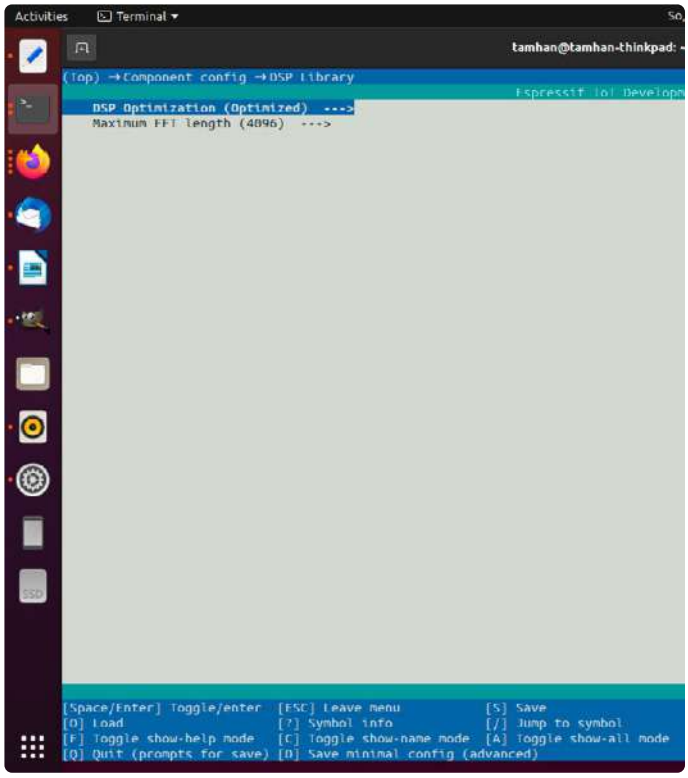


Figure 7. ESP-DSP s'imbrique sous (Top) → Component config → DSP Library dans le processus de compilation.

Lors de la prochaine exécution de la chaîne d'outils de compilation, vous verrez apparaître une nouvelle option permettant de configurer le comportement de la bibliothèque DSP, comme le montre la **figure 7.**

Résumé

Avec le support fourni par l'ESP-ADF, Espressif offre aux développeurs un environnement puissant et flexible pour simplifier la mise en œuvre d'applications audio. Grâce à lui, ma propre entreprise a déjà pu économiser plusieurs centaines d'heures de temps de développement, je peux fortement recommander son utilisation.

220600-04 — VF : Denis Lafourcade

Des questions, des commentaires ?

Envoyez un courriel à l'auteur à (tamhan@tamoggemon.com) ou contactez Elektor (redaction@elektor.fr).

À propos de l'auteur

L'ingénieur Tam Hanna est un développeur, auteur et journaliste indépendant (www.instagram.com/tam.hanna). Il développe des produits électroniques, des ordinateurs et des logiciels depuis plus de 20 ans maintenant. Pendant son temps libre, Tam conçoit et réalise des produits en impression 3D et, entre autres choses, se passionne pour le commerce et la dégustation de cigares haut de gamme.

Produits

› **ESP32-DevKitC-32D**
www.elektor.fr/18701

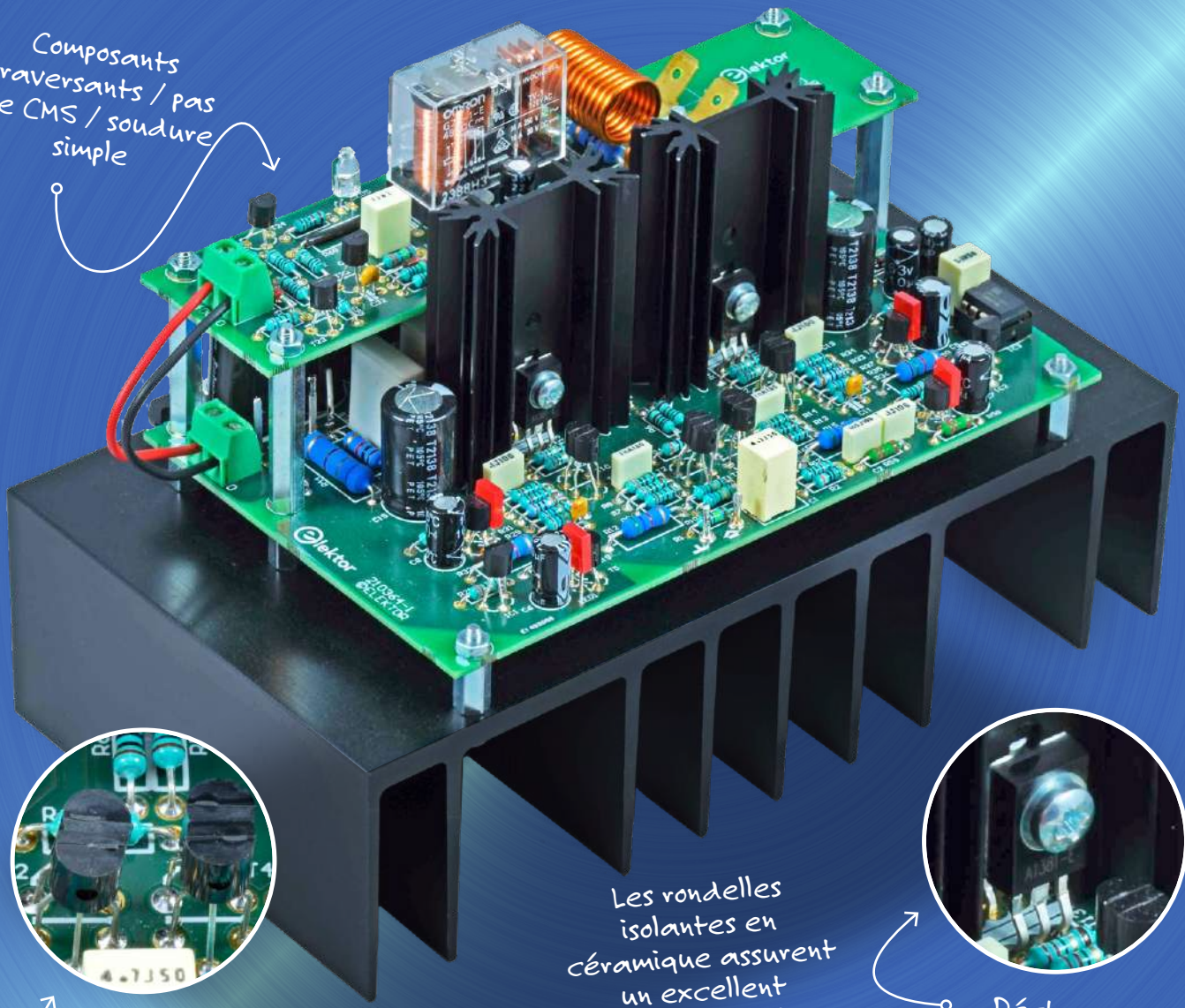


LIENS

[1] Périphériques ESP-ADF : https://elektor.link/ESPPeripherals
[2] ESP-ADF Button Peripheral : https://elektor.link/ESPButtonPeripheral
[3] Référence API ESP-ADF : https://elektor.link/ESPAPIReference
[4] Égaliseur ESP-ADF : https://elektor.link/ESPEqualizer
[5] esp-dsp : https://github.com/espressif/esp-dsp
[6] Guide de démarrage de l'ESP32-LyraT : https://elektor.link/ESP32LyraT

elektor JUMPSTARTER
from project to product

Kit **amplificateur de puissance Fortissimo-100** d'Elektor



Composants traversants / pas de CMS / soudure simple

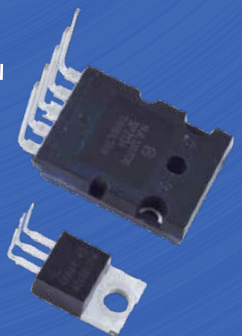
Transistors à petits signaux BC546B et BC556B

Les rondelles isolantes en céramique assurent un excellent couplage thermique avec le dissipateur

Décharges thermiques dans les pattes des transistors de l'étage push-pull



Transistors de puissance TO-264-5 et pilotes TO-220 montés au milieu sous le circuit imprimé



Transistors de l'étage de sortie fixés avec des vis M3

elektormagazine.com/posters
Format A2 et A3

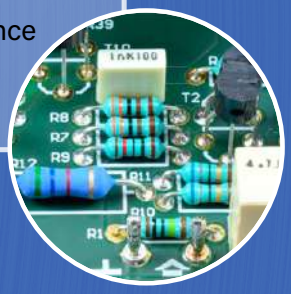


Circuit

- Amplificateur entièrement analogique, classe AB
- 100 % symétrique
- Distorsion et bruit extrêmement faibles
- Les transistors de puissance de ThermalTrackTM éliminent l'ajustement de la polarisation dans l'étage de sortie
- Étage de sortie à amorçage symétrique pour une oscillation de sortie maximale
- Une alimentation régulée de 40 V telle que SMPS800RE est exigée
- Impédance de charge minimale de 3 Ω
- Structure solide et compacte du circuit imprimé monté sur le dissipateur thermique.
- Travaux mécaniques limités à 12 trous M3 percés dans le dissipateur
- Détection de la tension d'alimentation de +40 V
- Protection de la sortie en CC

Composants

- Kit Elektor = circuits imprimés + tous les traversants + dissipateurs
- Transistors ThermalTrackTM avec diodes de polarisation intégrées
- Pilotes MJE15023 et MJE15033
- Transistors d'étage amplificateur push-pull KSC3503 et KSA1381 pour une linéarité élevée
- Boucle de contrôle CC pilotée par OPA177 (DIP-8)
- Résistances à film métallique de 0,6 W 1 % dans tous les états de faible puissance
- Condensateurs électrolytiques à longue durée de vie, haute température et faible TES pour l'amorçage et le découplage de l'alimentation.



Caractéristiques techniques

Sensibilité d'entrée	1,076 V (94 W/8 Ω, THD = 0,1 %, B = 22 kHz)
Impédance d'entrée	10 kΩ
Puissance de l'onde sinusoïdale	94 W (8 Ω, THD = 0,1 %) 181 W (4 Ω, THD = 0,1 %)
Bande passante	3,3 Hz – 237 kHz (–3 dB, 1 W/8 Ω)
Bande passante en boucle ouverte	≈ 20 kHz
Gain en boucle ouverte	≈ 140000 (8 Ω load)
Vitesse de balayage	45 V/µs
Rapport signal/bruit	103 dB (B = 22 Hz – 22 kHz linéaire)
Distorsion harmonique plus bruit	0,0008 % (1 kHz, 50 W, 8 Ω, B = 80 kHz) 0.002 % (20 kHz, 50 W, 8 Ω, B = 80 kHz) 0,0042 % (20 kHz, 100 W, 4 Ω, B = 80 kHz)
Distorsion d'intermodulation (50 Hz : 7 kHz = 4 : 1)	0,0015 % (50 W, 8 Ω) 0,0041 % (100W, 4 Ω)

En savoir plus
www.elektor.fr/20273

elektor
design > share > earn

utiliser la lumière pour les effets sonores

filtre synthétiseur à base de LDR, contrôlé en tension, 24 dB/oct



Raymond Schouten (Pays-Bas)

Cet article décrit un circuit de filtre réglable simple qui ne nécessite pas de puces spéciales, juste deux amplis-op alimentés en 5 V. Malgré sa simplicité, ce filtre présente un très faible taux de distorsion harmonique (THD < 0,01 %), une capacité de traitement des grands signaux et une large amplitude dynamique (> 90 dB).

Le filtre passe-bas audio de 24 dB/oct présenté ici est basé sur quatre résistances dépendant de la lumière, mieux connues sous le nom de LDR, éclairées par une seule LED. La luminosité de la LED contrôle la fréquence de coupure du filtre d'environ 20 Hz à 20 kHz. L'amortissement (c'est-à-dire le facteur Q, ou la résonance) peut être réglé jusqu'à l'oscillation complète, ce qui est une caractéristique intéressante pour les applications de synthétiseur musical. Le circuit est facile à construire, et le coût des composants est d'environ 4 €.

Ce filtre a été conçu pour des synthétiseurs musicaux, mais sa haute qualité de signal permet d'envisager son emploi dans d'autres applications audio. Le circuit a d'abord été simulé avant d'en réaliser un prototype sur une plaque d'essai pour mesurer ses performances. Des exemples de sons et une vidéo de démonstration sont disponibles sur la page du projet Elektor Labs [1].

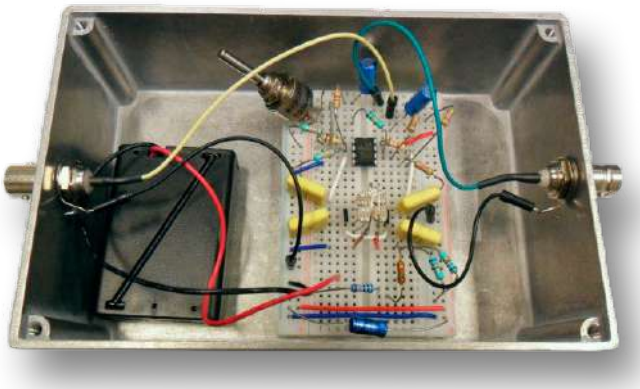


Figure 1. La version à alimentation simple du filtre a été construite sur une plaque d'essai et placée dans une boîte métallique pour vérifier les performances, avec les trois piles AAA qui l'alimentent.

Deux versions

Deux versions du filtre sont disponibles, la plus simple fonctionnant avec une alimentation double symétrique. En ajoutant quelques résistances et condensateurs, on peut en faire une version fonctionnant avec une alimentation simple. C'est cette deuxième version qui a été construite et mesurée (**figure 1**).

Le circuit (alimentation double)

Nous allons expliquer le circuit du filtre à alimentation double car il utilise moins de composants. Il est illustré à la **figure 2**. La version à alimentation simple (**figure 3**) fonctionne exactement de la même manière.

Le cœur du filtre est un filtre RC passe-bas de 6 dB/octave dont la résistance fixe R est remplacée par une LDR. Quatre de ces filtres sont mis en série pour obtenir une pente de 24 dB/octave. En contrôlant simultanément la résistance des quatre LDR par une seule LED, on obtient un filtre dont la fréquence de coupure peut être réglée sur une large plage (> 1 : 1000).

La mise en série de quatre circuits RC identiques sans tampon a l'air d'une invitation au désastre, résultant en un filtre très médiocre avec une transition grossière à la fréquence de coupure, mais puisque le célèbre filtre en échelle de Moog est lui aussi basé sur ce principe, ça ne peut pas être entièrement mauvais. Robert Moog a trouvé une solution astucieuse, reprise dans notre circuit. Elle est basée sur l'application d'une rétroaction négative globale

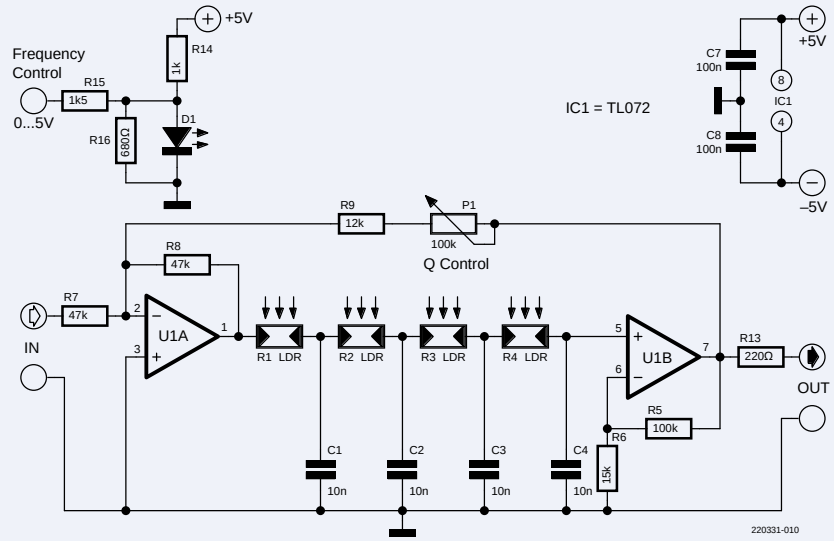


Figure 2. Schéma de circuit pour la version du filtre à alimentation double.

qui devient positive à la seule fréquence de coupure. L'encadré « Le contrôle Q du filtre expliqué » présente cela plus en détail.

L'ampli-op U1a tamponne l'entrée et ajoute le signal de rétroaction global via R9 et P1. L'ampli-op U1b est un tampon à haute impédance pour la cascade LDR/C et ajoute un certain gain nécessaire pour compenser la perte près de la fréquence de coupure.

P1 règle le niveau de rétroaction négative globale. Le filtre passe d'une réponse amortie à une oscillation complète en ajustant P1 de 100 kΩ à 0 Ω. Le remplacement de P1 par une autre combinaison LED-LDR permettrait d'obtenir un facteur Q contrôlé électroniquement. Reportez-vous à l'encadré intitulé « Résultats de la simulation de la réponse du filtre » pour le réglage de la fréquence et les courbes de contrôle du Q.

Une explication détaillée du choix de la LED D1 et du concept de son circuit de résistance se trouve dans l'encadré « La combinaison LED-LDR ».

Avantages du concept

La plupart des filtres analogiques réglables utilisent des circuits plus complexes limités à de faibles niveaux de signal (< 100 mVcc) pour maintenir la distorsion à un niveau acceptable. Il y a, par exemple, les filtres basés sur des transistors ou des diodes à polarisation réglable, discrets ou intégrés dans des circuits intégrés spéciaux (amplis-op à transconductance comme le célèbre LM13700). Notre circuit basé sur les LDR accepte des signaux de plusieurs volts d'amplitude avec des valeurs de distorsion très basses. En fait, l'amplitude de la tension est surtout limitée par les amplis-op et la tension d'alimentation, car la plupart des LDR peuvent supporter jusqu'à 100 V ou plus. Vous pourriez même construire ce filtre avec des tubes électroniques ! Comme le niveau de bruit du circuit est assez faible – il est principalement déterminé par les amplis-op et le bruit de la résistance équivalente des LDR – une large gamme dynamique peut être obtenue.

Un autre avantage du circuit est la fuite de tension de commande extrêmement faible (nulle ?) dans le chemin audio, car il n'y a qu'un couplage optique.

Avec des amplis-op adaptés, il fonctionne à partir d'une seule alimentation basse

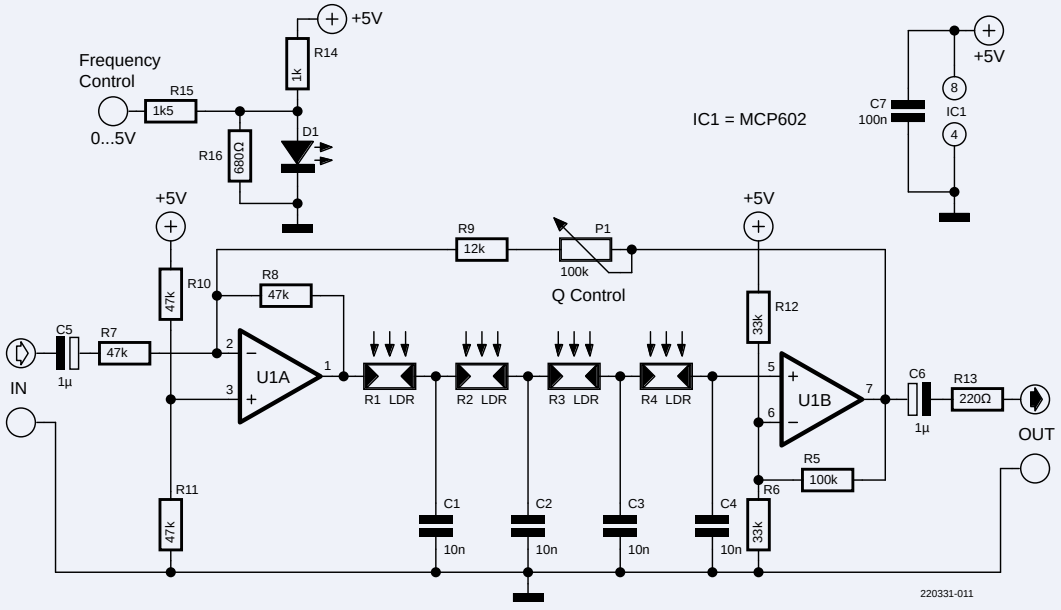
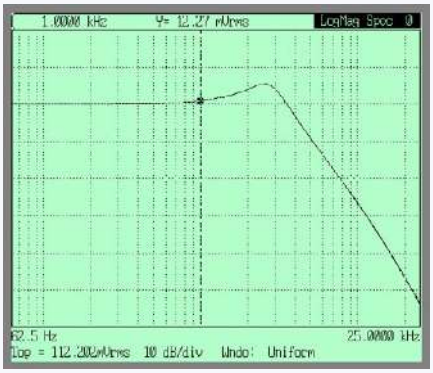


Figure 3. En ajoutant quelques résistances et condensateurs, le filtre à alimentation double peut être transformé en un filtre à alimentation simple.

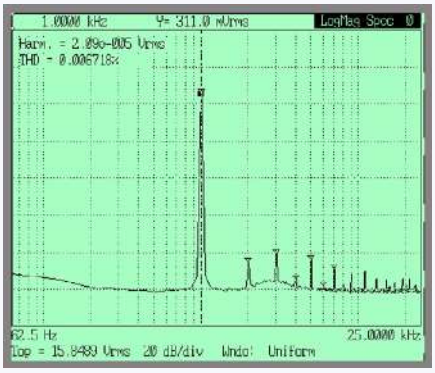
Mesures de la distorsion

Pour ces mesures, un analyseur de spectre FFT Stanford Research SR770 avec source intégrée a été utilisé. Pour éviter les parasites électriques et optiques, le circuit a été placé dans une boîte métallique fermée contenant également les3 piles AAA de l'alimentation simple (**figure 1**).

Tout d'abord, l'analyseur a été réglé pour mesurer la courbe de réponse du filtre. Puis il a été réglé pour générer une onde sinusoïdale de 1 kHz et mesurer le spectre de sortie du filtre jusqu'à 25 kHz. À partir de là, il a calculé la distorsion harmonique totale (THD) pour les six premiers harmoniques. Le filtre a été réglé sur une fréquence de coupure de 4 kHz. La raison de ce choix est qu'il y a une rétroaction positive à la fréquence de coupure, ce qui augmente la distorsion : on se trouve dans le pire scénario possible (les deuxième et troisième harmoniques sont en général dominants). On pourrait penser qu'une partie des harmoniques supérieurs est supprimée, mais un test à fc = 13 kHz a montré un profil harmonique presque identique.



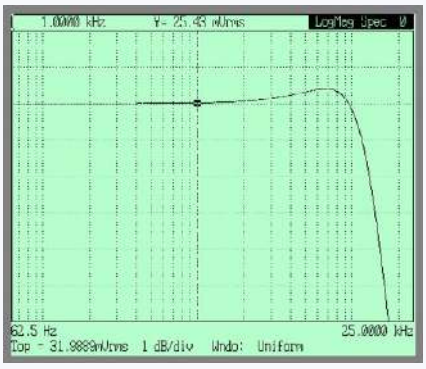
Réponse mesurée du filtre.



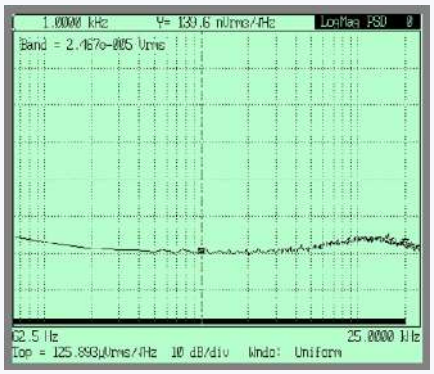
Spectre de distorsion du filtre mesuré.

Mesure du bruit et gamme dynamique

Le spectre de bruit de sortie est mesuré avec la source éteinte et il montre au total 25 µV effectif dans la bande audio jusqu'à 20 kHz. En combinaison avec la gestion maximale du signal de 4 Vcc (1,4 V effectif), cela conduit à une gamme dynamique de 95 dB. Notez que vous pouvez améliorer ce résultat en utilisant des tensions d'alimentation plus élevées.



Courbe de réponse mesurée du filtre réglé à -3 dB à 13 kHz (1 dB/div).



Spectre mesuré du bruit de sortie du filtre.

tension (3 V à 5 V), ce qui le rend compatible avec les applications à microprocesseur comme Arduino.

L'utilisation d'un contrôle de tension sur la LED (voir l'encadré « La combinaison LED-LDR ») permet d'obtenir une courbe de contrôle grossière, mais approximativement exponentielle, nécessaire pour les applications de synthétiseur. Pour un contrôle plus précis (mais linéaire), une source de courant ou un signal PWM peut être appliqué à la LED.

Limites du concept

Pour être juste, le concept de filtre présenté ici présente également quelques inconvénients par rapport aux autres concepts. Le réglage de la fréquence de coupure n'est pas très précis. C'est suffisant pour un synthétiseur basique, mais pour des applications plus exigeantes, une table d'étalonnage ou un circuit de contre-réaction supplémentaire pourraient s'avérer nécessaires.

De plus, la vitesse de variation de la fréquence de coupure est limitée par la lenteur relative des LDR. Lorsque l'on augmente la fréquence du filtre, la réponse mesurée reste de l'ordre de la milliseconde mais sa diminution peut prendre jusqu'à 50 ms et même jusqu'à 500 ms pour les fréquences de coupure très basses. Des temps d'« attaque » courts sont donc possibles pour ouvrir le filtre au début d'une note. En musique, le temps d'attaque d'un son est généralement plus court que son temps de déclin, ce qui rend ce filtre adapté aux applications musicales.

Suggestions et remarques

La plage dynamique peut être encore augmentée en utilisant des tensions d'alimentation plus élevées. Comme mentionné précédemment, les LDR peuvent généralement gérer des signaux jusqu'à 100 V.

Pour obtenir un contrôle de fréquence plus précis et plus stable, vous pourriez utiliser cinq LDR au lieu de quatre et en utiliser un dans la boucle de rétroaction d'un ampli-op pilotant la LED. Le filtre ayant été conçu pour des applications de synthétiseur de musique, le gain a été rendu assez dépendant de la valeur de Q pour éviter de fortes

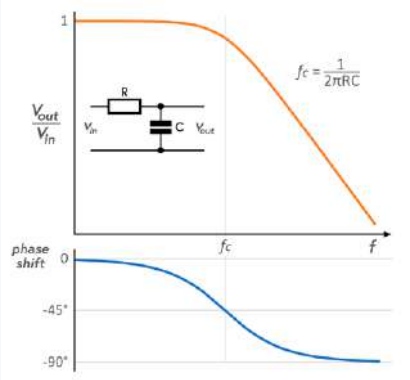
augmentations du signal de sortie. Comme alternative, le signal d'entrée pourrait être appliqué à l'entrée non-inverseuse de U1a au lieu de son entrée inverseuse. Cela limiterait l'influence de la valeur Q sur le gain du filtre et le filtre aurait une fonction de transfert non inverseuse.

(220331-04) — VF : Helmut Müller

Des questions, des commentaires ?

Envoyez un courriel à l'auteur (rs.elc.projects@gmail.com) ou contactez Elektor (redaction@elektor.fr).

Le contrôle Q du filtre expliqué

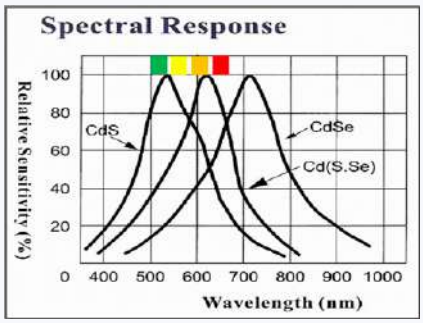


Un filtre passe-bas RC n'est pas seulement un filtre mais aussi un déphaseur dépendant de la fréquence, partant de 0°, passant à -45° à la fréquence de coupure et tendant vers -90° au-delà. Avec quatre de ces filtres RC en série, le déphasage à la fréquence de coupure devient 4 × -45 = -180°. Cela signifie que le signal change de signe à cette fréquence. Par conséquent, lorsque nous appliquons une rétroaction négative de la sortie vers l'entrée du filtre, il en résulte une rétroaction positive à la fréquence de coupure, ce qui produit un pic de gain à cette fréquence. Cela donne à votre filtre une transition nette à la fréquence de coupure. En ajustant la quantité de rétroaction négative, le pic de gain peut être modifié. À un certain niveau de rétroaction maximale, le filtre peut devenir un oscillateur (sinusoïdal) réglable de 20 Hz à 20 kHz.

La combinaison LED-LDR

Construction

La LED est montée en suspension à environ 5 à 7 mm au-dessus des quatre LDR. Les exigences en matière de précision mécanique sont modestes, les mesures ne montrent aucune différence de réponse significative lorsque la LED est poussée de 1 à 2 mm hors de l'axe. Veillez à protéger correctement le circuit de la lumière ambiante, car elle modulera également le filtre !



Source : JCLH, fiche technique du LDR GL5537-1

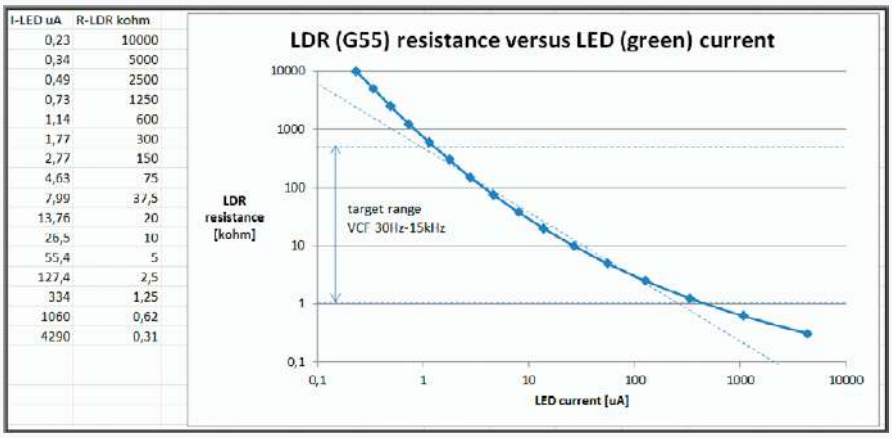
Choix de la couleur de la LED

J'ai utilisé des LDR à base de CdS, alors le vert ou le jaune fonctionne le mieux comme le montre ce diagramme de réponse spectrale.

Sensibilité

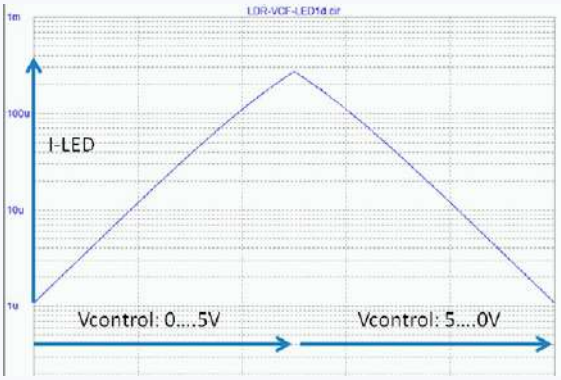
Les mesures ont montré que la résistance de la LDR est proche d'une fonction linéaire du courant dans la LED dans la plage de résistance de 500 kΩ à 1 kΩ. Au total, la résistance de la LDR a pu être contrôlée de 10 MΩ à 300 Ω, comme le montre cette courbe mesurée.

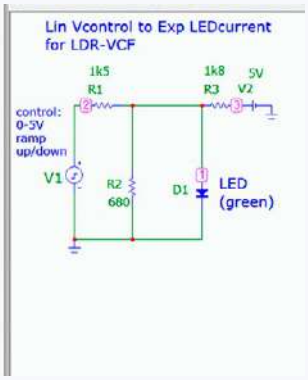
I-LED uA	R-LDR kohm
0,23	10000
0,34	5000
0,49	2500
0,73	1250
1,14	600
1,77	300
2,77	150
4,63	75
7,99	37,5
13,76	20
26,5	10
55,4	5
127,4	2,5
334	1,25
1060	0,62
4290	0,31



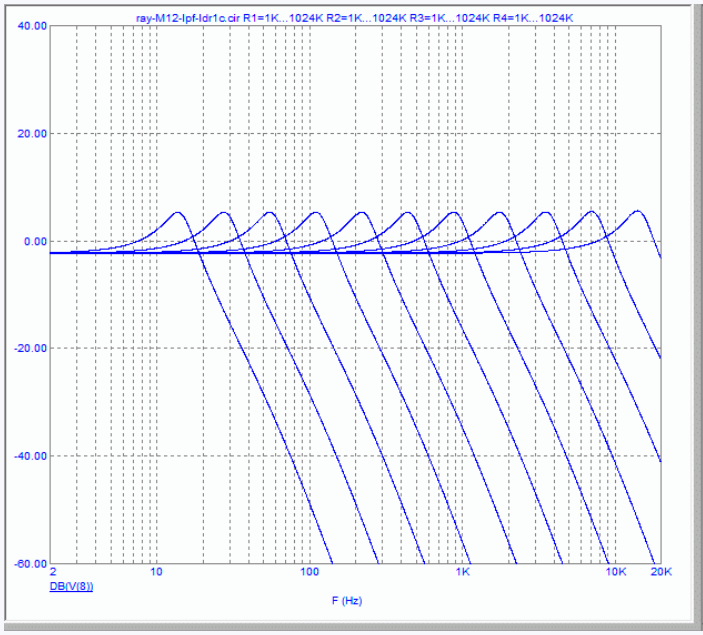
Note : Une LED verte à très haute luminosité a été utilisée ici (OVLBG4C7), 0,5 µA était suffisant pour avoir une lumière visible et pour commencer à diminuer la résistance de la LDR.

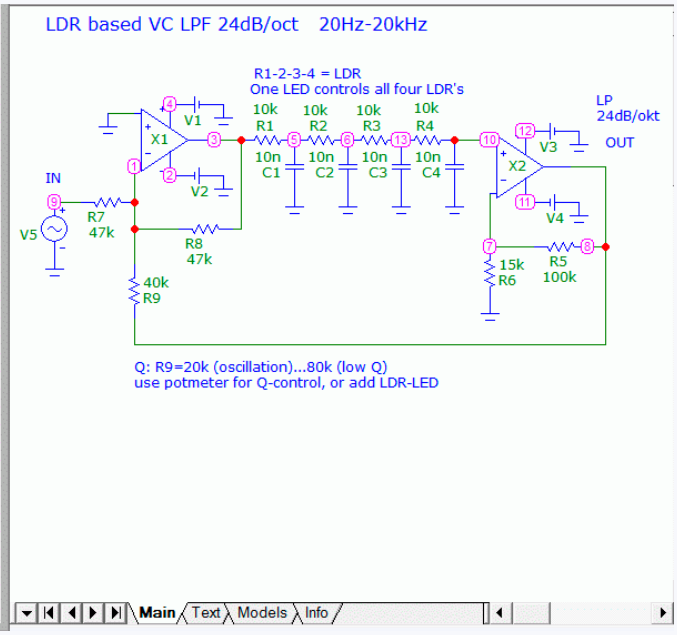
Il y a un moyen très simple mais approximatif d'obtenir une courbe de contrôle exponentielle sans utiliser de circuits supplémentaires. Le principe est le suivant. Lorsque l'on applique une tension à une diode (ici la LED), le courant résultant est une fonction exponentielle de cette tension. Connecter une source de tension directement à une LED n'est pas une bonne idée, mais ici nous utilisons des valeurs de résistance suffisamment faibles pour nous en approcher. La simulation ci-dessous montre que lorsque la tension appliquée à la LED varie linéairement de 0 à 5 V, le courant varie exponentiellement de 1 µA à 300 µA. Dans la pratique, vous devrez peut-être adapter un peu les valeurs des résistances à votre LED. Comme mentionné, cette méthode est assez imprécise et dépendante de la température mais c'est la façon la plus simple d'approcher le contrôle exponentiel pour les applications de synthétiseur. Les exemples sonores ont été réalisés en utilisant ce contrôle.



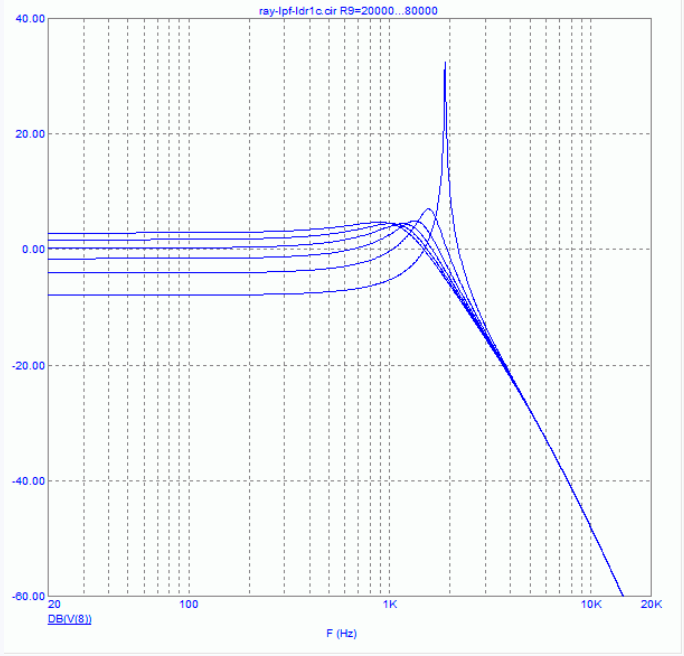


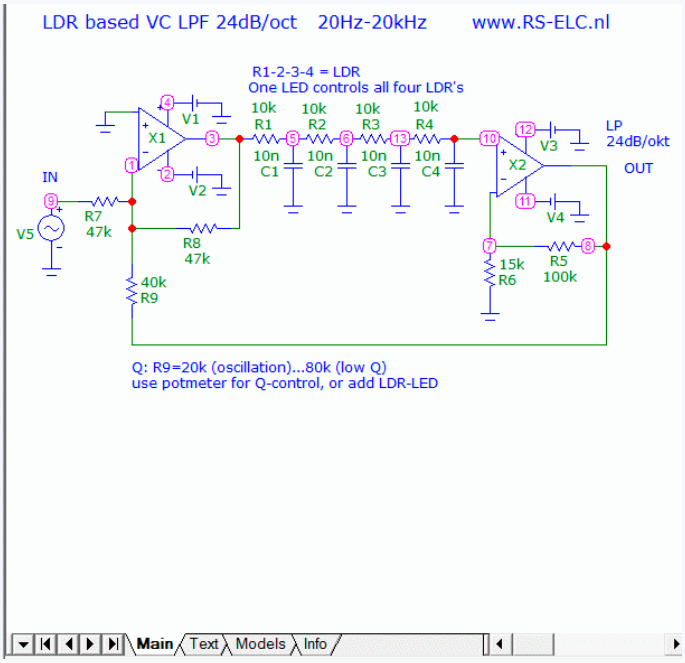
Résultats de la simulation de la réponse du filtre





Dans cette simulation, la fréquence de coupure du filtre est balayée de 20 Hz à 20 kHz tout en maintenant le contrôle Q à une valeur constante.





Ici, le contrôle du Q du filtre (R9) est balayé de 80 kΩ à 20 kΩ tout en gardant le contrôle de la fréquence de coupure à une valeur constante.

Produits

- **Joy-IT ScopeMega50 Oscilloscope USB (SKU 18277)** www.elektor.fr/18277
- **Carte d'entrée MIDI – circuit imprimé (SKU 18382)** www.elektor.fr/18382
- **Livre en anglais « The LTspice XVII Simulator » (SKU 19741)** www.elektor.fr/19741

LIENS

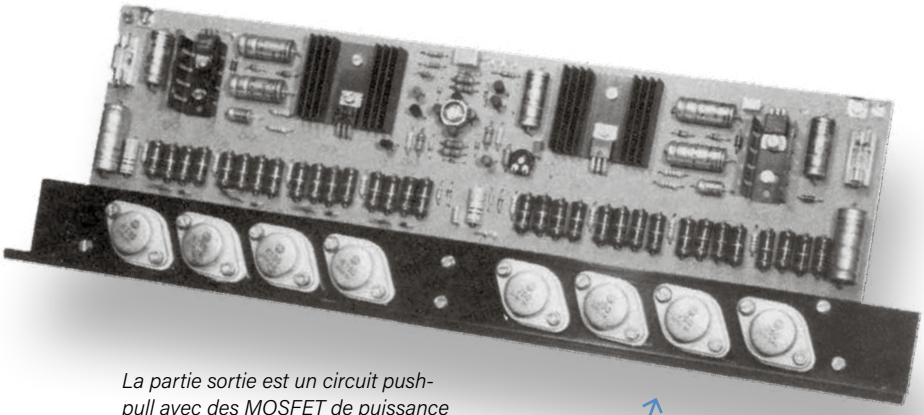
[1] Ce projet à Elektor Labs : www.elektormagazine.fr/labs/voltage-controlled-24dboct-synthesizer-filter-using-ldrs

amplificateur Hi-Fi d'Elektor

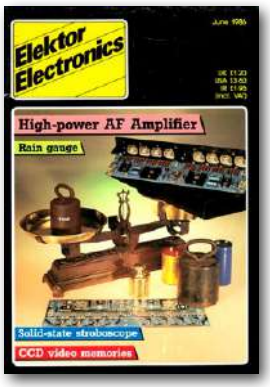
le plus **fort** de tous !

Larry Kossek (Elektor)

Les amplis audio font partie de l'ADN d'Elektor depuis le début. Thomas Scherer les a tous construits, de 40 W à 200 W, avant de tenter, en 1986, de faire une percée avec son propre circuit capable de délivrer un son de haute qualité à une puissance de sortie alors incroyable de 1000 W.



La partie sortie est un circuit push-pull avec des MOSFET de puissance complémentaires à canaux N et P de type horizontal, sélectionnés pour une bonne réponse transitoire et une linéarité à tous les niveaux de commande possibles.



Club des cloneurs de Segway

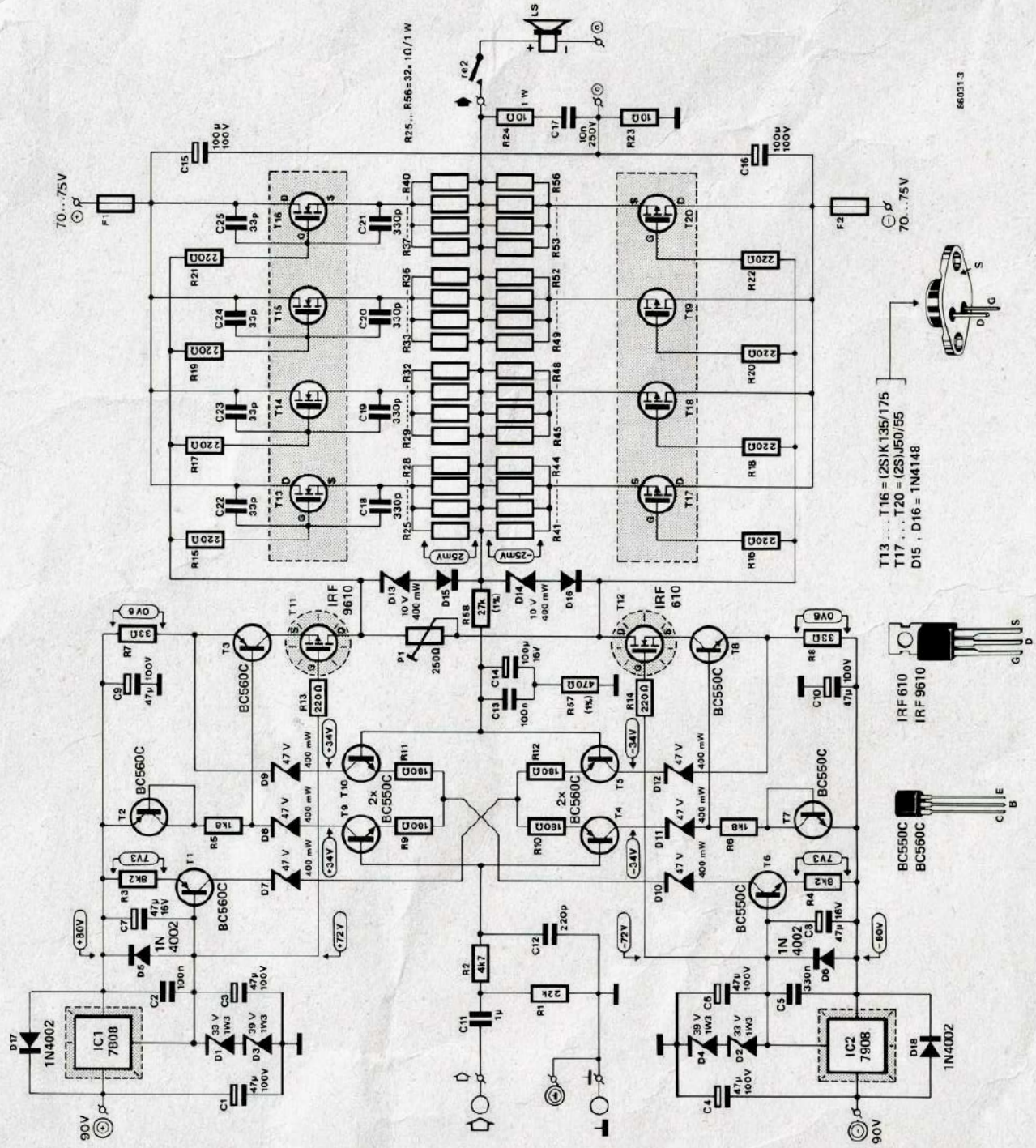
Fouiller dans les archives d'Elektor ravive des souvenirs et révèle des associations surprenantes. Thomas Scherer, l'homme qui a conçu ce gigantesque amplificateur Hi-Fi en 1986, est devenu plus tard un fan et un promoteur du projet Wheelie d'Elektor. Quel est le point commun entre un ampli audio de 1000 W et un clone de Segway ? C'est un composant à trois pattes, un boîtier métallique et une capacité de courant élevée. Exact, il s'agit du transistor de puissance MOSFET, le cheval de bataille qui a révolutionné l'électronique moderne en matière de gestion de la puissance. Dans les années 80, ces composants de puissance n'étaient pas encore banals dans les projets d'Elektor. Et Thomas avait déjà un penchant pour les circuits à courant élevé qui mettaient à l'épreuve les fusibles du laboratoire.

« Wer Großes will, muß sich zusammenraffen »

Librement : Si vous voulez quelque chose de grand, ressaisissez-vous. Bien sûr, les projets d'amplificateurs de puissance « normaux » étaient dans l'ADN d'Elektor dès le début. Ces amplis se sont succédé rapidement et Thomas les a tous construits, de 40 à plus de 200 W. En 1986, il était membre de la rédaction d'Elektor et rêvait d'un amplificateur audio si puissant qu'il pourrait fournir un son de haute qualité à la puissance alors inimaginable de 1000 W. Finalement, il en a conçu un chez lui, pendant son temps libre, sans que ses collègues ne le sachent. Il se couchait tard et beaucoup de ses rêves comportaient des paires de transistors de sortie complémentaires. « J'ai conçu les circuits analogiques en utilisant la puissance du calcul mental avec un

Caractéristiques

- Puissance de sortie : 2 × 250 W (8 Ω), 2 × 500 W (4 Ω), ou 1 × 1000 W (8 Ω)
- Bande passante : 8 Hz à 100 kHz
- Distorsion : 0,1 % (à 1 kW) ; 0,01 % (à 600 W)
- Facteur d'amortissement : > 100
- Mise sous tension progressive du transformateur, commande du ventilateur, surveillance de tension continue en sortie
- Sensibilité d'entrée : 0,775 Veff pour la puissance nominale
- Poids : devinez !



Remarquez la symétrie du circuit et la séparation claire entre l'amplification de tension et l'amplification de courant.

peu d'aide d'une calculatrice programmable Sharp. [...] Ma proposition a été acceptée par Elektor pour être examinée ».

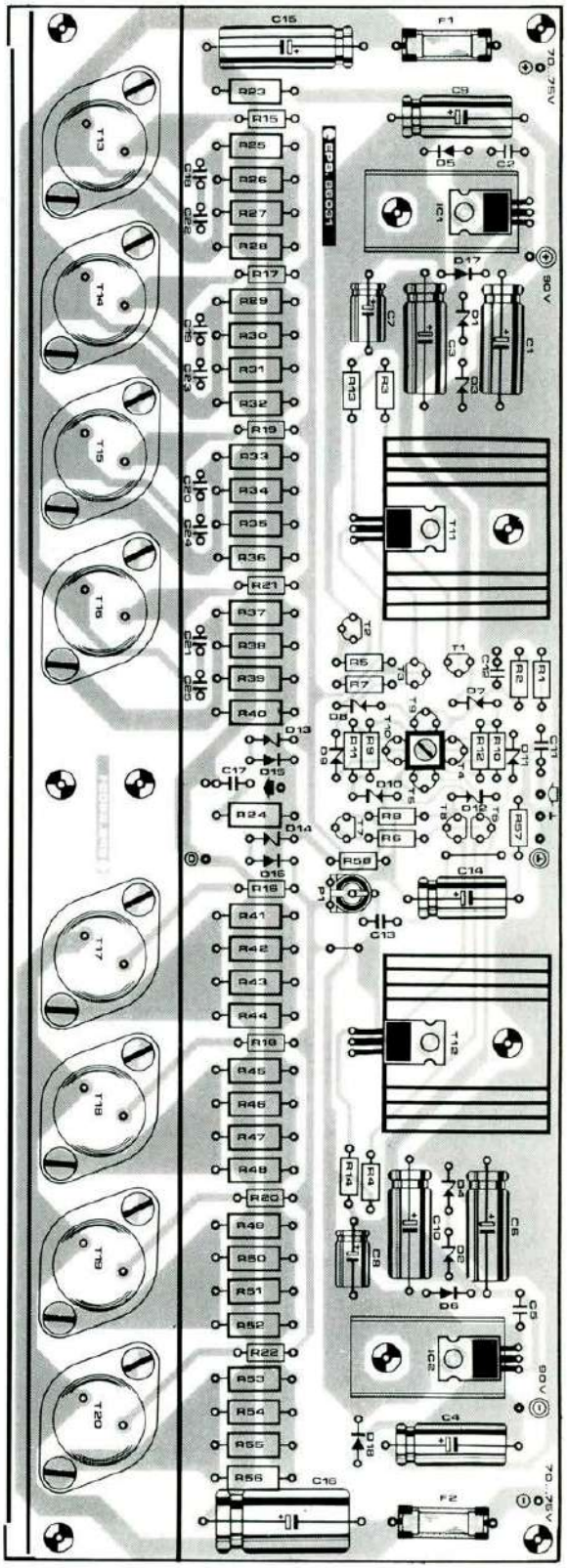
MOSFET à prix exorbitant

Rapidement, Thomas a dû passer du schéma théorique à un projet éprouvé, testé, grillé et reproductible : « l'Ampli Hi-Fi », sans griller trop de pièces, en particulier ces MOSFET Hitachi de type 2SK135/2SK175 (T13-T16) et 2SJ50/2SJ55 (T17–T20) au prix exorbitant.

« Avec des tensions d'alimentation aussi élevées que ± 90 V, j'ai dû faire face à des tensions de crête dépassant 125 V au niveau des connexions des haut-parleurs. Jusque-là, je m'étais trompé sur le côté sûr du 48 V. [...] Il était essentiel de retarder la connexion des haut-parleurs et d'avoir un démarrage progressif pour le transformateur de puissance. [...] Pour la distorsion et quelques autres mesures, ma référence était l'analyseur de spectre effroyablement cher de Brüel & Kjær qui honorait le laboratoire Audio d'Elektor auquel j'avais accès. »

L'amplificateur d'un kilowatt (aussi malicieusement appelé Killerwatts) a suscité un intérêt considérable et a résonné pendant de nombreuses années [1][2][3]. Rarement et plus de 35 ans après sa publication, un exemplaire de l'ampli kilowatt apparaît en vente sur Internet. Beaucoup de gens se sont pavanés avec leurs trucs funky et ont peut-être « fait sauter un fusible » avec des chansons jouées FORT à travers ce monstre d'amplificateur.

220234-04 — VF : Jean-Marie Gastineau



Les protections visent à empêcher les claquements sonores désastreux du ou des haut-parleurs et à éviter que les fusibles secteur ne sautent lorsque l'amplificateur est allumé.

Produits

› **Archives d'Elektor 1978-2021 (clé USB) (SKU 20073)**
https://www.elektor.fr/20073

LIENS

[1] « Balaise », T. Scherer, Elektor 5/1986 : https://www.elektormagazine.fr/magazine/elektor-198605/52715
[2] « Ampli Hi-Fi 1 kW (2ème partie) (Balaise) », T. Scherer, Elektor 6/1986 : https://www.elektormagazine.fr/magazine/elektor-198606/52734
[3] « Balaise, l'ampli Hi-Fi », T. Scherer, Elektor 5/2016 : https://www.elektormagazine.fr/magazine/elektor-201605/28946



visite à domicile

afficheur volumétrique made in Canada

Dan Foisy (Canada) et Eric Bogers (Elektor)

Un réseau de transducteurs ultrasonores permet de faire léviter une petite balle en mousse et de lui imprimer des mouvements très rapides. Éclairez-la à instants choisis avec des LED, et la persistance rétinienne fera le reste : l'œil verra un afficheur *véritablement* 3D.

L'idée d'une petite boule flottant au-dessus d'un tapis d'ondes ultrasonores ne date pas d'aujourd'hui. Elle est même désormais à la portée de tout amateur – voir p. ex. l'article Lévitation avec ondes acoustiques de mon collègue Luc Lemmens ; il y présente le kit d'un « lévitateur » acoustique conçu par Makerfabs [1].

En 2019, des chercheurs de l'université du Sussex ont présenté dans la revue Nature [2] un afficheur volumétrique à piège acoustique (abrégé ci-après en VDATP, pour Volumetric Display using an Acoustically Trapped Particle). La vidéo montrant son fonctionnement [3] fascina Dan Foisy, électronicien amateur vivant à Toronto. Dan avait assez d'expérience pour ne pas se laisser intimider par la complexité apparente de cet afficheur. Aussi décida-t-il d'en reproduire un semblable dans son labo. Mais laissons-lui la parole :

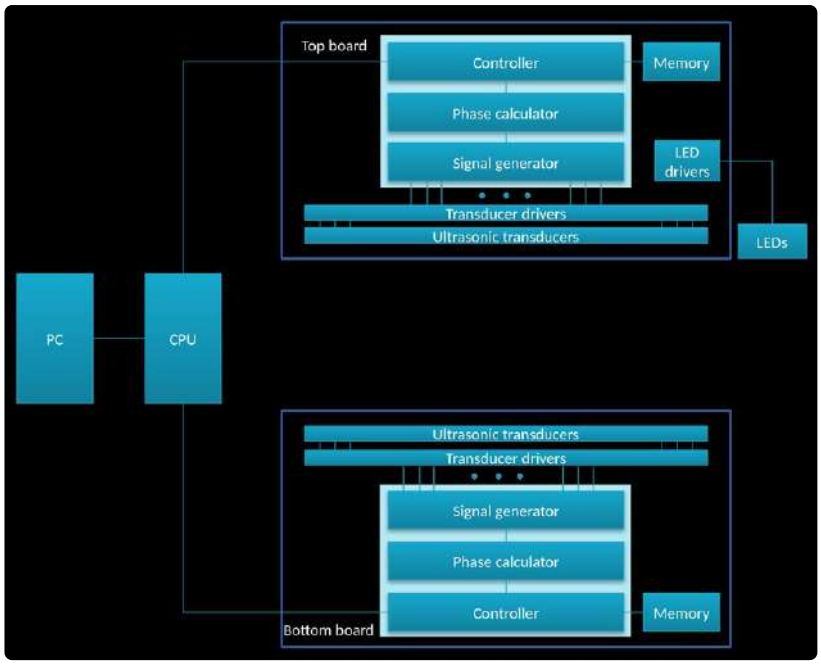
« Un réseau phasé de transducteurs ultrasonores permet de faire léviter une petite boule en mousse (1 mm) et de lui imprimer des oscillations rapides (plus de 1 m/s). On peut alors profiter de la persistance rétinienne pour que l'œil interprète ces mouvements comme des figures en 3D. Le volume de déplacement de la boule fait environ 100 x 100 x 140 mm. C'est en l'éclairant à des instants précis avec des LED RVB qu'on obtient l'illusion d'images multicolores. »

Ce type de réseau phasé est connu et exploité – en mode réception – depuis des décennies. On l'utilise par exemple en radioastronomie pour combiner les signaux de nombreuses petites antennes afin d'obtenir l'équivalent d'un radiotélescope de très grand diamètre (si grand qu'en pratique il serait impossible à construire). Imaginé aux Pays-Bas, le radiotélescope LOFAR [4] [5] comprend ainsi quelque 20 000 petites antennes réparties sur le continent européen. L'ensemble forme un interféromètre offrant une résolution sans précédent de 0,2 seconde d'arc.

« Il faut une grande puissance de traitement pour passer de la simple lévitation au contrôle de la balle et de sa position.



Figure 1. La boule en mousse lévite entre les deux cartes. Chacune comprend 100 transducteurs ultrasonores.



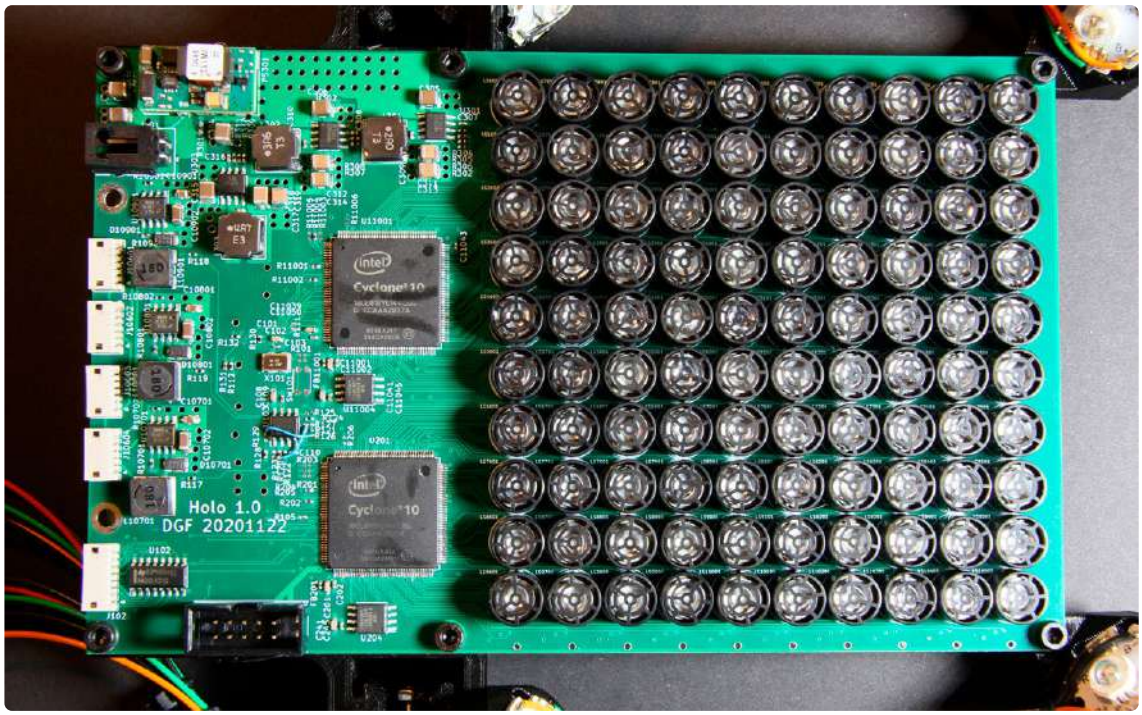
▲

Figure 2. Diagramme fonctionnel du VDATP. Les cartes du haut et du bas sont identiques, si ce n'est que celle du haut pilote aussi les LED.

Outre l'unité centrale, mon VDATP comprend quatre FPGA chargés de convertir une position en signaux de phase qu'interpréteront ensuite les transducteurs. Ces FPGA peuvent aussi tourner, déplacer, agrandir et rapetisser les images affichée, et même les animer – j'ai ainsi pu créer un papillon battant des ailes [6]. »

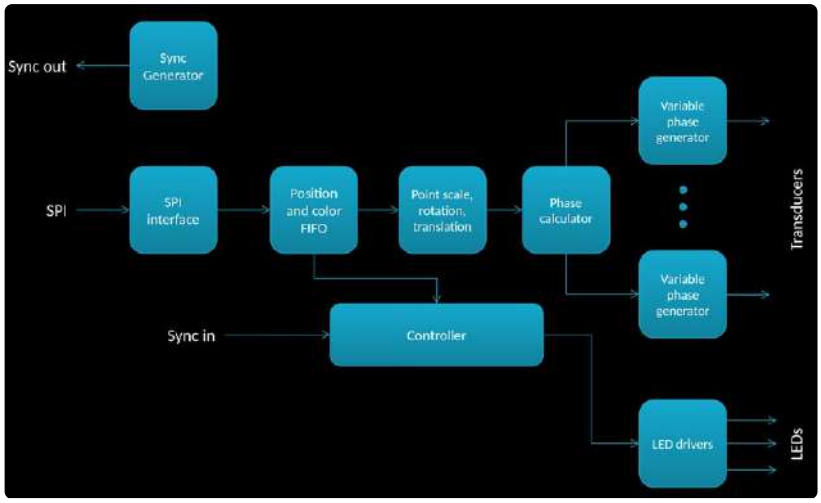
*À propos de transducteurs : ce projet en utilise 200 répartis sur deux cartes (***figure 1***). J'en profite pour dire que sa complexité en empêche toute description complète dans ces pages. S'il vous intéresse, contactez Dan Foisy par courriel (son adresse figure dans l'encadré* **Des questions, des commentaires ?***).*

« Un des défis du projet a été de concevoir le diagramme fonctionnel du VDATP (**figure 2**), pour l'essentiel constitué de deux cartes de circuit imprimé – une en haut, l'autre en bas. La **figure 3** montre l'une d'elles. La **figure 4** est une vue plus détaillée du diagramme fonctionnel de la carte du haut, celle qui pilote aussi les LED. »



◀

Figure 3. Une des cartes assemblées. Ses 100 transducteurs ont été testés individuellement avant montage.



▲

Figure 4. Diagramme fonctionnel du code FPGA.

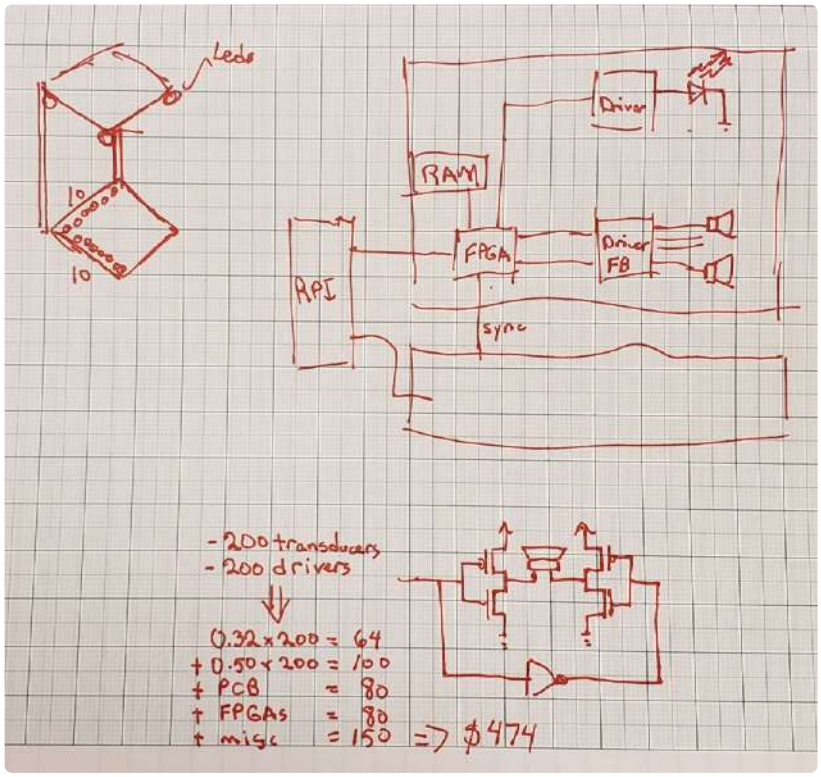
Comme le sait tout maker, passer de l'idée à la réalisation est une entreprise de longue haleine. Le VDATP n'a pas échappé à cette vérité. La **figure 5** montre un des premiers croquis du projet. Les **figures 6** et **7** révèlent son mûrissement et sa concrétisation en schémas – avec l'appui essentiel de KiCad.

« C'était un projet vraiment sympa à mener, et je prends toujours autant de plaisir à le regarder. Il y a quelque chose de magique à voir un objet léviter puis soudain s'agiter, si vite que ses mouvements sont à peine visibles. J'en tire une grande satisfaction. »

210237-04 — VF : Hervé Moreau

Figure 5. Un des premiers croquis du VDATP.

▼



À propos de Dan

Dan Foisy est né à Montréal (Canada) au milieu des années 1970 et a grandi en même temps que les ordinateurs personnels et l'Internet – une période exaltante. Sa famille a déménagé à Toronto à la fin des années 1970.

Le jour de ses 8 ans, Dan a reçu comme cadeau un coffret d'initiation à l'électronique, le 160-in-1 de Radio Shack. Il s'en servit pour construire des alarmes, des radios et autres amplificateurs, et apprit au passage à dessiner et réaliser ses propres circuits.

Dan a passé une licence et un master en Sciences appliquées à l'université de Toronto. Durant son master, il a intégré le laboratoire Space Flight Lab – qui venait d'être créé – et a travaillé sur le premier télescope spatial canadien, le MOST (Microvariability and Oscillation of Stars) [7]. Il fut par la suite responsable de cinq autres programmes spatiaux. Après avoir occupé différents postes, Dan est aujourd'hui employé par la Banque Royale du Canada. Il y dirige plusieurs équipes liées à divers domaines de l'IA, dont le traitement du langage naturel et les questions d'éthique liées aux applications de l'IA.

Lorsqu'il a demandé sa future épouse en mariage, Dan a accompagné sa demande d'une bague en diamant. Sa fiancée a alors trouvé normal de lui offrir quelque chose en retour. Ce fut du matériel pour son labo d'électronique personnel – qu'il utilise toujours. « Ce jour-là, j'ai fait une belle affaire :-) ! »

Des questions, des commentaires ?

Si la construction ou une déclinaison future de l'afficheur volumétrique vous intéressent, contactez l'auteur (danfoisy@gmail.com).

Produits

- **Kit : lévitateur acoustique de Makerfabs (SKU 19984)**
  www.elektor.fr/19984

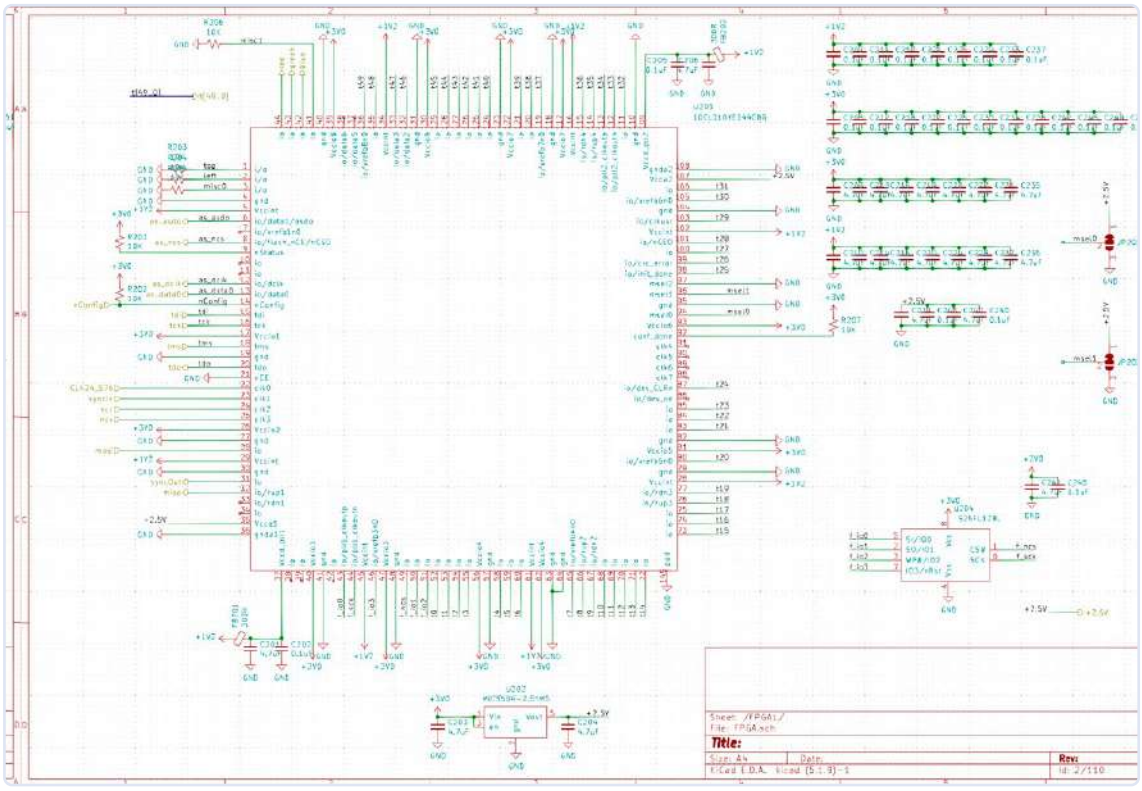


Figure 6. La version finale d'une partie du circuit. Papier et crayon ne suffisent pas pour un projet de cette envergure, un logiciel comme KiCad est indispensable.

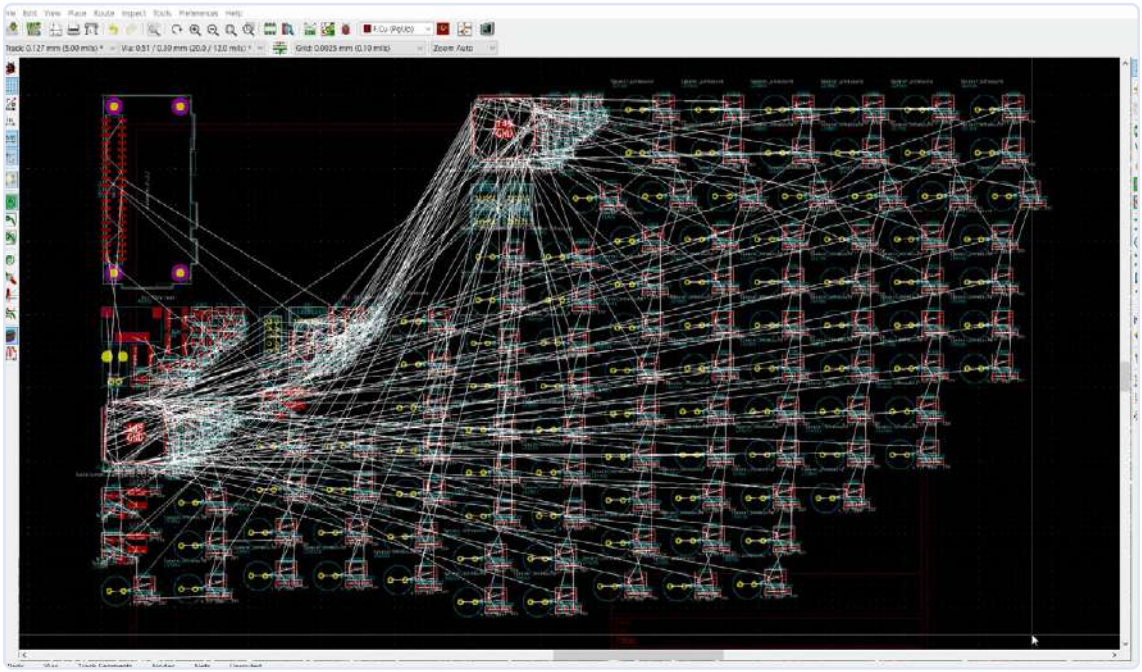


Figure 7. Un cochon n'y retrouverait pas ses petits. Dan, si.

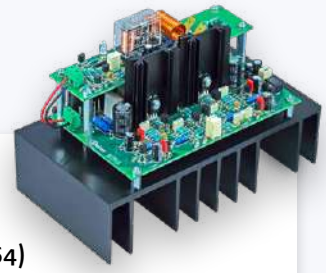
LIENS

[1] Luc Lemmens, « Lévitation avec ondes acoustiques » Elektor juillet/août 2022 : http://elektormagazine.fr/magazine/elektor-267/60601
[2] Article original de l'université du Sussex (2019) : https://sro.sussex.ac.uk/id/eprint/86930/
[3] Vidéo du VDATP de l'université du Sussex : https://youtu.be/Tm8JRlJ1q50
[4] Présentation du LOFAR : https://www.astron.nl/telescopes/lofar/
[5] LOFAR (Wikipédia) : http://fr.wikipedia.org/wiki/LOFAR
[6] Vidéo du VDATP de Dan Foisy, dont le papillon animé : https://youtu.be/hCC1C5KIeUA
[7] Télescope spatial MOST : https://elektor.link/MOSTSpaceTelescope

projet 2.0

corrections, mises à jour et courriers des lecteurs

Jens Nickel (Elektor)



Testeur de semiconducteurs de puissance

Circuits de vacances 2022, p. 56 (210707)



Cet article m'a permis de réparer une alimentation à découpage : j'ai dessoudé le « suspect » et pu le déclarer coupable sans erreur possible. Je l'ai remplacé par le composant d'une autre carte.

J'avais ce qu'il fallait en réserve pour assembler le testeur, à l'exception d'une ampoule de 12 V adaptée. J'ai donc utilisé une résistance (en métal) de 47 Ω/5 W. La chute de tension (via un pont redresseur DIL en raison d'une polarité différente) permet d'allumer une LED avec une résistance-talon de 820 Ω (cf. photo).
Merci à l'auteur David Ashton !

Ulrich Strohmeyer

Amplificateur haut de gamme Fortissimo-100

Elektor 11-12/2022, p. 6 (210364)

L'amplificateur est intéressant et l'article riche d'informations. Mais il en manque une, celle sur la puissance consommée. D'où ma question : combien de watts consomme un canal, au repos et à puissance moyenne ?

René Reynders

À l'état de repos, l'intensité totale maximale vaut environ 350 mA après mise en température. La puissance consommée vaut alors 28 W. L'alimentation ayant une tension stable de ±40 V, la puissance moyenne maximale est égale à la puissance continue spécifiée. La puissance de sortie moyenne pour une lecture musicale non déformée est toutefois bien inférieure à la puissance de sortie continue (maximale) spécifiée. La puissance totale pour une intensité CA et une tension CC est la moyenne de leur produit. Pour 181 W sous 4 Ω, l'intensité moyenne vaut $\sqrt{(181/2)}/\pi = 3{,}03$ A. Pour une onde sinusoïdale maximale non déformée, la puissance absorbée totale sous 4 Ω vaut donc 2 × 3,03 A × 40 V = 242 W – soit un rendement de 74,7 % pour l'étage de sortie.

Ton Giesberts

Avis de décès

C'est avec tristesse que nous annonçons le décès de *Peter Krengel*, un de nos auteurs. Peter était un *maker* créatif, toujours en quête d'idées nouvelles. Il avait de nombreux projets en cours – aussi bien des fours à refusion que des milliohmètres – et travaillait sur des livres Elektor.

Peter, tu vas nous manquer !

L'équipe Elektor

Filtre coupe-bande de Fliege pour les mesures audio

Elektor 9-10/2022, p. 80 (210551)

Cet article m'a de suite emballé : enfin un projet analogique « à l'ancienne », utile, avec des composants d'aujourd'hui et de grande qualité !

J'ai contacté l'auteur de l'article, et il s'est montré amical et courtois. Il lui restait quelques cartes qu'il m'a envoyé sans tarder, de plus à un prix intéressant. Nous avons aussi discuté des variantes possibles de son circuit.

La diversité est essentielle au succès de votre super magazine, gardez-la !

Jens Lemkamp

Merci pour votre commentaire, et heureux de vous savoir content !

Jens Nickel, équipe éditoriale

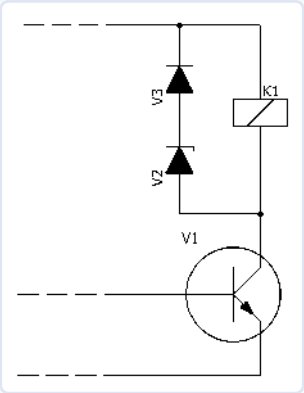


Figure 1

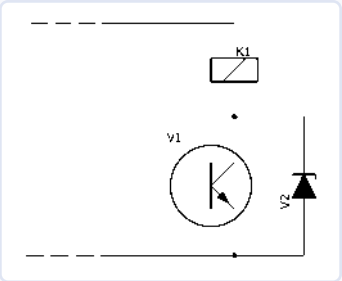


Figure 2

Démarrer en électronique

Elektor 9-10/2022, p. 34 (220256)

Merci beaucoup pour votre article du numéro de septembre et octobre expliquant clairement certaines applications des diodes, notamment celle d'une diode de roue libre dans un relais (**figure 7** de l'article).

Je m'étais déjà servi d'un tel montage pour commuter une électrovalve de remplissage. J'avais remarqué que la valve s'ouvrait immédiatement mais se fermait avec un certain retard. Ce délai différait d'une valve à l'autre (j'en utilisais plusieurs), de sorte qu'il m'était impossible de le compenser, par exemple en fermant la valve un peu plus tôt.

L'explication était, bien sûr, que le courant dans la bobine ne diminuait que progressivement, car la diode de roue libre n'avait qu'une tension directe de 0,6 V, de sorte qu'à un moment donné le ressort de fermeture prenait provisoirement le dessus et fermait la valve malgré la force magnétique restante.

J'ai résolu le problème en montant une diode Zener selon la **figure 1**. Sa tension fait décroître le courant dans la bobine beaucoup plus rapidement, ce qui réduit considérablement le temps de fermeture. La différence est perceptible à l'oreille : le léger « pling ! » initial s'est transformé en « dong ! » assez fort.

Je n'ai jamais vu cette astuce ailleurs, ni dans la littérature, ni en pratique, ce qui me surprend beaucoup.

Le circuit peut être simplifié en omettant la diode de roue libre et en reliant la Zener en parallèle avec le transistor (**figure 2**).

Cette variante pourrait intéresser d'autres lecteurs d'Elektor.

Thomas Klingbeil

Ce que notre système éducatif peut apprendre de la communauté des makers

Je suis dans une situation plutôt inhabituelle : j'ai 34 ans et fréquente un lycée technique allemand en vue d'obtenir un baccalauréat professionnel (ouvrant la porte à des études supérieures). Dans le même temps, je suis un cursus universitaire permettant aux lycéens méritants d'obtenir des points de crédits pour leurs futures études.

L'enseignement n'a guère changé depuis plus de 10 ans. Un professeur se tient devant sa classe et dicte des leçons sur des sujets théoriques dont les applications pratiques restent vagues dans l'esprit des élèves. Les enseignants ne semblent pas comprendre pourquoi, après quelques semaines passées sur la loi d'Ohm, demander à ces mêmes élèves de déterminer la résistance interne d'une source de tension lors de travaux pratiques ne suscite guère plus chez eux qu'un intérêt poli. La question qu'ils se posent est toujours la même : « À quoi ça va nous servir ? »

Si la communauté des *makers* était dans le même état d'esprit, nous aurions un problème. Arduino et Cie ont heureusement un point en commun : ils engendrent chez l'utilisateur une satisfaction presque immédiate. Les codes d'exemple et les projets à reproduire offrent souvent un point de départ à des idées personnelles. Le « pourquoi ? » spontané des élèves n'arrive ici que bien plus tard, lorsque certains ajustements deviennent nécessaires. Je suppose que pratiquement chaque lecteur de ce magazine, moi y compris, a commencé par en reproduire les circuits plus ou moins « à l'aveugle ». Lirait-on l'article d'un projet s'il ne montrait pas un circuit fini ? Ou s'il n'était présenté qu'au bout de quatre semaines ? Probablement que non.

Vouloir renouveler du jour au lendemain une structure aussi lourde que le système éducatif serait bien sûr chimérique. Pourtant, certaines choses pourraient être changées rapidement. Le débat sur la « jeunesse d'aujourd'hui » remonte à plusieurs décennies. Lorsque j'étais jeune, on accusait les jeux vidéo ; aujourd'hui, ce sont les médias sociaux qui sont pointés du doigt. Ce dont on les accuse ne fait pourtant que confirmer mon hypothèse : ils procurent un sentiment d'accomplissement dès le plus jeune âge. Imaginez que chaque jeu vidéo ou plateforme sociale nécessite un cours de préparation théorique de plusieurs semaines !

Un des programmes mis en place par l'université Albrechts de Kiel fournit un bon exemple de cette approche (einfachgutelehre.uni-kiel.de/allgemein/studieneingangsprojekt-elektrotechnik). Les étudiants en génie électrique de première année doivent assembler un détecteur de métaux avant même le début des cours magistraux. Les connaissances requises sont présentées au travers de contenus brefs et simples. Seul ce qui servira à la mise en œuvre du projet est transmis. Les étudiants débutent par la pratique, les questions de fond surgissent ensuite d'elles-mêmes.

Sebastian Westerhold

220601-04 — VF : Hervé Moreau

elektorstore

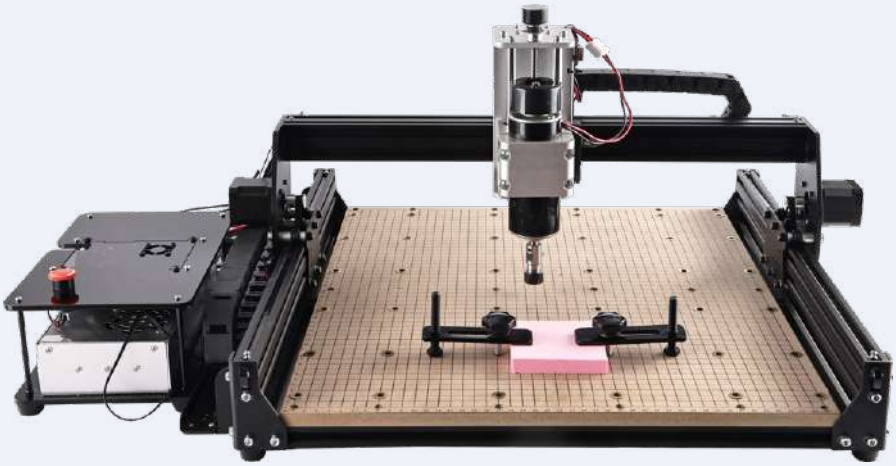
www.elektor.fr

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.fr. Seule exigence : **jamais cher, toujours surprenant !**

Anet 4540 fraiseuse CNC de bureau



-200 €

699,00 €
au lieu de
899,00 €
Prix (membres) : 629,10 €

www.elektor.fr/20260

Four à refusion à infrarouge T-962

version révisée d'Elektor

-40 €

229,00 €
au lieu de
269,00 €

www.elektor.fr/20346



Radio Logicielle HackRF One Great Scott Gadgets (1 MHz à 6 GHz)

-40 €



299,00 €
au lieu de
339,00 €

www.elektor.fr/18306

Raspberry Pi Pico W



Prix : ~~47,90 €~~
Prix spécial : 39,95 €

www.elektor.fr/20335

RISC-V Assembly Language Programming



Prix : ~~59,90 €~~
Prix spécial : 39,95 €

www.elektor.fr/20296

Coffret d'alimentation de labo JOY-iT RD6006 CC (360 W)



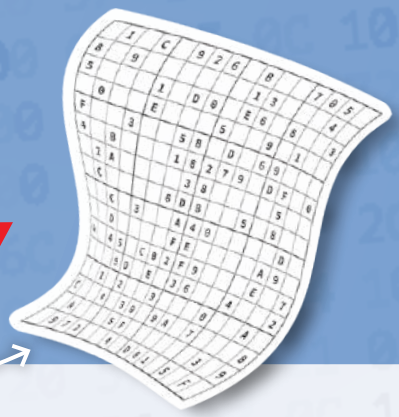
-35 €

Prix : ~~199,95 €~~
Prix spécial : 164,95 €

www.elektor.fr/19211

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 **15 février 2023** à l'adresse **hexadoku@elektor.fr**

LES GAGNANTS

La solution de la grille du numéro de novembre/décembre 2022 est **DBE39**.

La liste des gagnants est publiée ici : www.elektormagazine.fr/hexadoku

Bravo à tous les participants et félicitations aux gagnants !

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

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

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.

De la part **d'Arduino** et **d'Elektor** par **Amour de l'Ingénierie**

En vente maintenant

Édition spéciale du magazine Elektor !
Rédaction invitée : Arduino !

Des projets électroniques « maison », des idées pour les électroniciens, et plus encore, proposés par les experts d'Arduino et d'Elektor

De nombreux projets et tutoriels

Explorez des sujets d'actualité : MicroPython, TinyML et domotique avec Arduino

Découvrez l'Arduino : Les points de vue de Fabio, Massimo et David



Découvrez la Portenta X8

Achetez-le dès maintenant, dans votre kiosque préféré ou dans l'e-choppe d'Elektor.

Les liens dans les articles vous offrent un accès facile aux produits et solutions Arduino.

En savoir plus

www.elektor.fr/arduino-magazine


KL293-12
93-20
342
mouser.com