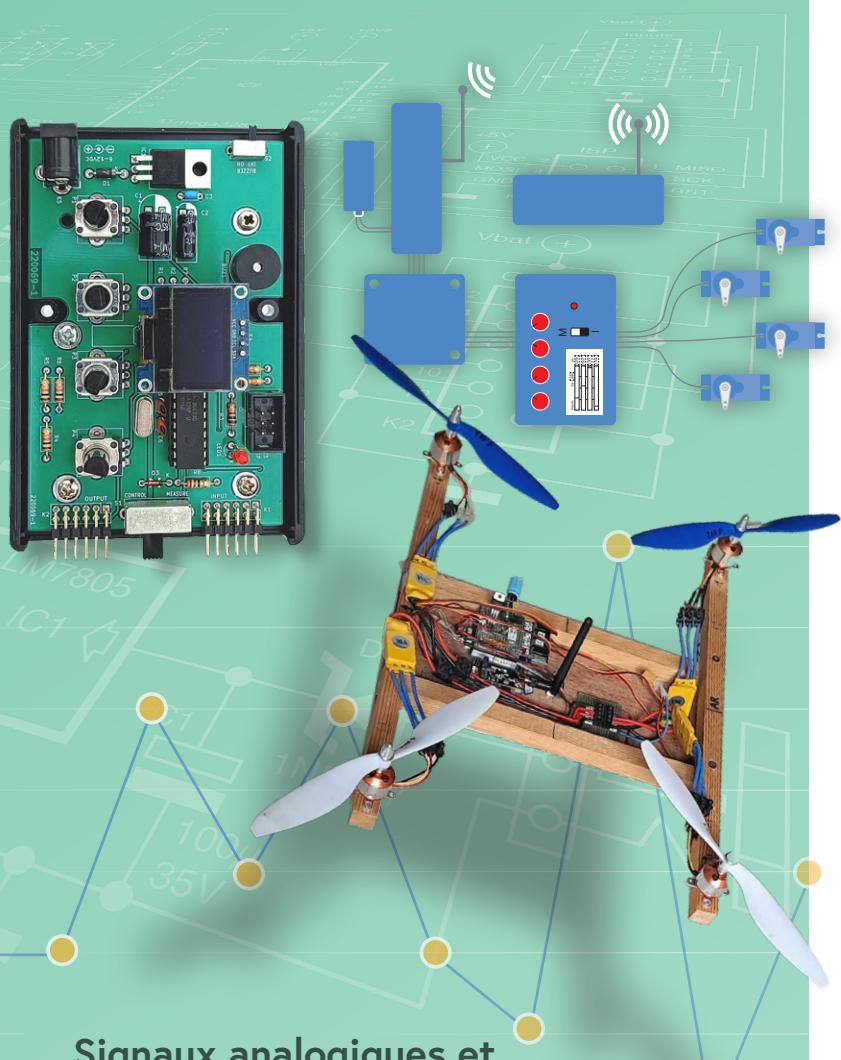




Testez jusqu'à quatre servos,
isolés ou in situ

SUPER Servo Tester



FOCUS SUR

Test et Mesure

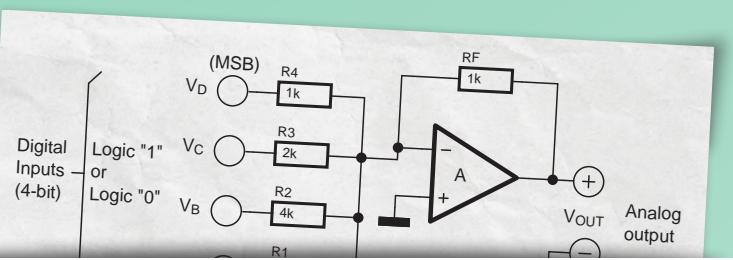
Automatisation des tests et des mesures

Programmation de l'équipement d'essai



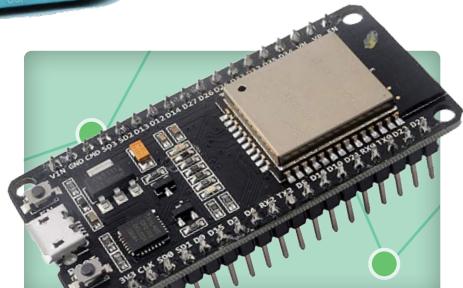
Signaux analogiques et microcontrôleurs

CA/N, CN/A, mesure du courant, et plus



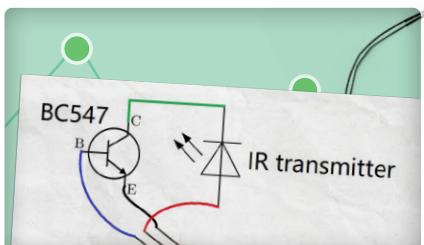
Enregistreur de données énergétiques
Mesurer la consommation d'énergie

p. 74



Connecter un smartphone Android à un ESP32
Projet avec l'API wifi d'Android

p. 26



Télécommande IR avec smartphone et ESP32
Adaptative et universelle

p. 48





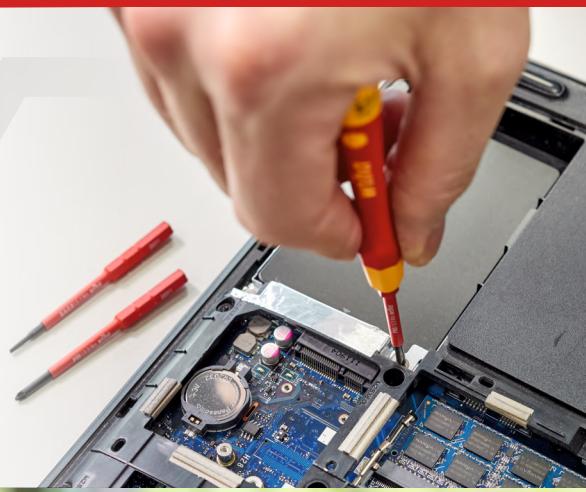
Electronic Components

TRANSFER
MULTISORT
ELEKTRONIK ...

■ GLOBAL DISTRIBUTOR OF ELECTRONIC COMPONENTS

DÉCOUVREZ LA GAMME DE TOURNEVIS
ELECTRICVARIO ET VOYEZ QUE
LA QUALITÉ COMpte

wiha 
Tools that work for you



En savoir plus:



Transfer Multisort Elektronik Sp. z o.o.
Ustronna 41, 93-350 Łódź, Poland, export@tme.eu

tme.eu

Rejoignez-nous :      

YOU NEED IT, WE HAVE IT!

■ ■ ■ ■ ■ tme.com ■

46^e année n° 501
mai-juin 2023
ISSN 0181-7450

N° de TVA Intracommunautaire : FR90319937454

Dépôt légal : mai 2023
CPPAP 1125 T 83713
Directeur de la publication : Donatus Akkermans

Elektor Magazine est publié 8 fois par an par
PUBLITRONIC SARL - c/o Regus Roissy CDG
1, rue de la Haye - BP 12910
FR - 95731 Roissy CDG Cedex

www.elektor.fr | www.elektormagazine.fr

Pour toutes vos questions :
service@elektor.fr

Devenez membre
www.elektormagazine.fr/abo

Publicité :
Ouafae Hassani
Tél. : +31 (0)6 41312932
ouafae.hassani@elektor.com
www.elektormagazine.fr/publicite

Tarifs Annuels :
France 1 an 129,95 € (8 numéros)

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).

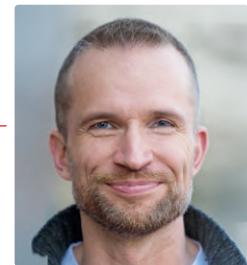
Certains circuits, dispositifs, composants, etc. décrits dans cette revue peuvent bénéficier de droits propres aux brevets; la Société éditrice n'accepte aucune responsabilité du fait de l'absence de mention à ce sujet. Conformément à l'art. 30 de la Loi sur les Brevets, les circuits et schémas publiés 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 Seneffeler Misset,
Mercuriusstraat 35, 7006 RK Doetinchem

Distribué en France par M.L.P. et en Belgique
par A.M.P.

Jens Nickel

rédacteur en chef d'Elektor Magazine



la connaissance pour tous !

La programmation est une discipline qui offre aux débutants un sentiment d'efficacité personnelle. Avec quelques connaissances de base d'un langage de programmation et quelques bonnes idées, vous pouvez créer un outil pour transformer un travail manuel laborieux en une tache simple. Soyons honnêtes : qui d'entre nous n'a jamais fièrement démontré ses compétences en programmation ?

La programmation d'une simple base de données, la configuration d'un serveur ou la réalisation de projets similaires ne sont cependant pas toujours appréciés par la famille et les amis. Mais l'appréciation augmente lorsque vous réalisez un projet de mesure, de régulation ou de contrôle – surtout si vous pouvez démontrer comment votre solution est utile pour se faciliter la vie au quotidien.

Certains jeunes makers dépassent rapidement leurs limites. Quiconque a déjà écrit quelques applications pour PC peut rapidement se familiariser avec la programmation des microcontrôleurs, grâce à des langages tels que Python. Mais comment connecter un microcontrôleur à des capteurs, des actionneurs et des afficheurs ? Quels sont les éléments à prendre en compte ? D'ailleurs, de nombreux étudiants en génie électrique ne le savent pas non plus, comme nous le confirment les professeurs d'université.

Chez Elektor, notre objectif est de transmettre ces connaissances de base aussi bien aux débutants qu'aux professionnels expérimentés. À la page 12, vouserez un article de Mathias Claussen qui décrit comment enregistrer des variables analogiques telles que la température avec des microcontrôleurs en utilisant des capteurs qui convertissent ces données en tensions et en courants. Comme d'habitude, l'aspect pratique n'est pas négligé. D'ailleurs, nos archives contiennent également quelques merveilles sur ce sujet - par exemple, la série « Sensors Make Sense » de Burkhard Kainka (Elektor 11/2016). Comme toujours, les membres d'Elektor peuvent télécharger gratuitement les PDF ; et, bien entendu, les articles sont également disponibles dans nos archives sur clé USB. (www.elektor.fr/20375).

Même à l'époque des modules enfichables prêts à l'emploi et de la génération automatique de code, de solides connaissances de base ne doivent pas être perdues. Je suis sûr que vous êtes d'accord avec moi sur ce point. Si vous avez des remarques ou des suggestions, écrivez-moi à l'adresse redaction@elektor.fr !



Proposez une contribution à Elektor!

Vos propositions sont les bienvenues ! Vous souhaitez proposer un article, un tutoriel vidéo ou une idée de livre ? Consultez le guide de l'auteur et la page de soumission d'Elektor : www.elektormagazine.com/submissions.

notre équipe

Rédacteur en chef : Jens Nickel | **Rédaction** : Asma Adhimi, Eric Bogers, Rolf Gerstendorf, Ton Giesberts, Hedwig Hennekens, Qielle Lont, Alina Neacsu, Thomas Scherer, Clemens Valens, Brian Tristam Williams | **Contributeurs réguliers** : Roberto Armani, David Ashton, Tam Hanna, Priscilla Haring-Kuijpers, Ilse Joostens, Prof Dr Martin Ossmann, Alfred Rosenkränzer | **Maquette** : Harmen Heida, Sylvia Sopamena, Patrick Wielders | **Des questions techniques** : redaction@elektor.fr

Super Servo Tester

testez jusqu'à quatre servos, isolés ou in situ

6



Rubriques

- 3 Édito**
- FOCUS**
- 24 Zone D**
l'échantillonnage sub-Nyquist en pratique
- 39 démarrez en électronique ...**
...les multivibrateurs
- 42 projet 2.0**
corrections, mises à jour et messages de lecteurs
- 72 sur le vif**
l'électronique des blocs
- 94 visite à domicile**
nouvelle cible
- 110 questions d'éthique**
IA générative
- 114 Hexadoku**
casse-tête pour elektorniciens

Articles de fond

- FOCUS**
- 12 signaux analogiques et microcontrôleurs**
CA/N, CN/A, mesure du courant, et plus
- 20 Embedded World 2023**
techniques intéressantes présentées lors du salon
- 26 connecter un smartphone Android à un ESP32**
projet pratique avec l'API wifi d'Android

- 44 le nouveau protocole I3C**
effet d'annonce ou vrai progrès
- 52 la documentation des microcontrôleurs sans peine (2)**
registres et schémas de principe
- 79 assemblage du kit Rover M.A.R.S. de 4tronix**
- 92 les coulisses de l'audio haut de gamme maison**
Ton Giesberts d'Elektor interviewé sur l'art de la conception analogique
- 106 kit bras robotique**
avec Raspberry Pi Pico et MicroPython

Industrie

- FOCUS**
- 56 automatisation des tests et des mesures**
programmation de l'équipement d'essai pour qu'il fasse ce que vous voulez
- FOCUS**
- 60 infographie**
test et mesure
- 62 immunité contre les surtensions**
protection pour modules de puissance CC/CC non isolés
- FOCUS**
- 66 appareil de mesure Wiha**
testeurs et compteurs d'installation sur lesquels vous pouvez compter
- FOCUS**
- 68 automatisation des tests et partage des résultats**

automatisation des tests et des mesures

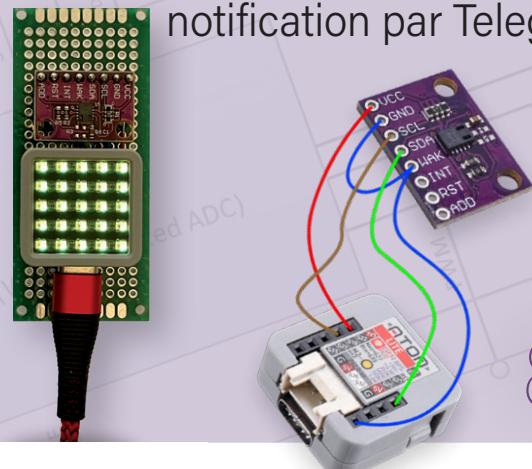
programmation de l'équipement
d'essai



56

eCO₂ Telegram Bot

moniteur de qualité de l'air avec notification par Telegram



89

Projets

FOCUS

6 Super Servo Tester

testez jusqu'à quatre servos, isolés ou in situ

FOCUS

34 filtre actif 1 kHz pour mesures de distorsion

mesures améliorées par l'optimisation du signal de mesure

48 BlueRC

télécommande IR avec smartphone et ESP32

FOCUS

74 enregistreur de données énergétiques

mesurer et enregistrer la consommation d'énergie

82 disque de stationnement avec affichage e-papier

une version informatisée innovante

FOCUS

89 eCO₂ Telegram Bot

moniteur de qualité de l'air avec notification par Telegram

98 lecture d'étiquettes RFID et serrure de porte RFID

exemples de projets tirés du guide d'expérimentation pour Arduino d'Elektor

FOCUS

103 sondes de courant RF pour oscilloscope

la mesure de courants RF sans peine



106

Bientôt dans ces pages

Le numéro de juillet - août 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 « IdO et capteurs ».

Quelques-uns des points forts :

station météo LoRa à faible consommation d'énergie

analyseur d'impédance basé sur ESP32

moniteur de vitesse basé sur GPS

le capteur Doppler en pratique

téléphone à cadran rotatif comme télécommande

tout sur le protocole Matter

horloge NTP du Raspberry Pi Pico W avec CircuitPython

guide pas à pas pour TinyML

et bien d'autres choses encore !

Le numéro de juillet - août 2023 du magazine Elektor sera publié aux alentours du 12 juillet 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.



Super Servo Tester

Testez jusqu'à quatre servos,
isolés ou in situ



Figure 1. Le quadricoptère pour lequel le Super Servo Tester a été conçu.

Olivier Croiset (France)

J'ai développé le Super Servo Tester présenté ici car le drone sur lequel je travaille produit parfois une désagréable odeur d'électronique brûlée et de vernis de cuivre roussi lors des vols d'essais. J'avais besoin d'un outil pour diagnostiquer le problème et m'aider à en comprendre la cause.

On trouve sur l'Internet de nombreux schémas de circuits permettant de tester un servomoteur seul. Ils servent à vérifier que le servo que vous venez d'acheter fonctionne correctement. Le testeur présenté ici va bien plus loin dans ce concept. Vous pouvez non seulement l'utiliser pour tester jusqu'à quatre servos en même temps, mais aussi pour tester l'unité de commande de vitesse électrique (ESC) ou le calculateur de vol avec ses servos dans de multiples configurations.

Le drone pour lequel j'ai eu besoin du Super Servo Tester est représenté sur la **figure 1**. Il est basé sur [1].

Le b.a.-ba des signaux servo

Pour rappel, un servo est commandé par un signal MLI dont la fréquence (ou taux de répétition des impulsions, PRR) est de 50 Hz. La largeur de l'impulsion (la durée

où le signal est haut) est censée être comprise entre une et deux millisecondes, le reste du temps le signal est bas. Le servo est dans sa position centrale lorsque la largeur de l'impulsion est de 1,5 ms. Lorsque l'impulsion est plus courte, le servo tourne dans un sens (par ex. à gauche) ; lorsqu'elle est plus longue, il tourne dans l'autre sens (par ex. à droite).

Dans un système de vol typique, on doit vérifier cinq paramètres : la durée des impulsions de commande des quatre servos (moteur, roulis, tangage et lacet) et la tension d'alimentation. Pour une conception complexe, par exemple le développement d'un calculateur de vol (d'un drone ou autre modèle RC) ou tout mélangeur d'impulsions, ce testeur permet de voir le résultat de ce mélangeur.

Notez que dans la suite de cet article, le terme « servo » peut être remplacé par ESC ou tout autre dispositif qui produit et/ou reconnaît les signaux servo.

Caractéristiques et capacités

Le Super Servo Tester (SST) mesure la durée des impulsions de commande de jusqu'à quatre servos et fournit des informations sur la qualité de l'alimentation électrique. Il peut être inséré entre le récepteur de télécommande et les servos, entre le récepteur et le calculateur de vol du drone, ou entre le calculateur de vol et les servos.

Les **figures 2 à 6** montrent les configurations possibles.

Modes de fonctionnement

La SST dispose de deux modes de fonctionnement, sélectionnables par un commutateur (à glissière) :

- **Manuel** : Dans ce mode, le SST génère les impulsions pour quatre servos ou pour le calculateur de vol. La largeur des impulsions est commandée par quatre potentiomètres. Dans ce mode, c'est le SST qui alimente les servos (ou l'ESC) qu'on ne doit pas connecter à l'alimentation du modèle. La tension d'alimentation du SST doit être comprise entre 7,5 et 12 V_{CC}.
- **Entrées** : Dans ce mode, on mesure la longueur des impulsions provenant du récepteur ou du calculateur. Les signaux sont également transmis aux sorties du SST pour commander les servos (ou le calculateur de vol). L'alimentation électrique du modèle alimente le SST et les servos (ou les ESC). Celle-ci ne doit pas dépasser 7,49 V et on ne doit pas connecter le SST à sa propre alimentation. De même, les quatre canaux d'entrée doivent être connectés, sinon la LED et le buzzer du SST signaleront un défaut.

Modes d'affichage

L'écran affiche la durée des impulsions sous forme de quatre graphiques à barres ainsi que leur valeur numérique en microsecondes (de 1000 µs à 2000 µs). Les graphiques à barres sont limités à la plage de 1000 µs à 2000 µs, mais la valeur numérique ne l'est pas. Lorsqu'une valeur est hors limites, une boîte est dessinée autour de sa valeur numérique. Dans ce cas, la LED s'allume également et le buzzer, s'il est activé, émet un bip.

L'écran affiche également la valeur de la tension d'alimentation des servos dans l'un ou l'autre des modes de fonctionnement (**Manuel** ou **Entrées**). En effet, la qualité de l'alimentation du modèle est importante pour le calculateur de vol et pour la sécurité du (ou de la) pilote et des personnes admirant ses talents de pilotage. La tension d'alimentation mesurée est encadrée lorsque sa valeur est inférieure à 4,5 V. La LED s'allume également et le buzzer émet un signal sonore, ce qui vous évite de devoir surveiller le testeur en permanence.

La SST propose deux modes d'affichage, empilé et en carré (voir **figure 7**). Le second type est mieux adapté à un quadrirotor. La position à la mise sous tension du potentiomètre P4 détermine le mode d'affichage. Lorsque P4 est tourné à fond vers la gauche avant la mise sous tension, le SST utilise l'affichage empilé. En tournant le potentiomètre P4 vers la droite avant la mise sous tension, c'est l'affichage en carré qui est sélectionné lorsqu'on allume l'appareil.

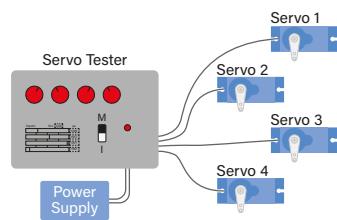


Figure 2. La configuration 1 (mode Manuel), permet de tester simplement quatre servos en même temps. Les servos sont alimentés par le testeur et commandés par les potentiomètres.

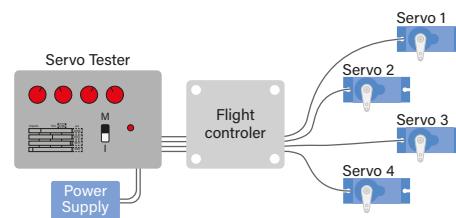


Figure 3. La configuration 2 (mode Manuel) permet de tester un calculateur de vol (de drone) sans émetteur ni récepteur. Les servos et le calculateur de vol sont alimentés par le testeur et commandés par les potentiomètres.

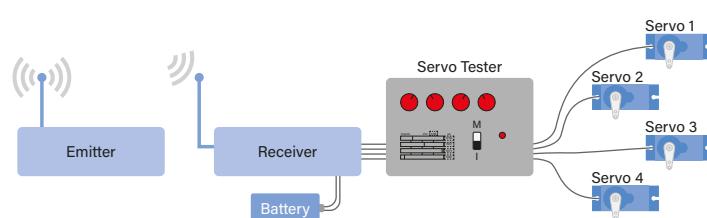


Figure 4. La configuration 3 (Mode Entrées) permet de vérifier l'émetteur et le récepteur. Le testeur, les servos et le calculateur de vol sont alimentés par la batterie du récepteur.

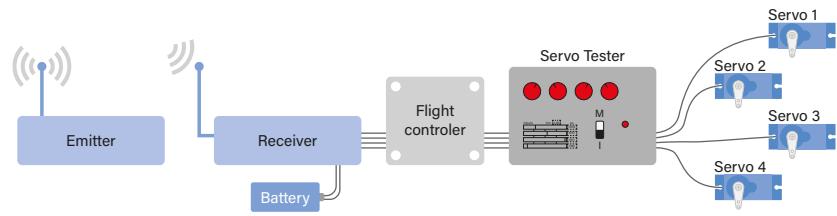


Figure 5. La configuration 4 (Mode Entrées) permet de vérifier le bon fonctionnement de l'émetteur et du récepteur avec le calculateur de vol (dans le cas d'un drone). Le testeur, les servos et le calculateur de vol sont alimentés par la batterie du récepteur.

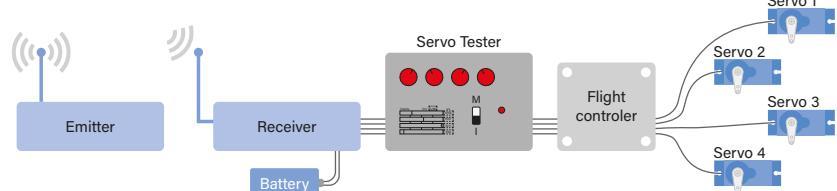
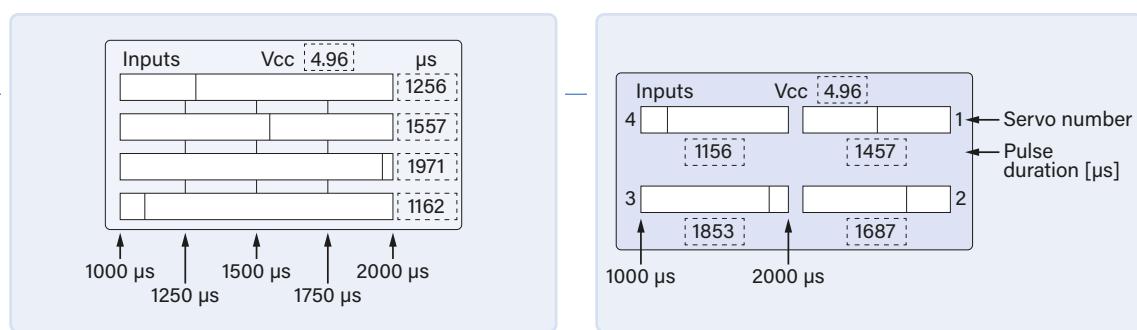


Figure 6. La configuration 5 (Mode Entrées) permet de tester le calculateur de vol avec l'émetteur et le récepteur. Le testeur, les servos et le contrôleur de vol sont alimentés par la batterie du récepteur.

Figure 7. L'écran affiche le mode de fonctionnement (Entrées ou Manuel) sélectionné par le commutateur S1, la tension V_{BATT} (comme V_{CC}) et un graphique à barres avec valeur numérique pour chacun des quatre servos. Trois index permettent d'estimer rapidement la valeur des impulsions reçues.



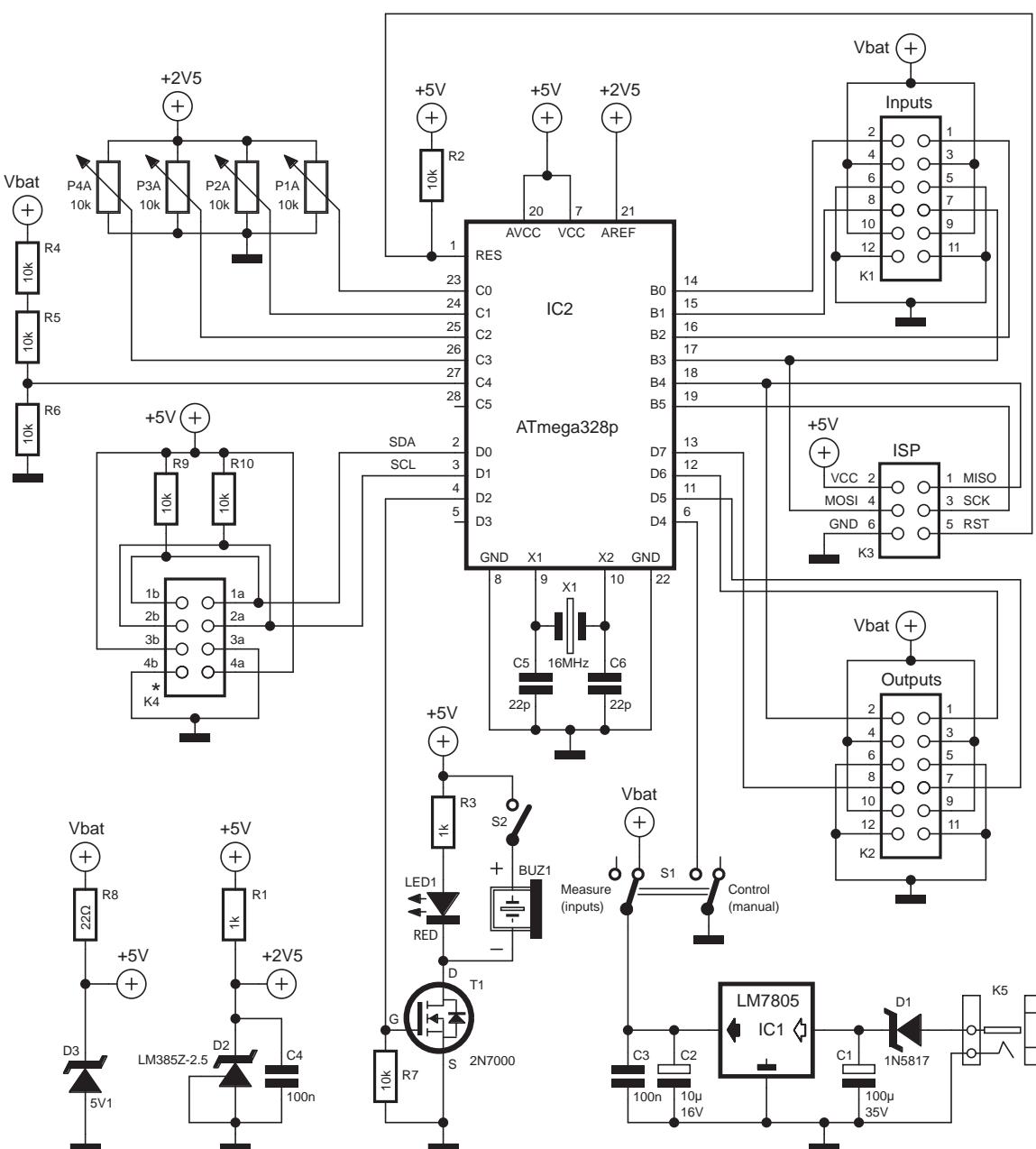
Circuit

Maintenant que nous savons comment utiliser le testeur, jetons un coup d'œil au circuit. Celui-ci est illustré à la **figure 8**. Il n'est pas trop compliqué car il est principalement constitué de connecteurs.

Le cœur du circuit est un microcontrôleur ATmega328P,

le même que l'on trouve sur une carte Arduino UNO. Sa fréquence d'horloge est déterminée par X1, un cristal de quartz de 16 MHz, aidé par C5 et C6.

Quatre de ses ports GPIO (PB0 à PB3) sont connectés aux entrées des signaux des servos sur K1. Les sorties de signaux servo (PD5, PD6, PD7 et PB4) sont connectées à K2.





Ces deux connecteurs sont câblés pour pouvoir utiliser directement des câbles servo standard. En d'autres termes, les triplets de broches 1-3-5, 2-4-6, 7-9-11 et 8-10-12 correspondent tous à une connexion servo. Ces connexions utilisent souvent un code couleur orange-rouge-marron. L'orange (signal d'impulsion) se connecte à la broche 1 (ou 2 ou 7 ou 8), le rouge (V_{CC}) à la broche 3 (ou 4 ou 9 ou 10) et le marron (GND) à la broche 5 (ou 6 ou 11 ou 12). Bien sûr, il y aura des exceptions à cette règle, alors vérifiez avant de connecter quoi que ce soit.

Entrées analogiques

Les quatre potentiomètres sont connectés aux entrées analogiques de l'UC sur PC0 à PC3. La tension d'alimentation est connectée par un diviseur de tension (R4, R5 et R6) à l'entrée analogique PC4. Le rapport entre (R4+R5) et R6 doit être de 2:1, mais leurs valeurs absolues ne sont pas critiques. L'utilisation de trois résistances de même valeur simplifie leur tri pour plus de précision.

Pour mesurer la tension d'alimentation du microcontrôleur, une tension de référence autre que la tension d'alimentation elle-même est nécessaire pour le convertisseur analogique-numérique (CA/N). L'ATmega328P possède une référence intégrée de 1,1 V, mais cette valeur est un peu basse. J'ai donc utilisé D2, un LM385-2,5, comme référence externe de 2,5 V. Ce composant est plus précis qu'une simple diode Zener à 2 bornes, ce qui améliore la qualité de la mesure de la tension d'alimentation du servo : 1 à 2 % au lieu d'environ 5 % pour une diode Zener ordinaire.

Comme le boîtier de D2 est haut, on peut la replier sur le CI pour limiter la hauteur de l'ensemble. Il est aussi possible de l'utiliser comme plot de soutien de l'écran. La mesure de la tension d'alimentation ne doit pas être supérieure à 7,49 V, car la tension d'entrée maximale de l'UC est de 5 V. Par conséquent, l'alimentation du récepteur et des servos ne doit jamais dépasser cette valeur.

Problèmes d'affichage

L'écran OLED I²C basé sur le SSD1306 est branché sur le connecteur K4. Le port I²C de l'écran n'est pas connecté au port I²C de l'UC, mais à PD0 et PD1. Le bus I²C est émulé par logiciel. En effet, sur l'ATmega328, le bus I²C est partagé avec l'entrée analogique PC4 qui est déjà utilisée pour mesurer la tension d'alimentation. Par conséquent, on ne peut pas ici utiliser le périphérique I²C intégré. R9 et R10 sont des résistances de rappel pour le bus I²C. En théorie, elles devraient avoir une valeur d'environ 4,7 kΩ, mais 10 kΩ fonctionne aussi et permet de gagner une ligne sur la nomenclature, car la plupart des autres résistances ont aussi une valeur de 10 kΩ.

Notez que K4 est dessiné comme un connecteur à 8 broches sur deux rangées, mais sur le CI, vous devez monter une barrette à une rangée de 4 broches dans la position « A » ou « B », pas les deux. La position dépend

de votre écran. Ces écrans ont les broches GND et VCC sur les broches 3 et 4, mais pas toujours dans le même ordre. Les K4A et K4B vous permettent d'utiliser l'un ou l'autre type. L'utilisation d'une double empreinte au lieu de cavaliers (à souder) consomme moins d'espace sur la carte et permet d'économiser deux cavaliers (à souder). L'inconvénient est, bien sûr, que la position de l'ouverture de l'écran dans le boîtier dépend de l'écran car K4A et K4B ne sont pas au même endroit. De plus, selon le fabricant, les dimensions de cet afficheur peuvent varier. Vous devez donc choisir votre écran avant d'usiner le boîtier.

Sélection du mode de fonctionnement

L'interrupteur à glissière S1, de type DPDT (double contact inverseur), sélectionne le mode de fonctionnement du Super Servo Tester. En mode *Manuel*, il connecte la tension d'alimentation de 5 V aux connecteurs des servos. En mode *Entrées*, la tension d'alimentation du SST est V_{BATT} , donc sa propre alimentation de 5 V doit être déconnectée pour éviter les conflits. En fait, vous êtes censé déconnecter l'alimentation du SST dans ce mode, mais comme prudence est mère de sûreté, S1 la déconnecte pour vous. La diode Zener D3 ainsi que R8 garantissent que l'alimentation du reste du circuit dérivée de V_{BATT} ne dépasse pas les valeurs maximales supportées par les autres composants. La position de S1 est lue par PD4, de sorte que l'UC peut choisir le mode de fonctionnement correspondant.

À proprement parler, S1 n'a pas besoin d'être du type DPDT, mais il doit être capable de passer la puissance consommée par jusqu'à quatre servos et même plus s'il y a également un ESC ou un calculateur de vol connecté aux sorties. Les interrupteurs à glissière DPDT appropriés sont souvent moins chers que les types unipolaires, ce qui explique pourquoi ils sont utilisés ici.

Divers

Le port GPIO PD2 de l'UC commande la LED d'alarme et le buzzer. Comme ils sont connectés en parallèle, le MOSFET T1 fournit un peu de puissance supplémentaire pour les alimenter tous les deux sans surcharger l'UC. S2 est un deuxième interrupteur à glissière utilisé pour couper le buzzer, car il peut parfois s'avérer gênant. Cet interrupteur est de petite taille et de faible puissance. L'alimentation de 5 VCC pour le SST en mode *Manuel* est obtenue par un régulateur de tension linéaire classique 7805 (IC1). Veillez à utiliser un type TO220 pour ce composant, car il doit alimenter jusqu'à quatre servos. Pour limiter la hauteur de l'ensemble, on peut monter C1 et C2 couchés.

Enfin, le connecteur K3, de type boîtier, permet de programmer le microcontrôleur sans le retirer du circuit. Il est câblé de la même manière que le connecteur Arduino ICSP (*In-Circuit Serial Programming*).

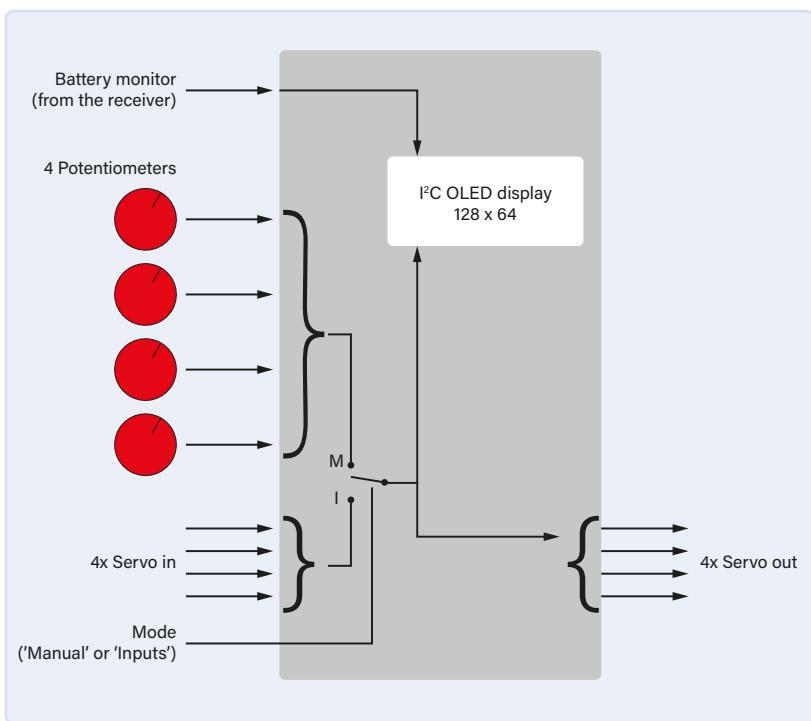


Figure 9. Vue d'ensemble de haut niveau montrant ce qui se passe à l'intérieur du microcontrôleur.

Quelques mots sur le logiciel

J'ai utilisé l'EDI Arduino pour ce projet, ce qui prouve qu'on peut l'utiliser pour des projets complexes. J'ai d'abord réalisé un démonstrateur de faisabilité avec un Arduino UNO et une platine d'expérimentation. Lorsque tout a fonctionné comme prévu, j'ai chargé le programme dans le dispositif final via le connecteur ICSP. Notez que, lorsque vous programmez le microcontrôleur via le connecteur ICSP K3, rien ne doit être relié aux connecteurs K1 et K2 car ils partagent chacun un signal avec K3. Toutes les bibliothèques nécessaires au projet sont incluses dans le téléchargement [3], à l'exception de la bibliothèque Servo, car elle fait partie de l'EDI Arduino. Le projet est organisé en trois fichiers : *Display.ino*, *Entrées.ino* et le fichier principal du croquis. Les noms des fichiers expliquent assez bien ce qu'ils contiennent. Le fichier croquis principal contient les fonctions `setup()` et `loop()` mais aussi les routines d'interruption utilisées pour minoter les impulsions d'entrée. Les changements de niveau des signaux d'entrée produisent des interruptions par changement d'état de broche. On peut alors utiliser la fonction `micros()` pour mesurer la longueur des impulsions. Les largeurs d'impulsion mesurées sont reportées sur la sortie et transmises aux fonctions d'affichage. La figure 9 présente un aperçu de haut niveau des flux de signaux à l'intérieur du microcontrôleur.

Utilisation de la bibliothèque OLED_I2C

Comme mentionné précédemment, l'écran OLED est commandé par un bus I²C. Sur l'ATmega328P, ce bus est partagé avec le CA/N. PC4 peut être soit la broche SDA soit l'entrée analogique A4. Pour contourner ce problème, le bus I²C est émulé par logiciel. La bibliothèque *OLED_I2C* de Rinky-Dink Electronics [2] le fait et nous permet d'assigner n'importe quel port GPIO au bus I²C.

J'ai dû modifier légèrement le fichier *OLED_I2C.cpp* de cette bibliothèque car une directive `include` incorrecte empêchait sa compilation. Remplacer la ligne 25 :

```
#include "hardware/avr/HW_AVR.h"
```

par

```
#include "HW_AVR.h"
```

Aussi, dans le fichier *HW_AVR.h*, dans la fonction `OLED::update()`, j'ai commenté la ligne 24, comme ci-dessous :

```
//noInterrupts();
```

Ceci est nécessaire car la bibliothèque Servo utilise les interruptions pour générer les signaux MLI à 50 Hz sur les sorties du SST. Elles doivent également rester activées pour mesurer les impulsions entrantes.

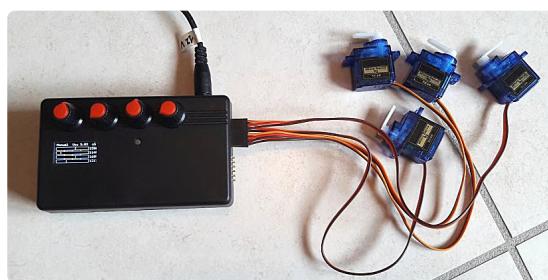


Figure 10. Le prototype de l'auteur commandant quatre servos en mode Manuel.

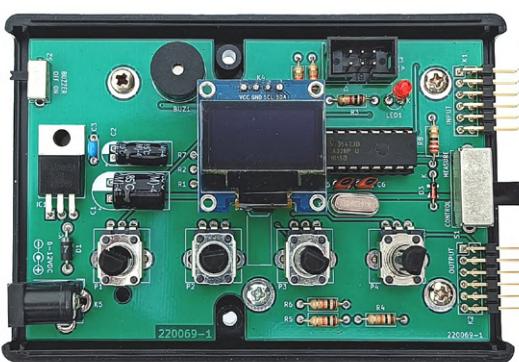
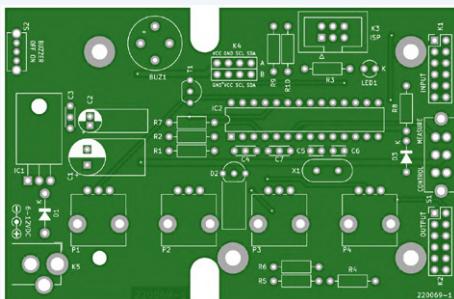


Figure 11. Le prototype d'Elektor Labs monté dans son boîtier.



LISTE DES COMPOSANTS



Résistances (traversantes, 0,25 W, 5%)

R1, R3 = 1 kΩ

R2, R4, R5, R6, R7, R9, R10 = 10 kΩ

R8 = 22 Ω

P1, P2, P3, P4 = 10 kΩ, linéaire, vertical

Condensateurs (traversants)

C1 = 100 µF, 35 V, diamètre 8 mm, pas 3,5 mm

C2 = 10 µF, 16 V, diamètre 5 mm, pas 2 mm

C3, C4, C7 = 100 nF, pas 2,5 mm ou 5 mm

C5, C6 = 22 pF, pas 2,5 mm

Semiconducteurs (traversants)

D1 = 1N5817

D2 = LM385Z-2.5, TO-92

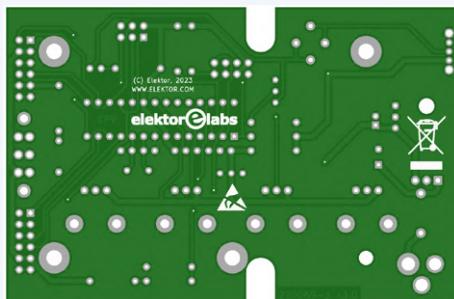
D3 = BZX79-C5V1

IC1 = 7805, TO220

IC2 = ATmega328-P

LED1 = LED, 3 mm, rouge

T1 = 2N7000, TO-92



Divers (traversants)

BUZ1 = Buzzer piézoélectrique avec oscillateur, diam. 12 mm, pas 6,5 ou 7,6 mm

K1, K2 = Connecteur à 12 broches coudées à 90° sur 2 rangées, pas 2,54 mm

K5 = Connecteur baril

K4* = Connecteur à 4 broches, pas 2,54 mm (montage A ou B selon l'écran)

K3 = Connecteur à 6 broches sur 2 rangées, pas 2,54 mm

S1 = Interrupteur à glissière DPDT, 90° (par ex. MFS201N-16-Z)

S2 = Interrupteur à glissière SPDT, 90° (par ex. OS102011MA1)

X1 = Quartz, 16 MHz, HC-49S (profil bas)

Écran OLED, 2,4 cm, 128 x 32 pixels, 4 broches I²C (par exemple www.elektor.fr/18747)

Elektor PCB 220069-1

Boîtier suggéré : Hammond 1593N

* = voir le texte

Avant la mise sous tension

Si le potentiomètre P1 est à sa position minimale (tourné vers la gauche) à la mise sous tension, l'écran affiche un cadre blanc. Cela permet de positionner la découpe pour l'écran et d'ajuster l'ouverture si nécessaire. Tout le monde ne dispose pas d'une machine CNC à la maison. Tournez P1 vers le centre pour revenir à un fonctionnement normal. De même, si P4 est dans sa position minimale à la mise sous tension, l'affichage utilisera le mode empilé. S'il est dans sa position maximale, c'est le mode en carré qui sera activé. Dans ce deuxième cas, cela signifie que le moteur connecté à la sortie 4 sera à pleine vitesse au démarrage.

C'est le moment de décoller

Concevoir et développer le Super Servo Tester a été amusant et c'est un outil utile à avoir sous la main si vous aimez les drones, les quadrioptères ou les modèles RC en général. Non seulement ce projet m'a permis d'améliorer

mes compétences en programmation, mais je peux enfin sortir mon drone, qui attend impatiemment de décoller !

VF : Denis Lafourcade — 220069-04

Des questions, des commentaires ?

Contactez Elektor (redaction@elektor.fr).



Produits

➤ **Arduino Uno R3 (SKU 15877)**
www.elektor.fr/15877

➤ **Écran OLED, 0,96 pouce, 128x32 pixels (I²C)
(SKU 18747)**
www.elektor.fr/18747

LIENS

[1] Site web du drone de Joop Brokking : http://www.brokking.net/ymfc-al_main.html

[2] Bibliothèque OLED_I2C : <http://www.rinkydinkelectronics.com/index.php>

[3] Téléchargements du projet sur Elektor Labs : <https://www.elektormagazine.fr/labs/super-servo-tester>



signaux analogiques et microcontrôleurs

CA/N, CN/A, mesure du courant, et plus

Mathias Claussen (Allemagne)

Alors qu'un µc traite en interne des uns et des zéros, le monde qui l'entoure est analogique. Il est donc parfois nécessaire de lire ou d'émettre des valeurs analogiques. Quelques composants suffisent pour connecter un µc au monde analogique.

Que votre microcontrôleur (µc) soit un Arduino UNO, un STM32 ou autre, il faut souvent traiter plus que des signaux numériques. Le monde qui nous entoure est analogique ; il n'y a pas seulement de la lumière ou de l'obscurité, mais aussi toutes les nuances entre les deux. C'est pourquoi cet article donne un aperçu de la manière dont les valeurs analogiques peuvent être lues aussi bien qu'émises par un microcontrôleur. Pour les développeurs expérimentés, ce texte appartient certainement à la catégorie « rudimentaire », mais pour chacun d'entre nous il faut un début à tout. Et s'il est très facile de configurer et lire des broches numériques, les choses se compliquent lorsqu'il s'agit des entrées/sorties de valeurs analogiques.

Les questions fondamentales sont : Comment introduire des valeurs analogiques dans un µc à fonctionnement numérique et comment doit-on les conditionner pour cela ? Et comment un µc peut-il émettre des valeurs analogiques ? Pour répondre à ces questions, examinons d'abord deux éléments de base importants, le convertisseur analogique-numérique (CA/N) et le convertisseur numérique-analogique (CN/A).

Convertisseur analogique-numérique (CA/N)

Un convertisseur analogique-numérique (CA/N) est capable de convertir un signal analogique en une représentation numérique. Cette représentation numérique peut ensuite être utilisée par le microcontrôleur pour traiter les données. Un CA/N est fréquemment qualifié par deux paramètres, sa résolution en bits et sa fréquence d'échantillonnage en échantillons par seconde. Dans les microcontrôleurs, la résolution est généralement comprise

entre 10 et 12 bits. Cela signifie que le CA/N subdivise sa plage de tension d'entrée analogique en 1024 (10 bits) ou 4096 valeurs (12 bits) et affecte ensuite la tension appliquée à l'une de ces valeurs. La plage de valeurs est désormais finie et le signal analogique doit être affecté à la valeur numérique la mieux adaptée, même si elle ne correspond pas exactement au niveau déterminé par calcul. Cet écart est appelé erreur de quantification.

La seconde quantité, la fréquence d'échantillonnage en échantillons par seconde, indique combien de fois par seconde le CA/N est capable de convertir une valeur analogique en une valeur numérique. À moins que le signal ne soit une tension continue, la fréquence d'échantillonnage détermine également la bande passante maximale du signal qui peut être traitée. L'inverse de la fréquence d'échantillonnage indique le temps nécessaire au CA/N pour effectuer une conversion analogique-numérique – en d'autres termes, le temps que le logiciel doit attendre après le début d'une conversion avant que la valeur résultante puisse être traitée.

Plage de tension, résolution et tension de référence

Si nous supposons une plage de tension d'entrée de 0 à 5 V avec une résolution de 10 bits, nous obtenons

$$5 \text{ V} / (2^{10} \text{ bit} - 1) = 4,89 \text{ mV/bit}$$

Une tension de 452 mV, par exemple, se verrait ainsi attribuer en interne la valeur 92

$$452 \text{ mV} / 4,89 \text{ mV/bit} = 92$$

et serait donc traitée comme 449,88 mV. L'erreur de quantification est donc de 2,12 mV. L'écart ne peut être supérieur à 0,5 LSB (bit de poids faible), ce qui correspond à un peu moins de 2,5 mV dans cet exemple.

Si la plage de tension à mesurer est sensiblement inférieure à la plage de tension d'entrée autorisée du CA/N, on peut sur la plupart des CA/N modifier cette plage. Le CA/N compare toujours les valeurs analogiques à une tension de référence. De nos jours, de nombreux µc fournissent une tension de référence interne, voire plusieurs. L'AVR128DB de Microchip servira d'exemple. La **figure 1** montre le schéma fonctionnel de la référence de tension de ce contrôleur.

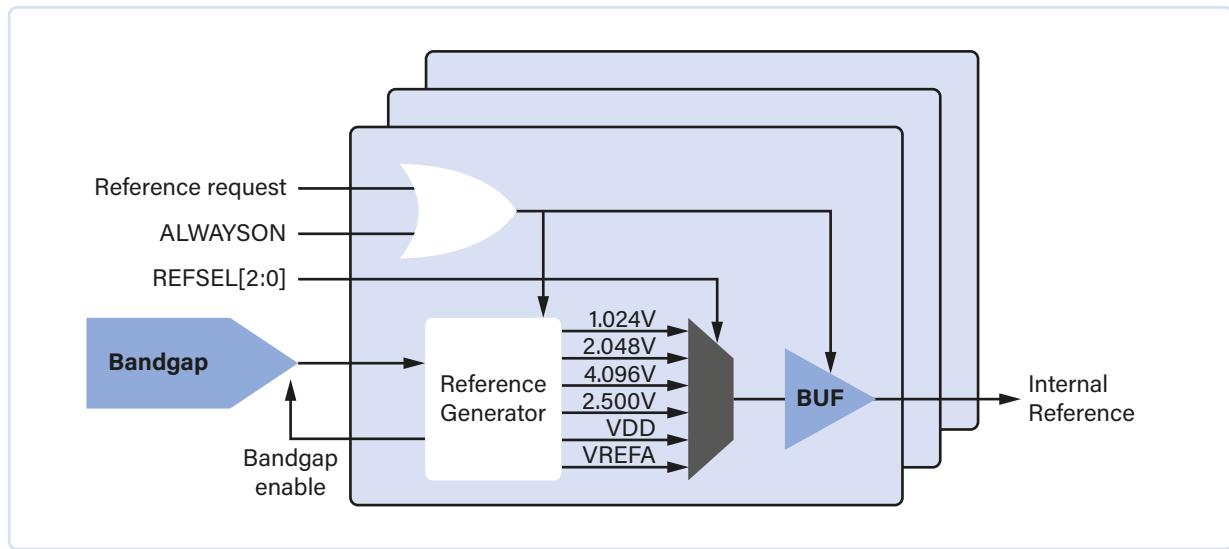


Figure 1. Référence de tension de l'AVR128DB (Source : Microchip, AVR® DB Family datasheet [PDF] - <https://elektor.link/AVR128DBDatasheet>).

Six tensions de référence de 1,024 V, 2,048 V, 4,096 V, 2,5 V, VDD et V_{REFA} sont disponibles. Les trois premières tensions peuvent être représentées par 2^N et permettent de convertir facilement les valeurs numériques du CA/N en volts par la suite. Un élément que l'on retrouve sur presque tous les µc est la différence entre GND (masse) et AGND (masse pour la partie analogique du µc). Comme il peut y avoir du bruit à haute fréquence dans le domaine numérique qui passe aussi par les liaisons de masse, tous les CA/N ou CN/A ont leur propre masse. Ainsi, dans la conception du circuit, on peut découpler l'une de l'autre GND et AGND pour éviter les interférences. S'il est prévu que la quantité mesurée ne dépasse pas 1 V, on peut choisir 1,024 V comme référence. Il en résultera une résolution de 1,00098 mV/bit. 452 mV correspondent à la valeur numérique 452 ; lors du traitement ultérieur, l'erreur de quantification serait donc nettement plus faible que dans l'exemple ci-dessus. Cependant, le choix approprié de la tension de référence ne suffit pas pour garantir que les résultats des conversions soient plus proches de la valeur physique (analogique). La qualité de cette tension influe également.

Dans l'AVR128DB, VDD et V_{REFA} peuvent également être considérées comme des tensions de référence. VDD est la tension d'alimentation de la puce, et V_{REFA} est une broche externe de l'AVR128DB qui vous permet de connecter votre propre source de tension de référence externe. L'utilisation de VDD comme référence peut être un choix malheureux, car cette tension est chargée de bruit provenant de la partie numérique du µc, et il est difficile de « filtrer » ce bruit. La situation est différente avec V_{REFA} : ici, on peut utiliser comme référence une source de tension externe de haute précision. Bien sûr, elle est toujours connectée à la tension d'alimentation « sale », mais s'agissant d'un composant externe, on peut bien mieux la filtrer. Une référence stable et sans interférence est la base d'une bonne conversion analogique-numérique.

Il existe une autre façon d'augmenter la résolution d'un CA/N sans modifier la tension de référence : le suréchantillonnage. Le suréchantillonnage utilise le bit de poids faible, qui passe de zéro à un si la valeur du signal ne peut pas correspondre exactement

à la résolution disponible du CA/N. En prenant la moyenne de ce bit de poids faible, on obtient une fraction de un, ce qui signifie une résolution plus élevée. Toutefois, cette méthode réduit la fréquence d'échantillonnage effective. Pour chaque bit ajouté par le suréchantillonnage, la fréquence d'échantillonnage est divisée par quatre. Si, par exemple, un CA/N de 12 bits peut traiter un maximum de 130 ksps (kilo-échantillons par seconde), quatre bits de suréchantillonnage, c'est-à-dire 16 bits au total, ne permettent d'obtenir que 507 échantillons par seconde :

$$130 \text{ ksps} / 4^4$$

En outre, selon le CN/A, le processeur doit effectuer le calcul de la moyenne nécessaire au suréchantillonnage, ce qui représente une charge de calcul supplémentaire. Certains CA/N plus récents, comme ceux de certaines familles de contrôleurs STM32, effectuent cette tâche de manière totalement autonome, sans charge supplémentaire pour le processeur.

Les CA/N se présentent sous différentes formes. Les CA/N les plus courants dans les contrôleurs sont du type SAR (approximation successive). Pour les résolutions élevées de 24 bits et plus à des taux d'échantillonnage relativement faibles (de l'ordre du ksps à quelques échantillons par seconde), les CA/N delta-sigma sont les plus couramment utilisés. Les détails de la conversion delta-sigma et sa mise en œuvre dans un CPLD/FPGA (Complex Programmable Logic Device / Field-Programmable Gate Array) sont décrits dans un article d'Elektor [1]. Il existe d'autres types de CN/A, dont le type Flash ADC qui offre le temps de conversion le plus rapide, raison pour laquelle il est souvent utilisé dans les mémoires flash modernes.

Convertisseur numérique-analogique (CN/A)

Avec un convertisseur numérique-analogique, vous générerez à nouveau une valeur analogique à partir d'une valeur numérique. Moins courants que les CA/N, certains µc intègrent des CN/A. Sur l'AVR128DB, il s'agit d'un convertisseur numérique-analogique de

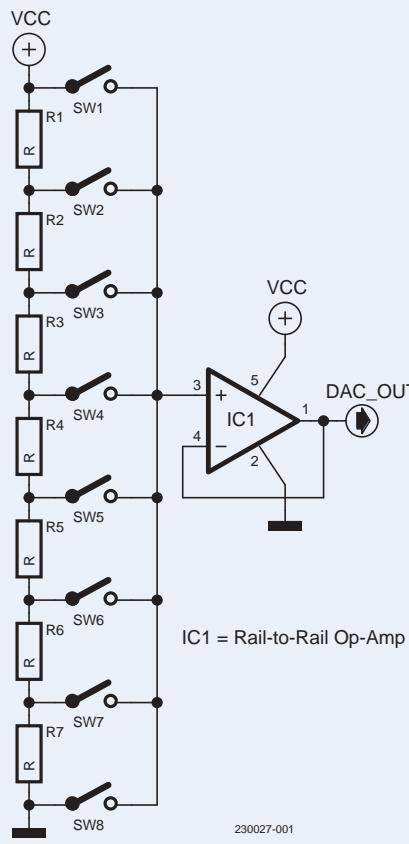


Figure 2. Structure d'un CN/A à réseau.

10 bits, qui peut donc délivrer 1024 tensions différentes. S'il n'y a pas de CN/A dans le µc, on peut utiliser un CN/A externe comme le MCP4922 de Microchip. Les deux sont des CN/A à réseau de résistances, le MCP4922 est un CN/A 12 bits à deux canaux. Alors qu'un CN/A à réseau de résistances peut être bien intégré dans une puce, il est difficilement réalisable avec des résistances discrètes. Un coup d'œil au CN/A à chaîne de résistances à trois bits de la **figure 2** en donne une idée : le CN/A nécessite

$$2^N - 1 \quad (N = \text{nombre de bits})$$

résistances et N interrupteurs. Pour un CN/A de 16 bits, il s'agirait donc de 65 535 résistances et 65 536 interrupteurs !

Tout comme un CA/N, un CN/A nécessite également une tension de référence qui spécifie la plage dans laquelle la tension de sortie peut évoluer. Cependant, le CN/A lui-même ne peut piloter que quelques milliampères, même si la sortie est tamponnée par un ampli-op.

En revanche, s'il n'y a pas de CN/A interne et que l'on dispose encore de suffisamment de broches sur un µc, on peut construire un CN/A externe autrement. L'une des options est un CN/A à pondération binaire, comme le montre la **figure 3**. Ici, seules N+1 résistances et un ampli-op sont nécessaires pour N bits.

Les valeurs des résistances sont données par

$$R_N = R_{(N-1)} \cdot 2$$

et sont donc doublées à chaque fois. Un tel CN/A peut être utilisé pour générer des signaux VGA analogiques, par exemple, comme le montre la fiche technique [2].

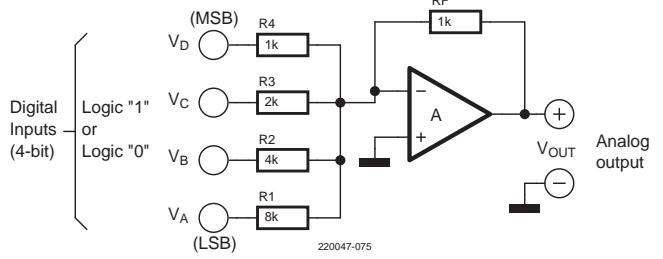


Figure 3. Circuit d'un CN/A à pondération binaire.

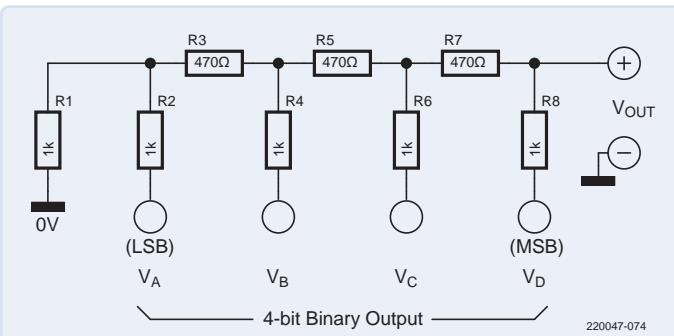


Figure 4. Principe d'un CN/A R-2R.

Une autre variante fréquemment utilisée est le CN/A R-2R. Contrairement à un CN/A à pondération binaire, N × 2 résistances et un ampli-op sont nécessaires pour N bits, mais il n'y a que deux valeurs de résistance, alors que dans un CN/A à pondération binaire N, il y a surtout des valeurs fractionnaires gênantes. Le schéma d'un tel CN/A est illustré à la **figure 4**.

Malgré tout, s'il y a pénurie de broches d'E/S sur le µc utilisé, avec quelques circuits externes l'unité MLI peut également servir de convertisseur numérique-analogique (CN/A). La fonction `analogWrite()` de l'Arduino UNO produit des valeurs analogiques de cette manière en utilisant les broches numériques.

La **figure 5** montre un exemple tiré de l'article d'Elektor [3] pour un Arduino UNO.

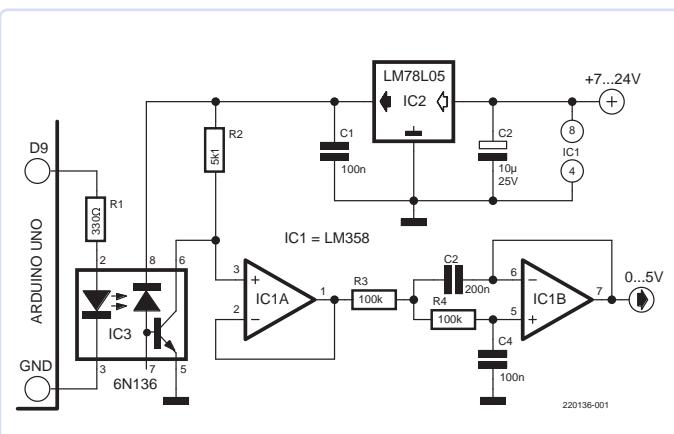


Figure 5. Sortie analogique isolée pour l'Arduino UNO.

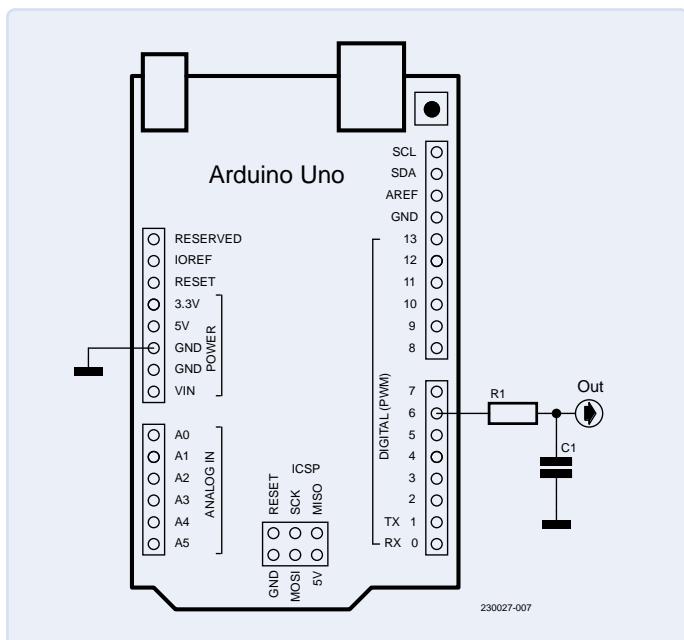


Figure 6. Arduino UNO avec filtre passe-bas RC.

Une configuration plus simple est le circuit de la **figure 6**, qui consiste simplement en un filtre passe-bas RC pour lisser le signal. Cependant, un CN/A comme celui-ci présente des inconvénients. Comme le signal MLI doit être lissé, les valeurs de R et C doivent être calculées pour l'application souhaitée. En outre, il y aura toujours un petit résidu de MIL dans le signal analogique, qui sera perçu comme un bruit indésirable.

Conditionnement du signal

Les signaux à mesurer ou à émettre ne sont pas toujours directement compatibles avec le µc choisi. Dans le monde numérique, de nombreux développeurs connaissent le problème de la connexion de composants 5 V à des systèmes en 3,3 V. Des problèmes similaires se posent également dans le monde analogique et ses interfaces. L'audio, la vidéo, ainsi que les boucles de courant 4-20 mA ne sont que quelques exemples qui nécessitent un conditionnement du signal. La forme la plus simple de conditionnement du signal est probablement le diviseur de tension, qui peut être utilisé non seulement pour les valeurs analogiques, mais aussi pour adapter les signaux numériques.

Diviseurs de tension et CA/N

La **figure 7** montre un diviseur de tension non chargé, qui est une forme idéale du diviseur de tension. Lorsqu'une tension (U) est appliquée, un courant (I) apparaît au niveau de $R_1 + R_2$. Une tension $R_1 \times I = U_1$ s'établit maintenant aux bornes de la résistance R_1 et une tension $R_2 \times I = U_2$ aux bornes de R_2 , où $U = U_1 + U_2$. Si vous voulez maintenant mesurer, par exemple, une plage de 0 à 10 V avec un CA/N qui ne peut traiter que 0 à 5 V, la tension doit être divisée par deux. Il est évident qu'il faut pour cela que $R_1 = R_2$. Cependant, le courant (I) doit également être pris en compte lors du choix des valeurs de résistance. Bien que les valeurs de 10Ω pour R_1 et R_2 donnent mathématiquement les tensions souhaitées pour U_1 et U_2 , un courant de 500 mA circule à travers les résistances. Cela pourrait non seulement surcharger la source de tension, mais aussi les résistances elles-mêmes. En effet, une dissipation de puissance de $P = R \times I^2 = 2,5 \text{ W}$, que chaque résistance doit dissiper sous forme de chaleur, est beaucoup trop élevée pour la plupart des petits modèles.

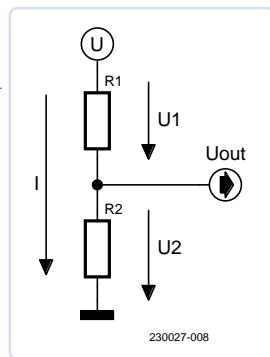


Figure 7. Diviseur de tension non chargé.

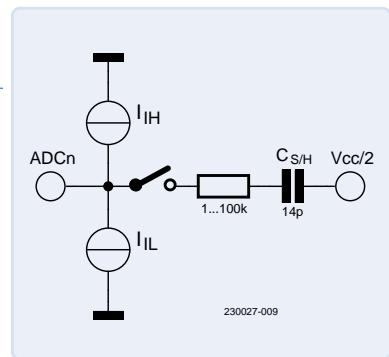


Figure 8. Structure interne de l'entrée CA/N de l'ATmega328P (Source : Microchip, Atmel Atmega328P datasheet [PDF] - <https://elektor.link/ATmega328PDatasheet>).

Une autre option pour réduire la dissipation de puissance serait d'utiliser des résistances de $1 \text{ M}\Omega$. Dans ce cas, un courant de seulement $5 \mu\text{A}$ circulerait et la dissipation de puissance serait de $25 \mu\text{W}$, mais cet autre extrême pose un nouveau problème. Si un étage d'échantillonnage et de maintien est installé dans le CA/N, comme dans l'ATmega328 illustré à la **figure 8**, cela peut conduire à des valeurs mesurées incorrectes. Il y a un petit condensateur qui est chargé à l'entrée du CA/N. Lors d'une conversion, le condensateur est connecté à l'entrée analogique pendant un court instant (pour la durée du temps d'échantillonnage configuré) et chargé. Si le courant est trop faible, le condensateur peut ne pas être complètement chargé pendant le temps d'échantillonnage. Si le CA/N effectue la conversion avec le condensateur incomplètement chargé, le résultat sera incorrect.

Un coup d'œil à la fiche technique du µc choisi devrait aider à déterminer les valeurs appropriées. Ici pour l'AVR, une valeur de $4,7 \text{ k}\Omega$ pour R_1 et R_2 et donc un courant de $1,06 \text{ mA}$ seraient appropriés.

Tensions négatives

Les tensions positives sont assez faciles à mesurer, par exemple à l'aide d'un diviseur de tension. Mais que se passe-t-il si la tension à mesurer est négative ? En utilisant un ampli-op en amplificateur

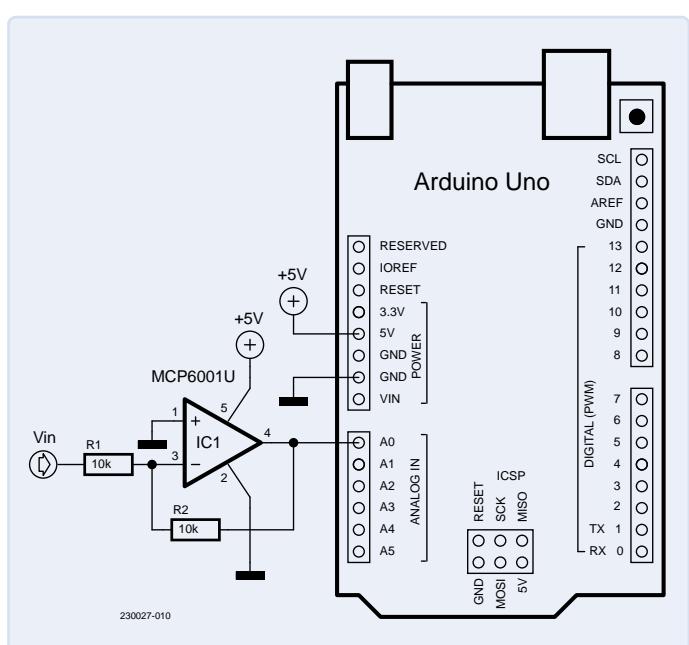


Figure 9. Amplificateur inverseur.

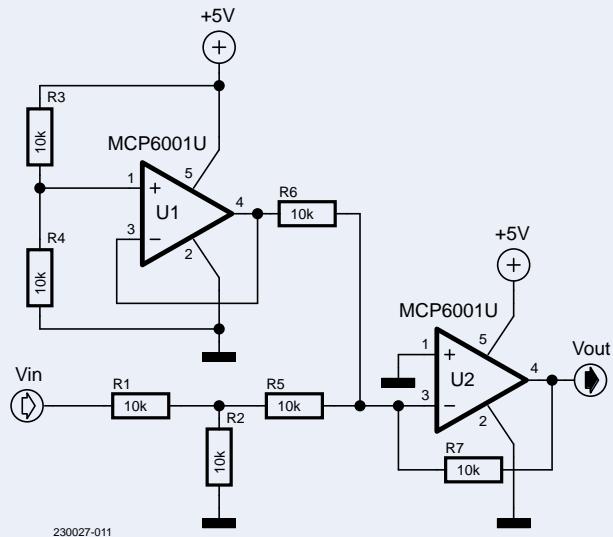


Figure 10. Décalage CC avec des ampli-op.

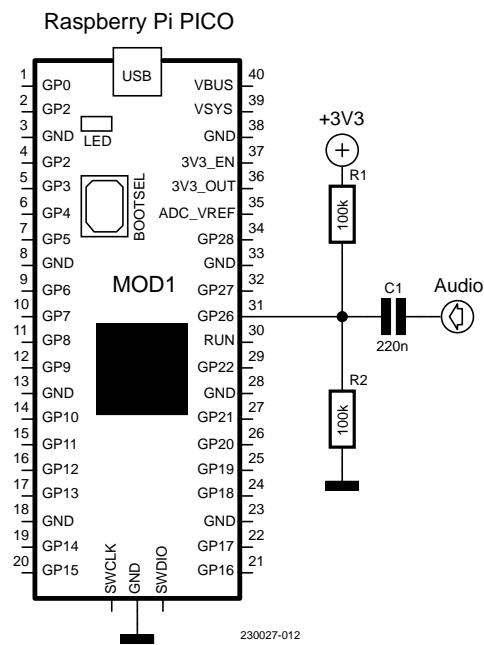


Figure 11. Entrée audio pour le CA/N d'un AVR Microchip.

inverseur et une paire de résistances comme le montre la **figure 9**, on peut inverser la tension.

Tant que la tension à l'entrée de ce circuit reste inférieure à 0 V, l'ampli-op fonctionne correctement. Cependant, si une plage de tension de -5 V à +5 V doit être couverte, il faut se donner un peu plus de mal. En supposant que la plage d'entrée autorisée du contrôleur est de 0 à 5 V, le signal avec une amplitude de 10 V doit d'abord être divisé par deux par un diviseur de tension R1 / R2. Nous devrions donc traiter des valeurs comprises entre -2,5 V et +2,5 V. Cette plage est maintenant décalée de 2,5 V par le diviseur de tension R3 / R4. Le deuxième ampli-op non inverseur, U1, tamponne cette tension. Le schéma complet du circuit est illustré à la **figure 10**.

Lors de l'évaluation ultérieure des valeurs mesurées, il faut tenir compte du fait que le point zéro de la tension d'entrée est

maintenant augmenté de la moitié de la tension d'alimentation, c'est-à-dire de 2,5 V dans cet exemple.

Signaux audio et CN/A

Une application intéressante pour un µc est le traitement des signaux audio. Selon la source qui fournit le signal, les signaux audio courants ont une tension crête à crête comprise entre 0,894 V_{PP} (-10 dBV) et 3,472 V_{PP} (+4 dBV). La plupart des appareils non professionnels fonctionnent à -10 dBV, soit 0,894 V_{PP}. Le signal a donc une tension minimale de -0,477 V et maximale de +0,477 V. Le CA/N d'un µc tel que le Raspberry Pi RP2040 ne peut pas gérer cela. Mais pourquoi traiter de l'audio avec un CA/N 10 bits ? Par exemple, l'AVR et les µc plus rapides sont tout à fait capables de calculer et d'afficher le spectre de fréquence d'un signal audio. Pour cette application, nous pouvons profiter du fait que la partie CC n'est pas intéressante ; seule la partie CA du signal audio est nécessaire (C1) et pourvue d'un décalage CC de la moitié de la tension d'alimentation du CN/A (R1 / R2) pour éléver le signal audio dans la plage positive. Un schéma de principe est présenté à la **figure 11**. Avec un tel signal, il faut tenir compte de la plage de tension (assez étroite). Avec une tension maximale inférieure à 1 V, le signal doit être amplifié ou la tension de référence interne de l'ADC doit être choisie de manière appropriée afin d'éviter des erreurs de quantification inutilement importantes.

Mesure du courant

Un CA/N ne peut généralement mesurer que des tensions. Mais dans de nombreux cas, il faut également déterminer les courants. La méthode la plus simple consiste à utiliser une résistance shunt. Si la consommation de courant d'un circuit doit être mesurée, il existe deux options : la mesure côté bas et la mesure côté haut. La **figure 12** montre un circuit très simple pour une mesure côté haut à l'aide de l'Arduino Nano.

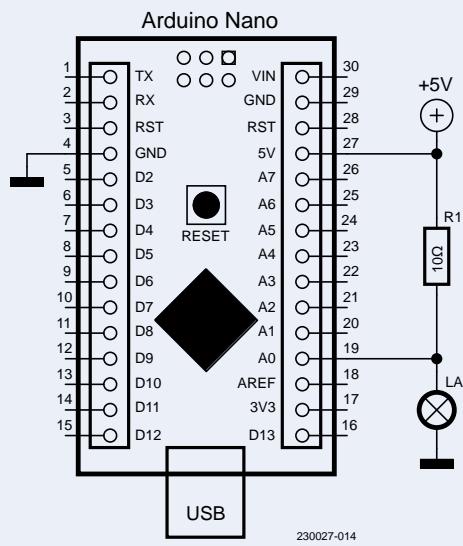


Figure 12. Mesure du côté haut.

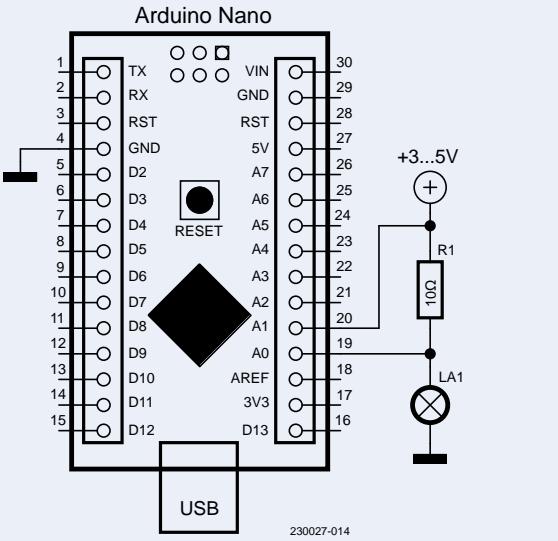


Figure 13. Mesure du côté haut avant et après la résistance de shunt.

La résistance shunt est placée entre la tension d'alimentation positive et la charge. Pour déterminer le courant qui traverse la résistance et donc la charge, il faut mesurer la chute de tension aux bornes de la résistance. Si la tension (constante et connue) fournie à la charge est inférieure ou égale à la tension de référence du CA/N, on peut déterminer sans difficulté la chute de tension dans le shunt. Cependant, si la tension d'alimentation de la charge est variable (mais toujours inférieure à la tension de référence du CA/N), les tensions des deux côtés de la résistance de shunt doivent être mesurées, comme le montre la **figure 13**.

Certes, cet exemple est très idéalisé car la tension d'alimentation de la charge est inférieure ou égale à la tension de référence du CA/N. Les choses se compliquent si la tension d'alimentation est supérieure à la tension de référence du CA/N ou même du µc.

Contrôle d'une lampe halogène à l'aide d'une mesure de courant

Dans l'exemple suivant, nous voulons mesurer et surveiller la consommation de courant d'une lampe halogène G4 à gradation, alimentée par une tension continue maximale de 12 V. La lampe est autorisée à « consommer » un maximum de 20 W (1,67 A à 12 V). La lampe est considérée comme défectueuse par le µc pour les courants supérieurs à 2 A ou inférieurs à 25 mA.

Nous pourrions maintenant utiliser des diviseurs de tension pour amener la tension avant et après la résistance de shunt dans une plage qui puisse être traitée par le CA/N. La **figure 14** montre un circuit qui a été modifié en conséquence.

Le µc est alimenté en 3,3 V et les résistances sont choisies pour que, à 12 V, un maximum de 2,6 V soit appliqué aux broches du CA/N. La valeur de la résistance shunt est de 0,1 Ω. À 2 A, elle provoque une chute de tension de 0,2 V, mais dissipe encore 0,4 W sous forme de chaleur. La méthode fonctionne en principe, mais elle n'est pas très astucieuse, comme nous le verrons dans un instant.

Le diviseur de tension réduit encore la plage de mesure, de sorte que le CA/N du µc doit pouvoir gérer une chute de tension maximale de moins de 0,05 V à 2 A. Ce n'est pas une bonne idée. De plus, le fait que la chute de tension soit mesurée via deux canaux CA/N ne facilite pas le traitement ultérieur.

Dans ce cas, l'utilisation d'un amplificateur à détection de courant côté haut, tel que l'INA138 de Texas Instruments ou le MCP6C02

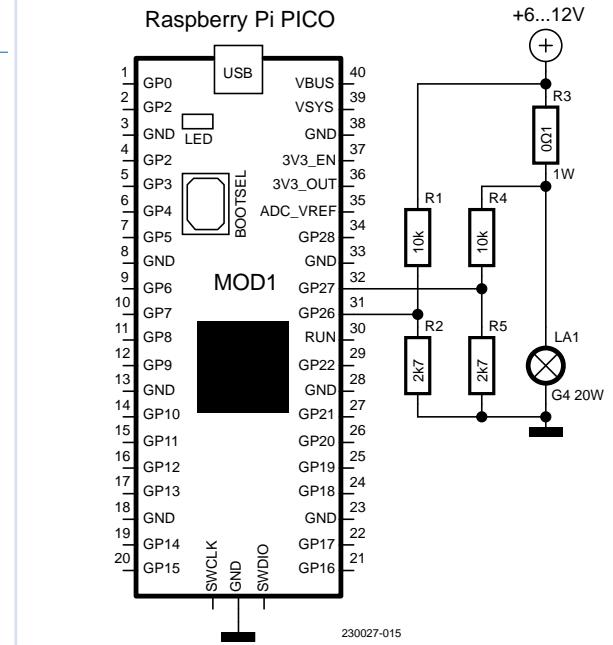


Figure 14. Circuit modifié avec des diviseurs de tension.

de Microchip, est une bien meilleure solution. En outre, on peut ici réduire considérablement la valeur de la résistance shunt. Le shunt de 10 mΩ entraîne une dissipation de chaleur nettement plus faible que son homologue dix fois plus grand. La **figure 15** montre le circuit avec l'amplificateur à détection de courant. La tension de sortie est proportionnelle à la chute de tension dans la résistance et peut donc être facilement traitée avec un canal CA/N.

Mesure côté bas

Au lieu de placer une résistance de shunt directement devant la charge, on peut également la connecter entre la charge et la masse.

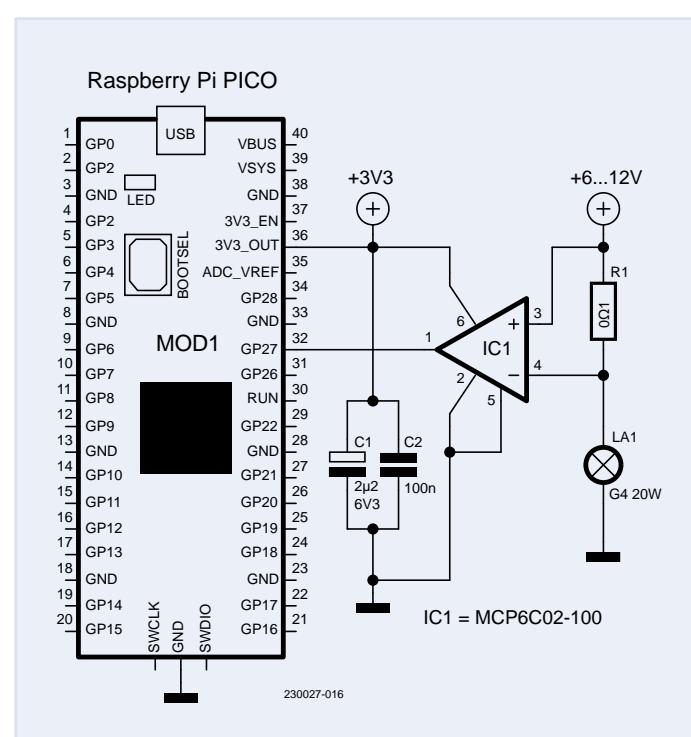


Figure 15. Circuit avec amplificateur de mesure du courant.

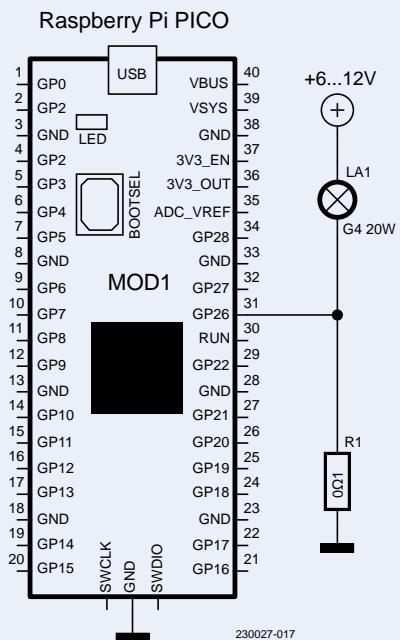


Figure 16. Mesure du côté bas.

Restons sur l'exemple de la lampe halogène. La **figure 16** montre une telle mesure du côté bas.

Ici, on utilise une résistance de $0,1\ \Omega$ qui provoque une chute de tension de $0,2\ V$ à $2\ A$. L'avantage est que les niveaux de tension à mesurer sont inférieurs à ceux de la tension de référence du CA/N et de la tension d'alimentation du µc. Cependant, à $25\ mA$, seuls $2,5\ mV$ peuvent être mesurés ($0,1\ mV$ par mA). Dans des conditions idéales, le CA/N 12 bits avec une tension de référence de $3,3\ V$ résoudrait encore $0,8\ mV$ par bit. Toutefois, il convient d'examiner le nombre effectif de bits (ENOB) du CA/N, qui prend en compte les erreurs de mesure qui se produisent en pratique à cause de l'erreur de quantification, de la distorsion, du bruit et d'autres facteurs défavorables similaires. Pour le RP2040, un ENOB d'environ 9 est spécifié, ce qui signifie environ $6,44\ mA$ par bit. Pour cette application, la résolution de la mesure du courant est à peu près suffisante. Cependant, si une résolution plus fine est nécessaire, il peut être utile d'utiliser un amplificateur optique, ce qui permettrait également d'utiliser une valeur de résistance plus petite.

Amplificateurs opérationnels et capteurs de température

Des bibliothèques entières ont été écrites sur les ampli-op. Lorsqu'ils sont correctement câblés, ces composants fondamentaux et polyvalents ont une myriade d'applications, dont certaines ont déjà été mentionnées dans cet article. Lorsqu'il s'agit de mesurer des tensions, il n'y a pas que les résistances shunt pour lesquelles les tensions se situent dans une plage qui n'est pas idéale pour le CA/N d'un µC. Les capteurs de température sont également concernés par ce problème, du moins s'il s'agit de capteurs purement analogiques, résistifs comme le PT100 ou les thermocouples composés de deux métaux différents, contrairement à ceux qui disposent d'une interface numérique comme le DS18B20 ou le DHT11.

Thermocouples

Ces éléments constitués de deux métaux agissent comme des sources de tension en fonction de la température. Le thermocouple de type K, très répandu, sert d'exemple. À 21 °C ou 69,8 °F, un tel composant délivre une tension de 0,838 mV en raison de l'effet Seebeck ; à 22 °C ou 71,6 °F, cette tension est de 0,879 mV. La différence de seulement 40 µV par degré met le CA/N d'un MCU à rude épreuve. En outre, l'utilisation de thermocouples nécessite également une compensation de la soudure froide [4].

En revanche, on peut utiliser un ampli-op pour amplifier la tension d'un thermocouple. La **figure 17** montre un exemple d'un tel circuit mis en œuvre pour un Arduino Nano Every, où une plage de température de 0 °C à 400 °C est attendue (0 à 16,396 mV de tension du thermocouple).

Ici, l'ampli-op non inverseur est utilisé avec un gain de $A = 200$. On a ajouté un filtre RC pour minimiser les interférences à l'entrée de l'ampli-op ; un simple capteur de température numérique LM75 se charge de la détection de la température pour la compensation de la soudure froide.

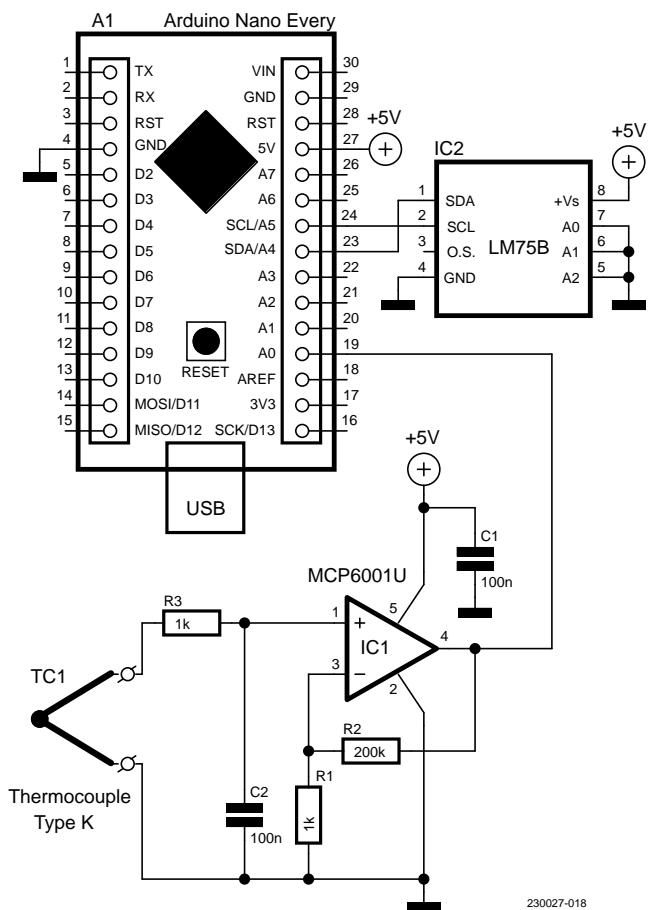


Figure 17. Lecture d'un thermocouple avec l'Arduino Nano Every.

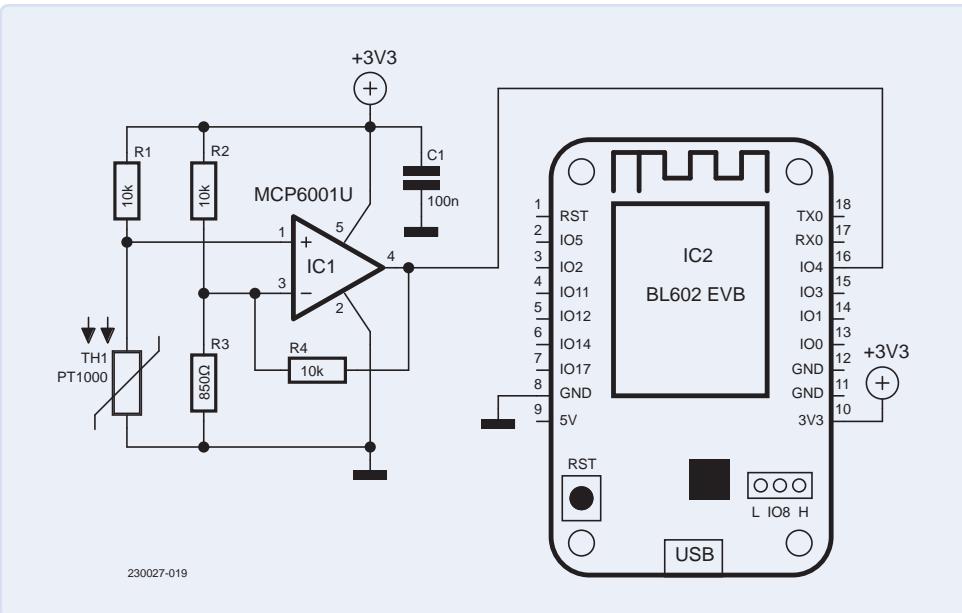


Figure 18. Pont de Wheatstone pour une PT1000.

PT100 et PT1000

Les PT100 et PT1000 sont des résistances qui dépendent de la température, avec une résistance de $100\ \Omega$ (PT100) ou de $1000\ \Omega$ (PT1000) à $0\ ^\circ\text{C}$. Pour calculer la température, il faut d'abord mesurer la valeur de la résistance du capteur. Un pont de Wheatstone constitue une bonne option pour déterminer avec précision la valeur de la résistance. Comme pour les thermocouples, les tensions sont assez faibles avec les PT100 et PT1000, et la petite différence entre deux points de mesure doit être déterminée. Là encore, un ampli-op câblé comme un amplificateur différentiel non inverseur fournit l'aide nécessaire. Le circuit pour une carte avec une tension d'alimentation de 3,3 V, telle que la carte d'évaluation Pinecone BL602, est illustré à la **figure 18**.

On amplifie la différence de tension entre l'entrée positive et l'entrée négative de l'ampli-op. Mais que se passe-t-il si la température descend en dessous de $0\ ^\circ\text{C}$? Dans ce cas, l'ampli-op devrait délivrer une tension négative, ce qui n'est pas possible. Une solution consiste à remplacer la résistance R3 de $10\ k\Omega$ par une résistance de $850\ \Omega$. La tension entre R1 et R2 serait alors de $0,3917\ \text{V}$, ce qui s'appliquerait également au point situé entre R3 et le PT1000 à une température de $-38\ ^\circ\text{C}$. Ainsi, la fenêtre de mesure commence à $-38\ ^\circ\text{C}$ avec une différence de tension de $0\ \text{V}$. À $20\ ^\circ\text{C}$, la différence de tension dans le circuit serait de $0,09444\ \text{V}$.

Résumé

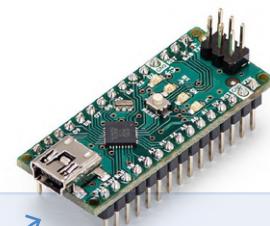
Le traitement et la sortie de valeurs analogiques peuvent être réalisés avec un microcontrôleur sans trop d'efforts. Même si l'ensemble ne semble pas aussi simple que de commander une broche d'E/S

numérique, la manipulation du CA/N est quelque chose que tout le monde devrait essayer. Les exemples présentés ici sont des suggestions pour vos propres tentatives de faire interagir le microcontrôleur avec le monde analogique à l'extérieur de son boîtier. ↗

VF : Denis Lafourcade — 230027-04

Des questions, des commentaires ?

Contactez Elektor (redaction@elektor.fr).



Produits

- **Arduino Nano (SKU 17002)**
www.elektor.fr/17002
- **Raspberry Pi Pico RP2040 (SKU 19562)**
www.elektor.fr/19562
- **Carte d'évaluation Pinecone BL602 (SKU 19914)**
www.elektor.fr/19914

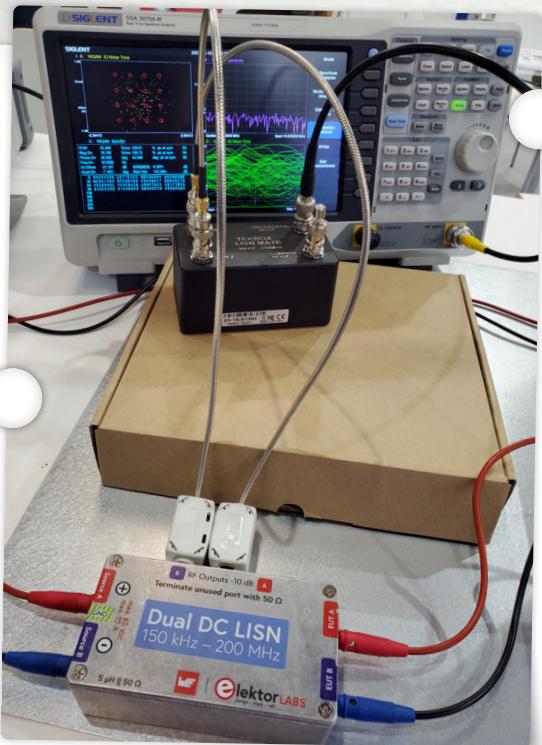
LIENS

- [1] Guido Nopper, « convertisseur A/N à PLD, simple à construire », Elektor 9-10/2019 : <https://www.elektormagazine.fr/magazine/elektor-111/51059>
- [2] Conception matérielle avec le RP2040 : <https://datasheets.raspberrypi.com/rp2040/hardware-design-with-rp2040.pdf>
- [3] Giovanni Carrera, « sortie analogique isolée pour Arduino Uno », Elektor 11-12/2022 : <https://www.elektormagazine.fr/magazine/elektor-283/61184>
- [4] Compensation de la soudure froide (Wikipédia) : https://en.wikipedia.org/wiki/Thermocouple#Reference_junction

embedded world 2023

Embedded world 2023 à Nuremberg, le rendez-vous incontournable pour tous ceux qui s'occupent de microcontrôleurs et d'outils de développement, a de nouveau eu lieu au printemps comme d'habitude. Cette année, il a accueilli 27 000 visiteurs, soit une augmentation de 50 % par rapport au salon 2022, avec plus de 950 exposants répartis dans 6 halls. Stuart Cording, rédacteur d'Elektor, C.J. Abate, directeur de contenu, et Jens Nickel, rédacteur en chef, ont eu la chance de faire le tour du salon et de trouver beaucoup de choses intéressantes, mais évidemment leur sélection personnelle ne peut représenter qu'une infime partie de tout ce qui se trouve sur le marché.

Vous en trouverez encore plus sur notre chaîne YouTube « Elektor TV : Industry » à l'adresse suivante : <https://youtube.com/@ElektorIndustry>.



Siglent

Le LISN/RSIL CC double d'Elektor a été mis à contribution sur le stand de Siglent. Le LISN CC double de 5 µH, 50-Ω, supporte des tensions jusqu'à 60 V et des courants jusqu'à 10 A. Il mesure les interférences RF sur les deux canaux au moyen de selfs de blocage de 5 µH. Le réseau d'atténuation interne de 10 dB (un dans chaque canal) contient un filtre passe-haut de troisième ordre avec une fréquence de coupure de 9 kHz, afin de protéger l'entrée d'instruments comme un analyseur de spectre, contre les tensions continues ou les basses fréquences potentiellement nuisibles provenant de l'équipement testé.

Regardez la démo sur : www.elektormagazine.com/ew-lisn



oxid[OS]

Le langage de programmation C est le pilier de l'industrie automobile depuis des décennies. Mais pour combien de temps encore ? Nous avons parlé avec Flavia Oprea pour savoir pourquoi Rust, un langage en plein essor, est considéré comme un moyen d'éviter les erreurs traditionnelles inhérentes à la programmation C embarquée. Basée en Roumanie, la startup a développé un système d'exploitation en temps réel (RTOS) basé sur Rust, qui a été présenté sur un microcontrôleur STM32 lors du salon. Les développeurs automobiles s'y intéressent déjà en raison de la promesse d'une sécurité accrue et de la prise en charge du code C/C++ isolé.

<https://oxidos.io>

Voir l'interview :

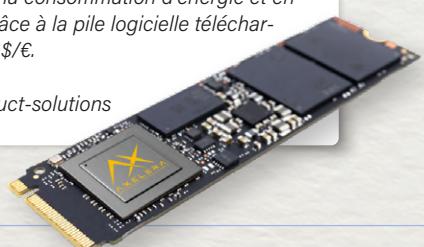
https://youtu.be/BNdvgJdN5_Q?t=1398

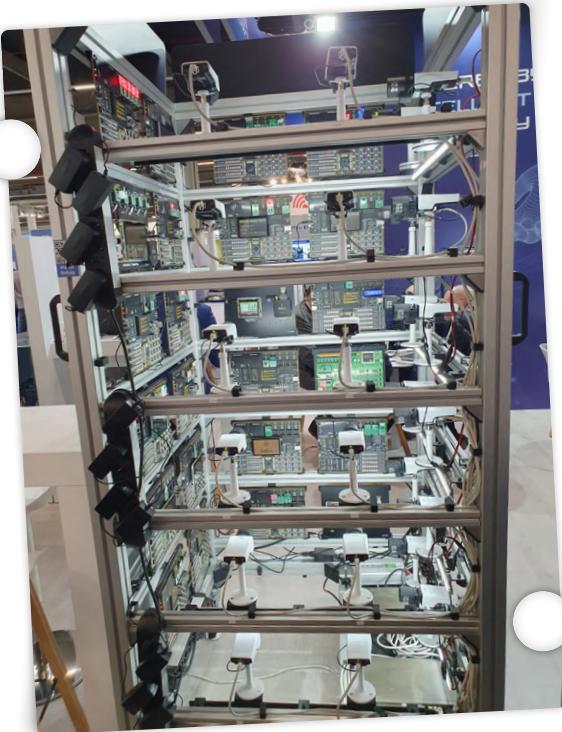


Axelera

La société néerlandaise Axelera propose des modules en périphérie, capables de traiter des applications d'inférence dans l'IA telles que la reconnaissance d'images. Les cartes sont disponibles dans différents facteurs de forme comme M.2 (image) ou PCIe. Alimenté par une seule AIPU Metis, le module M.2 fournit jusqu'à 214 téraflops par seconde (TOP) tout en minimisant la consommation d'énergie et en simplifiant l'intégration grâce à la pile logicielle téléchargeable. Le prix est de 149 \$/€.

<https://axelera.ai/ai-product-solutions>





Mikroelektronika

Sur le salon, Mikroe a présenté un cadre équipé d'une caméra pour son système Planet Debug : Les développeurs et les étudiants peuvent développer et déboguer ensemble un projet de microcontrôleur à distance, sans avoir besoin de disposer des mêmes cartes électroniques sur leur bureau. Les utilisateurs peuvent désormais configurer leur matériel selon leurs propres besoins spécifiques, en sélectionnant des cartes mères, des cartes processeurs et un grand nombre de périphériques et d'écrans.

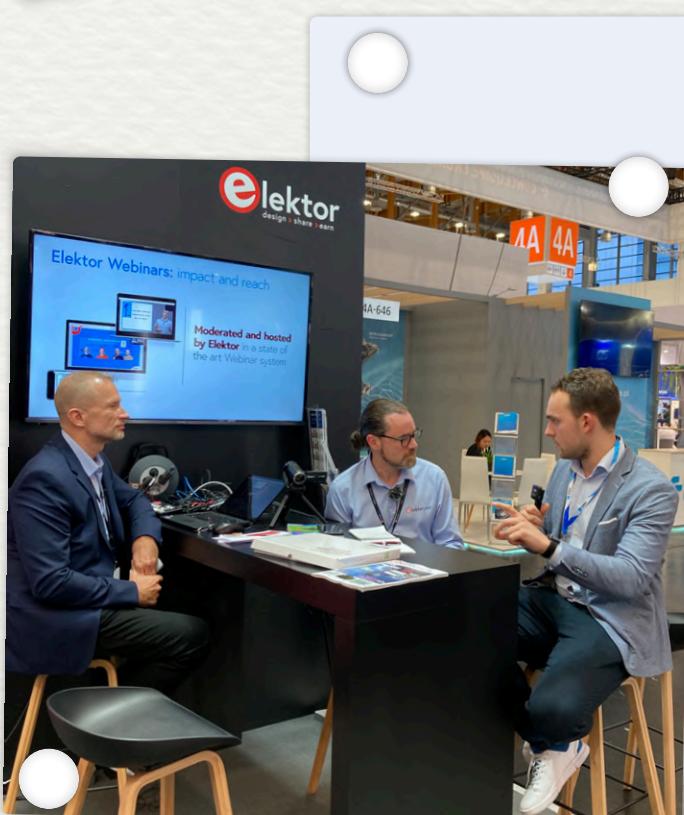
<https://mikroe.com/planet-debug>



Espressif

L'ESP32 est bien connu pour sa facilité d'utilisation lors de la mise en œuvre d'applications sans fil. C'est pourquoi l'ESP32 et d'autres solutions Espressif ont été présentées dans des dizaines de démonstrations intéressantes lors du salon Embedded World 2023. Parmi les cas mentionnés : un module ESP-ZeroCode pour les stores intelligents, des démonstrations de l'ESP RainMaker et des dispositifs Matter Wifi. Elektor a également rencontré Amey Inamdar d'Espressif à Embedded World pour découvrir les nouveautés. Parmi les points forts : les SoC RISC-V avec une version double cœur en préparation, la prise en charge de Matter pour faciliter le déploiement d'appareils IoT utilisant des protocoles différents, et la plateforme RainMaker pour la mise en œuvre du cloud backend de l'IoT.

Regardez l'interview de Stuart Cording !
<https://youtu.be/EFnUtAJX2aA>

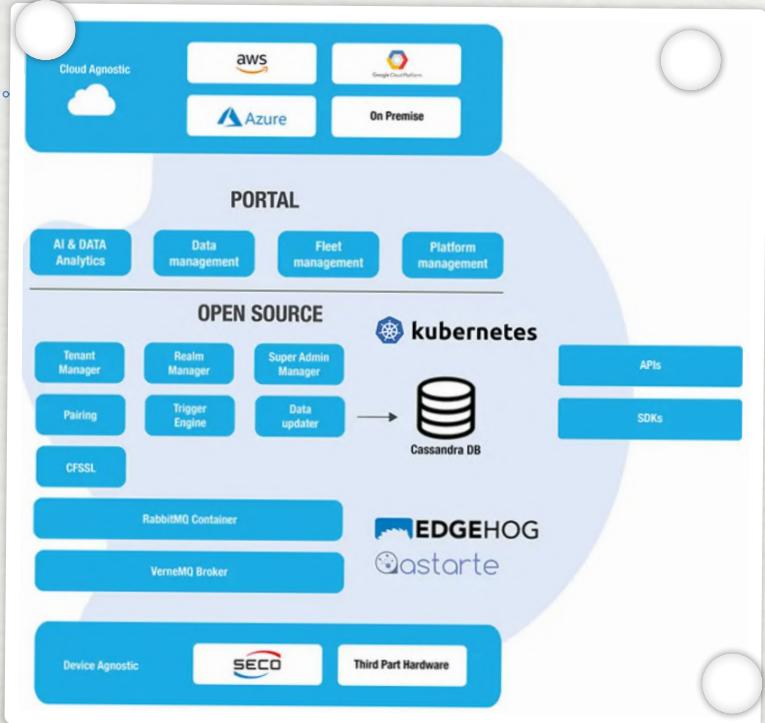


Mélange d'émissions en direct

Pour la deuxième fois, Stuart Cording d'Elektor Engineering Insights et Jens Nickel d'Elektor Lab Talk se sont réunis pour faire un reportage en direct d'un salon de l'électronique. Dans cette émission d'une heure, Stuart et Jens ont discuté avec le directeur du contenu d'Elektor, C.J. Abate, des choses les plus intéressantes vues dans Embedded World 2023 (par exemple, un robot servant de la bière et contrôlé par le cloud). Invité spécial : Viacheslav Gromov, auteur d'Elektor, qui a remporté le «embedded award» dans la catégorie «AI» avec sa société AITAD. ▶

Elektor TV
 Regardez la vidéo en entier ici :
www.youtube.com/watch?v=00d5lVKZx7Q





Seco

Seco, fournisseur italien d'IPC et de solutions, a présenté la plateforme IoT modulaire CLEA. Au plus près, elle peut surveiller les données d'événements du matériel lui-même et des applications du client. À cette fin, CLEA propose des API ouvertes et des convertisseurs de protocole, ainsi qu'une logique de périphérie basée sur l'IA pour l'analyse des données brutes et des images. CLEA est indépendante en termes de matériel et d'informatique en cloud, avec une base open-source, ce qui garantit une disponibilité durable. Parmi les applications déjà réalisées figurent des distributeurs de café intelligents, des stations de recharge et des appareils médicaux.

<https://north.seco.com/en/products-and-services/clea-iot-platform>



Arm

Comme nous l'avons expliqué dans notre émission Elektor Engineering Insights, le monde de l'automobile est en train de changer radicalement. Aujourd'hui, une fois que le logiciel d'un calculateur est finalisé, il n'est jamais mis à jour, à moins qu'un problème de sécurité ne soit détecté. Mais cette situation est en train de changer, comme l'explique Robert Day avec la nouvelle initiative SOAFEE d'Arm. S'inspirant de la manière dont les orchestrateurs et les conteneurs sont utilisés dans les logiciels cloud et les processus de développement CI/CD, ces méthodologies sont adaptées aux logiciels automobiles critiques pour la sécurité. Actuellement, une série de fournisseurs de semi-conducteurs, d'équipementiers automobiles et leurs fournisseurs sont impliqués.

Regardez la vidéo de l'interview !
https://youtu.be/nr_6W4UgihM



Slint

Slint est une boîte à outils permettant de développer efficacement des interfaces graphiques pour les écrans de bureau et embarqués. Plusieurs langages de programmation sont pris en charge, tels que Rust, C++ et JavaScript. Selon les développeurs, la solution ne nécessite que quelques centaines de ko de RAM et une faible puissance de traitement. Il existe des licences commerciales, mais aussi en open-source sous GPLv3.

<https://slint-ui.com/#tryout>



Digi

Le module Digi XBee® XR 868 est une solution compacte et fiable qui prend en charge l'utilisation d'applications de connectivité à longue portée dans les régions de l'Union européenne. Le module pré-certifié fonctionne entre 863 et 870 MHz et prend en charge les protocoles de réseau point à point et maillé avec une portée en visibilité directe de plus de 14 kilomètres. Bien entendu, un kit de développement est également disponible (image).

<https://digi.com/products/embedded-systems/digi-xbee/rf-modules/sub-1-ghz-rf-modules/digi-xbee-xr-868>

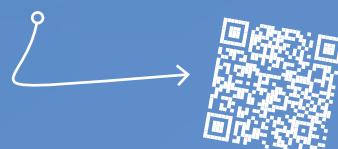
Elektor TV Shows



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.

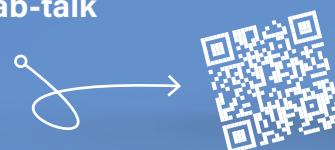
www.elektormagazine.com/elektor-engineering-insights



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



l'échantillonnage sub-Nyquist en pratique

capture fiable des hautes fréquences à l'aide du sous-échantillonnage

Sebastian Westerhold (AI5GW) (Allemagne)

Quiconque s'occupe de systèmes d'échantillonnage connaît rapidement les limites imposées par le théorème d'échantillonnage de Nyquist-Shannon. Selon cette règle, les signaux dont la fréquence est supérieure à la moitié de la fréquence d'échantillonnage ne peuvent plus être détectés de manière fiable. Mais ce n'est que la moitié de la vérité. Le sous-échantillonnage, également appelé échantillonnage sub-Nyquist, est une méthode fiable pour capturer les signaux de fréquence plus élevée. Cet article présente cette méthode de manière pratique, en utilisant un simple Arduino Uno.

Le problème

Le théorème d'échantillonnage de Nyquist-Shannon est très important lorsqu'il s'agit de systèmes d'échantillonnage. Sa version très simplifiée stipule que la fréquence d'échantillonnage doit être au moins le double de la fréquence la plus élevée apparaissant dans le signal à échantillonner. Vous avez certainement déjà entendu ou lu quelque chose de ce genre. Que se passe-t-il si vous échantillonnez un signal sinusoïdal pur de 120 kHz avec une fréquence d'échantillonnage de seulement 100 kS/s (kHz) ? Il en résultera ce que l'on appelle un repliement à 20 kHz. Les échantillons acquis ressembleraient donc à un signal sinusoïdal d'une fréquence de 20 kHz (120 kHz - 100 kS/s) qui aurait été échantillonné. Et les erreurs de repliement doivent être évitées par tous les moyens. Vraiment ? La question de savoir si les erreurs de repliement constituent un facteur d'interférence ou peuvent être utilisées comme une technique utile dépend de l'application spécifique. Dans l'exemple de la fréquence d'échantillonnage de 100 kS/s, il serait impossible de distinguer un signal de 20 kHz d'un signal de 120 kHz ; les deux signaux ressembleraient à un signal de 20 kHz. Incidemment, un signal de 80 kHz produirait également un repliement de 20 kHz (100 kS/s - 80 kHz) (**figure 1**). Toutefois, cette circonstance n'est critique que si une distinction entre les fréquences du signal mentionnées est nécessaire. Si toutes les composantes de fréquence effectivement présentes dans le signal se situent dans une plage clairement identifiable, l'échantillonnage peut être significatif si les effets de repliement sont intentionnellement exploités. L'unicité est assurée si toutes les composantes de fréquence apparaissant dans le signal à échantillonner se situent dans la même zone de Nyquist.

La plage de fréquences allant de 0 Hz (CC) à la moitié de la fréquence d'échantillonnage

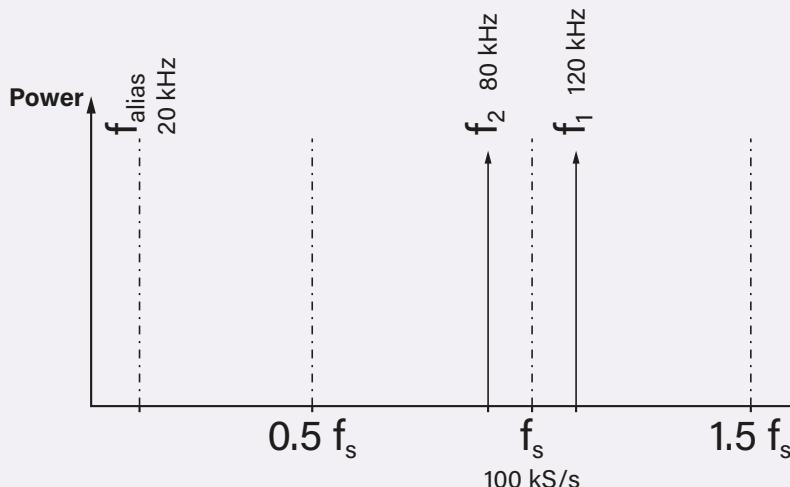


Figure 1. À une fréquence d'échantillonnage de 100 kS/s, un signal de 120 kHz (f_1) et un signal de 80 kHz (f_2) produisent un effet de repliement (f_{ALIAS}) de 20 kHz.

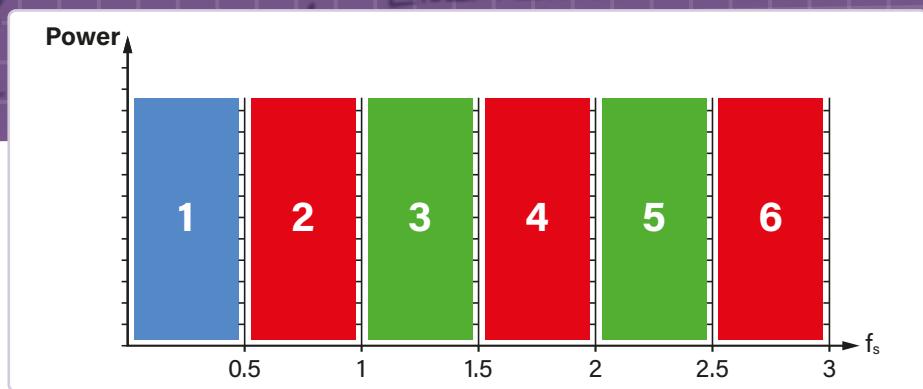


Figure 2. Zones de Nyquist 1 to 6. zone 1 = plage normale, zones 2, 4, 6 = plages d'inversion, zones 3 and 5 = zones non inversées.

(f_s) est appelée zone 1 de Nyquist (**figure 2**). Cette zone est la plage de fonctionnement « normale » d'un système d'échantillonnage. La zone 2 de Nyquist s'étend de $0,5 f_s$ à f_s . Ces zones se poursuivent ensuite théoriquement à l'infini à intervalles de $0,5 f_s$. Les zones de Nyquist avec des nombres pairs ont une caractéristique particulière : à l'intérieur de ces zones, une inversion du spectre a lieu. Par conséquent, si la largeur de bande d'un signal à capturer est limitée à la plage de l'une de ces zones de Nyquist, le sous-échantillonnage peut être utilisé pour capturer des signaux dont les composantes de fréquence dépassent de manière significative la fréquence d'échantillonnage.

Un exemple pratique

Dans le cadre d'un projet, la fréquence d'un signal dans la plage d'environ 160 kHz à 165 kHz doit être acquise de manière fiable avec un Arduino UNO [2]. Pour des raisons techniques, le taux d'échantillonnage a été fixé à environ 154 kS/s en utilisant les registres appropriés (*prescaler* 1:8) (à proprement parler, il s'agit de 153,846 S/s ; par souci de clarté, le chiffre arrondi est utilisé dans

cet article). Ainsi, la fréquence du signal est supérieure de quelques kHz à la fréquence d'échantillonnage elle-même, mais inférieure à $1,5 f_s$ (correspondant à la zone 3 de Nyquist). En conséquence, le repliement à attendre du fait du sous-échantillonnage se situe entre 6 kHz (160 kHz - f_s) et 11 kHz (165 kHz - f_s). Le code Arduino est structuré de telle sorte que le convertisseur A/N de l'Arduino utilise une routine de service d'interruption (ISR) pour échantillonner le signal à l'entrée analogique 0 jusqu'à ce qu'un tampon de 512 échantillons soit rempli. La fréquence du signal (de repliement) est alors calculée à partir de ces valeurs avant que les 512 échantillons suivants soient échantillonnés et que le jeu recommence. Pour déterminer la fréquence, l'algorithme de Goertzel a été utilisé, une forme spéciale de la transformée de Fourier discrète qui permet d'économiser des ressources [1]. Le résultat est ensuite envoyé à des fins de démonstration sur le port série, visuellement formaté pour le *Serial Monitor* d'Arduino (**figure 4**). En utilisant le même principe, on pourrait, par exemple, capturer une FI de 455 kHz à partir d'un récepteur radio dans la zone 6 de Nyquist ($3 f_s$, équivalent à environ 462 kHz) en utili-

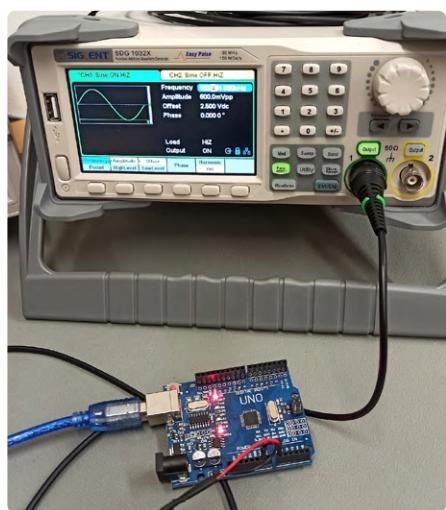


Figure 3. Système de test avec Arduino et un générateur de signaux.

```

159555.12kHz: #
159555.05kHz: #
160156.09kHz: #
160456.56kHz: ##
160757.04kHz: ###
161057.53kHz: ##
161358.01kHz: ##
161658.50kHz: ####
161958.96kHz: ####
162259.45kHz: #####
162559.39kHz: ######
162860.40kHz: #######
163160.89kHz: ##########
163461.37kHz: #####
163761.85kHz: #####
164062.34kHz: #####
164362.81kHz: #####
164663.29kHz: #####
164963.18kHz: ####
165264.25kHz: ####
165564.73kHz: ####
165865.21kHz: ####
166165.70kHz: ####
166466.18kHz: ####
166766.65kHz: ####
167067.14kHz: ####
167367.62kHz: ####
167668.09kHz: ####
167968.57kHz: ####
168269.06kHz: ####
168569.54kHz: ####

```

Figure 4. Les résultats sont affichés dans le *Serial Monitor*.

sant le sous-échantillonnage et démoduler les signaux FSK (NAVTEXT, RTTY, etc.) même avec un Arduino UNO à vitesse relativement faible.

Remarques

Pour que le sous-échantillonnage fonctionne comme souhaité, le CAN utilisé doit avoir la bande passante analogique requise. Celle-ci est généralement beaucoup plus élevée que la fréquence d'échantillonnage. Dans la fiche technique, vous trouverez généralement ce chiffre sous la forme d'une « bande passante à pleine puissance ». Avec l'ATmega328P, le sous-échantillonnage semble fonctionner très proprement jusqu'à la gamme des MHz, d'après mes propres expériences. Malheureusement, mes expériences avec le Raspberry Pi Pico n'ont pas donné de résultats prometteurs. ─

VF : Maxime Valens – 220629-04

Des questions, des commentaires ?

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



Produits

- **Siglent SDG1032X générateur de signaux à 2 voies (30 MHz)**
<https://www.elektor.fr/20276>
- **OWON XSA810 analyseur de spectre (1 GHz)**
<https://www.elektor.fr/19714>
- **Radio Logicielle HackRF One Great Scott Gadgets (1 MHz à 6 GHz)**
<https://www.elektor.fr/18306>
- **MonoDAQ-U-X Système USB polyvalent d'acquisition de données (50 kéch/s)**
<https://www.elektor.fr/18766>

LIENS

- [1] Sebastian Westerhold (2022) : Bibliothèque Goertzel pour Arduino : <https://github.com/AI5GW/Goertzel>
- [2] Croquis Arduino de cet article : <https://www.elektormagazine.fr/220629-04>

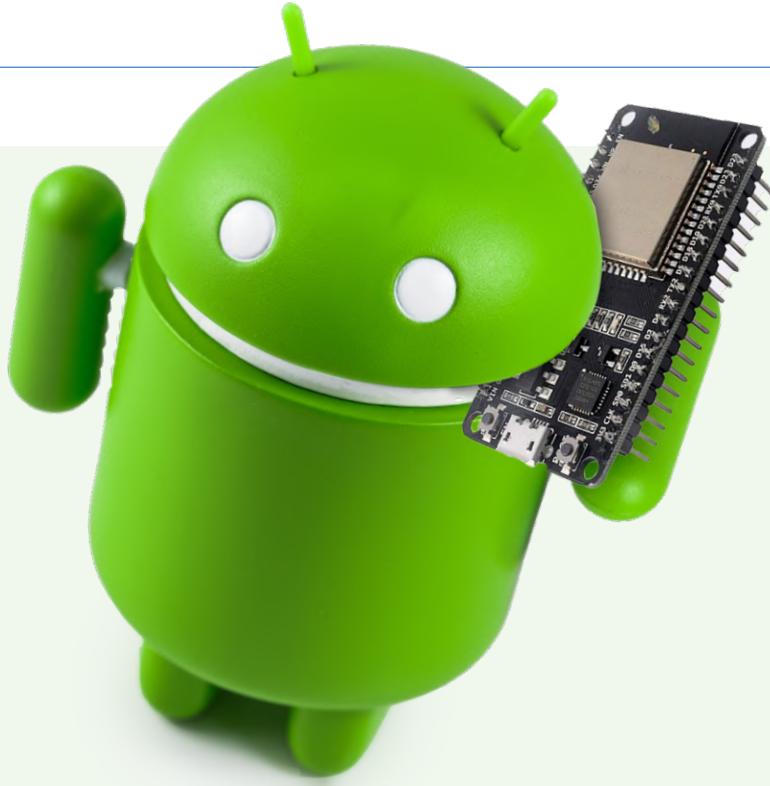
connecter un smartphone Android à un ESP32 ?

projet pratique avec l'API wifi d'Android

Tam Hanna (Hongrie)

Les développeurs d'applications Android savent que le code nécessaire pour se connecter à un réseau wifi n'est pas simple, surtout lorsqu'il s'agit de rendre l'ensemble du processus aussi convivial que possible. Les développeurs de logiciels doivent également s'assurer que leur code est compatible avec les différentes versions d'Android et prendre en compte les mesures de sécurité exigées par Google pour que l'application fonctionne sans problème. Cet article devrait vous aider à vous frayer un chemin dans la jungle.

Si vous n'êtes pas préoccupé par sa consommation d'énergie relativement élevée, le wifi est la norme radio de choix, en particulier pour les petits projets. On peut exclure le Bluetooth à cause des coûts de certification du Bluetooth SIG et des frais de licence. La communication peut se faire via une architecture d'interface REST et les émetteurs wifi sont disponibles un peu partout. Il n'est guère surprenant que les petites cartes de développement ESP32 – avec leurs points d'accès wifi intégrés – soient si populaires auprès des développeurs d'applications Android.



Le système d'exploitation Android est conçu pour privilégier l'expérience de l'interface utilisateur plutôt que le temps de réponse aux stimuli externes. Cette latence élevée le rend inadapté à la gestion d'événements critiques. Android est doté de piles d'interfaces graphiques importantes et, pour les tâches complexes telles que l'affichage de graphiques, il existe des bibliothèques que vous pouvez charger avec Gradle.

Si un système nécessite une réponse rapide à des événements en temps réel et a également des exigences en matière d'interface utilisateur, il semble logique de répartir le traitement comme le montre la **figure 1**.

Souvent, le maillon faible de la chaîne est l'utilisateur final, qui doit entrer des valeurs dans le système. Si l'utilisateur final doit établir manuellement une connexion à un certain réseau wifi, votre équipe d'assistance produit ne manquera pas de recevoir d'innombrables appels de clients frustrés.

La situation serait plus confortable s'il était possible d'établir une connexion automatique à un réseau. Cependant, Google a des préoccupations en matière de sécurité fonctionnelle et de protection des données personnelles concernant une telle technique.

De quoi s'agit-il ?

L'objectif de cet article est de décrire le code Android nécessaire à l'établissement d'un lien de communication avec un réseau wifi cible. Un ESP32 utilise SoftAP pour établir un point d'accès wifi à distance, auquel l'appareil Android peut se connecter. En pratique, il est possible d'utiliser ce code avec de nombreux autres contrôleurs compatibles wifi (voir le téléchargement [1]).

Cet article est basé sur un projet que l'auteur a développé pour un client. Pratiquement toutes les cartes ESP32 qui exécutent une version du programme d'exemple, *esp-idf/examples/wifi/getting_started/* (inclus dans l'ESP-IDF [2]) configurées comme point d'accès conviendraient



Figure 1. Le smartphone Android gère l'interface graphique, tandis que l'ESP32 est responsable de la communication matérielle.

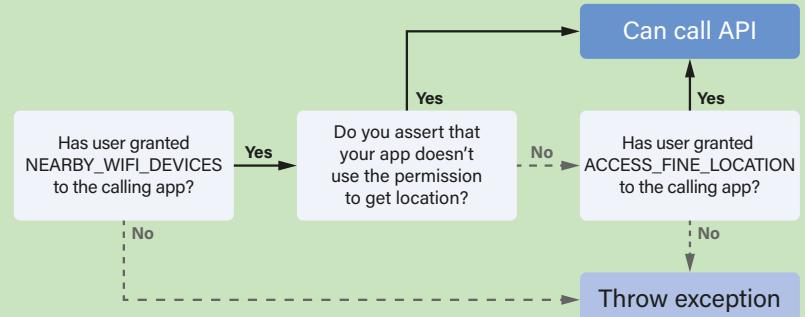


Figure 2. Des autorisations supplémentaires peuvent être requises pour la communication sans fil sous Android 13 (voir l'image [4]).

également. Nous supposons que vous avez une expérience préalable de l'environnement de développement ESP32. Côté Android, vous devez disposer de deux smartphones : l'un fonctionnant sous Android build ≥ 10 , et l'autre sous Android 9 ou antérieur. Android Studio [3] sert d'environnement de développement. Nous supposons également que le lecteur a une certaine expérience préalable de l'EDI Android.

Configuration

La compilation/configuration d'un projet Android est un peu compliquée. En général, vous trouverez une version du fichier `build.gradle` appartenant au module `app` et permettant de configurer tous les paramètres et dépendances nécessaires à l'application dans Android Studio. Ce fichier contient généralement des déclarations telles que :

```
android {
    compileSdkVersion 29
    buildToolsVersion "30.0.1"
    defaultConfig {
        minSdkVersion 26
        targetSdkVersion 31
    }
}
```

Le champ `minSdkVersion` permet au développeur de spécifier la version minimale du système d'exploitation Android pour l'appareil cible ou final avec lequel l'application est compatible. La valeur 26 correspond à Android 8 (Oreo). `compileSdkVersion` indique la version du kit de développement logiciel (SDK) avec lequel l'application sera compilée, et `buildToolsVersion` indique la version des outils de construction Android qui sera utilisée pour compiler le projet. La ligne `targetSdkVersion` indique la version du système d'exploitation Android pour lequel une application est développée.

Cette distinction, qui semble compliquée à première vue, est nécessaire car le « comportement » de nombreuses API Android change en fonction de la version du système d'exploitation sous laquelle l'application est exécutée.

Si `targetSdkVersion` est défini à une certaine valeur, le système d'exploitation suppose que le développeur a pris en compte tous les changements introduits dans cette version lors de la conception de l'application - si la valeur est inférieure, le mode de compatibilité est activé. Malheureusement, Google a des directives strictes concernant les valeurs spécifiées dans `targetSdkVersion`. Si une application est conçue pour fonctionner uniquement sur des versions plus anciennes

du système d'exploitation, Android Studio vous alertera pendant la compilation en affichant le message « *Google Play requires that apps target API level 30 or higher* », indiquant que le backend du Play Store refusera de télécharger l'APK généré. Dans le futur proche, Google a l'intention de supprimer progressivement les « très vieilles » applications du Play Store.

En tant que développeur, vous devez être conscient de ces limites et difficultés éventuelles.

Autorisations

Le monde d'Android se marque par deux phases : avant et après l'introduction du nouveau système de gestion des autorisations et/ou du *Storage Access Framework*. Auparavant, les développeurs étaient libres de faire ce qu'ils voulaient, mais Google a ensuite pris au sérieux la question de la sécurité des données personnelles.

Pour le wifi, la situation est délicate, car les informations de localisation peuvent être obtenues à partir de la découverte des réseaux sans fil dans la zone (par exemple, le premier iPod touch d'Apple, dépourvu de GPS ou de capacité cellulaire, pouvait déterminer sa propre localisation ainsi).

Il est maintenant nécessaire de spécifier les déclarations suivantes dans le fichier `manifest` de l'application Android pour demander certaines autorisations.

```
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<uses-permission android:name="android.permission.CHANGE_WIFI_STATE" />
<uses-permission android:name="android.permission.CHANGE_NETWORK_STATE" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
```

Si vous souhaitez que votre application prenne en charge Android 13, vous devez apporter une modification supplémentaire : dans le cas du WLAN, la logique des autorisations d'Android 13 est décrite dans la **figure 2**.

Il est maintenant nécessaire de spécifier les déclarations suivantes dans

le fichier *manifest* de l'application Android pour demander certaines autorisations.

```
<uses-permission android:name="android.permission.NEARBY_WIFI_DEVICES" android:usesPermissionFlags="neverForLocation" />
```

L'autorisation `android.permission.NEARBY_WIFI_DEVICES` est une demande faite par le développeur dans le but de permettre à l'application de rechercher et d'accéder à tous les appareils wifi à proximité, tandis que `android:usesPermissionFlags="neverForLocation"` est un mode spécial qui attribue un *flag* “*never for location*” à accorder à l'application. Cela signifie que l'application n'utilise les informations wifi qu'à des fins de configuration et non pour obtenir des informations de localisation. Dans ce cas, l'autorisation `android.permission.ACCESS_FINE_LOCATION` n'est pas requise sous Android 13.

Les versions antérieures du système d'exploitation requièrent cette autorisation, et dans ce cas, une configuration similaire à celle décrite ci-dessous est recommandée :

```
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" android:maxSdkVersion="32" />
```

Dans ce contexte, l'attribut `maxSdkVersion` garantit que la déclaration d'autorisation n'est reconnue que par les versions antérieures d'Android.

Développement à deux volets

Pour configurer l'activité chargée de « rechercher », dans l'environnement, un point d'extrémité wifi accessible, nous devons d'abord vérifier si le wifi du téléphone est activé :

```
@Override  
protected void onCreate(Bundle savedInstanceState) {  
...  
if (android.os.Build.VERSION.SDK_INT <  
    Build.VERSION_CODES.Q) {  
    wifiManager =  
        (WifiManager) getApplicationContext()  
            .getSystemService(Context.WIFI_SERVICE);  
    if (!wifiManager.isWifiEnabled()) {  
        wifiManager.setWifiEnabled(true);  
    }  
}
```

Si nous utilisons une ancienne version d'Android qui n'est pas affectée par les modifications de l'API, notre programme peut activer le wifi sur le téléphone sans avoir à demander le consentement de l'utilisateur. Malheureusement, cela ne s'applique pas aux versions plus récentes d'Android - le consentement de l'utilisateur est alors requis.

Pour distinguer les deux cas lors de l'exécution, nous utilisons la constante `android.os.Build.VERSION.SDK_INT`. Elle fait référence à la version de l'API prise en charge par le micrologiciel du téléphone utilisé.

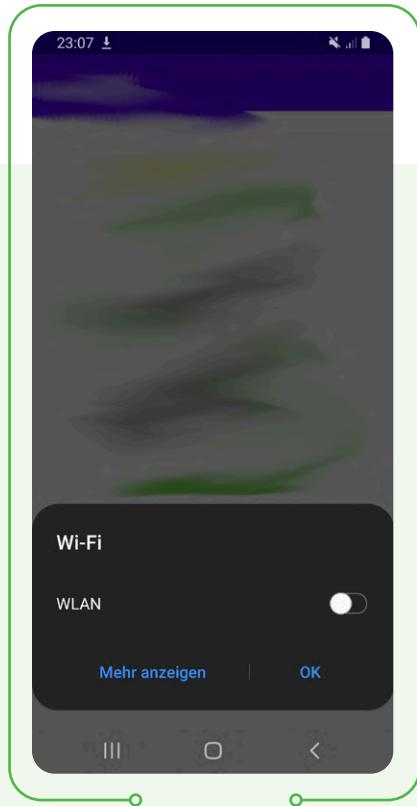


Figure 3. Google exige le consentement explicite de l'utilisateur.

Pour activer le wifi, nous devons envoyer un *Intent* de type `Settings.Panel.ACTION_WIFI` :

```
else {  
    wifiManager =  
        (WifiManager) getApplicationContext()  
            .getSystemService(Context.WIFI_SERVICE);  
    if (!wifiManager.isWifiEnabled()) {  
        Toast toast =  
            Toast.makeText(getApplicationContext(),  
                "PLEASE enable WiFi so that we can connect!",  
                Toast.LENGTH_LONG);  
        toast.show();  
        Intent panelIntent =  
            new Intent(Settings.Panel.ACTION_WIFI);  
        this.startActivityForResult(panelIntent,  
            IGY_WIFI_REQCODE);  
    }  
}
```

Le résultat de tous ces efforts est le message de la **figure 3** qui apparaît au moment de l'exécution. L'utilisateur doit basculer le bouton pour autoriser le processus d'activation.

Ensuite *Intent* est envoyé à l'utilisateur, l'informant de la nécessité d'activer le wifi.

Dans le cas contraire, il est nécessaire de rappeler à l'utilisateur la nécessité d'activer le wifi.

Pour les débutants en Android, il peut sembler étrange que `IGY_WIFI_REQCODE` soit transmis à `startActivityForResult()`, mais il s'agit d'un code de corrélation qui permet au point de réception final d'identifier le facteur de déclenchement à l'origine de la demande entrante.



Vieux mais précieux

Une fois que vous aurez terminé votre première application pour smartphone, pourquoi ne pas revenir en arrière et découvrir les principes fondamentaux sur lesquels Jeff Hawkins s'est appuyé pour créer Palm OS, qui fonctionnait sur les appareils Palm et Palm Pilot. Même si ces appareils sont aujourd'hui obsolètes, la philosophie qui a inspiré la conception de Palm OS est intéressante. Zen of Palm est assez court, et vous pouvez en trouver une version PDF sur [5].

Les codes de corrélation sont des entiers normaux auxquels le système d'exploitation n'attribue aucune valeur significative. L'auteur les déclare dans le programme avec le mécanisme suivant ; il est seulement important que le code généré soit unique.

```
public class MainActivity extends
AppCompatActivity implements
AdapterView.OnItemClickListener {
    private final int MY_PERMISSIONS_
ACCESS_COARSE_LOCATION = 1;
    int ICY_WIFI_REQCODE = 4242;
```

La tâche suivante consiste à déclencher le processus de recherche de wifi. Cet article est limité ; ceci nous empêche de nous plonger dans des détails sur la pile de l'interface graphique d'Android, mais nous sommes convaincus que vous trouverez un moyen d'activer la méthode `getWifi()`.

La première étape du traitement des informations de localisation consiste à vérifier si nous disposons déjà d'une autorisation d'accès pour le `Manifest.permission.ACCESS_FINE_LOCATION`. Sous Android 13, vous devez également inclure la nouvelle permission dans le code d'analyse :

```
private void getWifi() {
    if (ContextCompat.
checkSelfPermission(MainActivity.this,
Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
        ActivityCompat.
requestPermissions(MainActivity.this,
new String[]{Manifest.permission.
ACCESS_FINE_LOCATION},
MY_PERMISSIONS_ACCESS_COARSE_LOCATION);
    }
}
```

Rappelons que, sous Android 6.0 et plus, les autorisations « sensibles » ne sont activées par le système d'exploitation que lorsque l'utilisateur les accepte explicitement. Des dialogues sont utilisés à cet effet, comme le montre la **figure 4**.

Si nous avons déjà donné la permission, l'étape suivante est de changer encore de version. Si nous travayons avec une version antérieure d'Android, nous appelons directement la méthode `getWifiWorkerOld()`

```
:  
else {
    if (android.os.Build.VERSION.SDK_INT <
Build.VERSION_CODES.Q) {
        getWifiWorkerOld();
    }
}
```

Dans le cas des nouveaux téléphones, et pour éviter un plantage, l'état d'alimentation de l'émetteur wifi est à nouveau vérifié à ce stade, car il est tout à fait possible que les utilisateurs éteignent le wifi avant que la méthode ne devienne active.

```
else {
    if (wifiManager.isWifiEnabled()) {
        getWifiWorkerOld();
    }
    else {
        Toast toast =
        Toast.makeText(getApplicationContext(),
        "PLEASE enable WiFi so that we can connect!",
        Toast.LENGTH_LONG);
        toast.show();
        Intent panelIntent =
        new Intent(Settings.Panel.ACTION_WIFI);
        this.startActivityForResult(panelIntent,
        ICY_WIFI_REQCODE);
    }
}
```

Une fois que nous avons vérifié l'état de l'alimentation de l'émetteur, nous activons le processus de recherche avec de la procédure suivante :

```
private void getWifiWorkerOld() {
    ...
    wifiManager.startScan();
}
```

`wifiManager` est une classe système que nous avons obtenue dans le cadre de l'activation de l'activité. La méthode `startScan()` se charge ensuite de lancer le processus d'analyse.

Le renvoi des informations collectées par le balayage s'effectue par *Broadcast*. Pour les recevoir, nous avons besoin d'un récepteur d'un *Broadcast Receiver*. Comme nous ne voulons pas le créer dans le

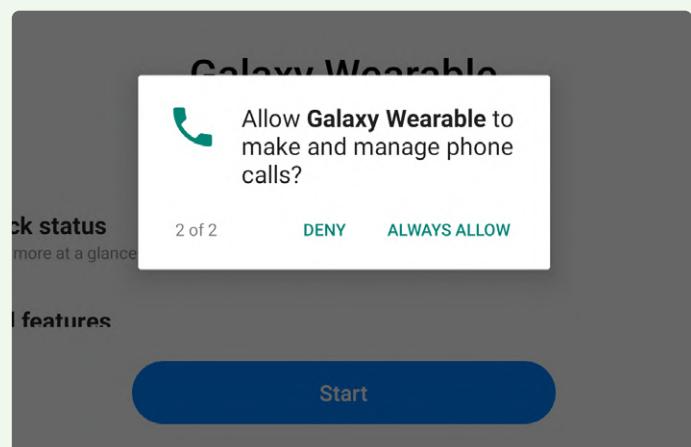


Figure 4. Tenez compte du nouveau système d'autorisations.

fichier *manifest*, nous l'écrivons dans la méthode `onPostResume()` de l'activité :

```
@Override  
protected void onPostResume() {  
    super.onPostResume();  
    receiverWifi =  
    new WifiReceiver(wifiManager, wifiList, this);  
    IntentFilter intentFilter =  
    new IntentFilter();  
    intentFilter.addAction  
(WifiManager.SCAN_RESULTS_AVAILABLE_ACTION);  
    registerReceiver(receiverWifi, intentFilter);  
}
```

Afin d'assurer la conformité avec le Play Store, il est important que les *Broadcast Receivers* enregistrés manuellement soient également désenregistrés par le système d'exploitation. On peut le faire de la manière la plus pratique avec l'une des méthodes du cycle de vie de l'activité. J'ai décidé d'utiliser ici la méthode `onPause()` :

```
@Override  
protected void onPause() {  
    super.onPause();  
    unregisterReceiver(receiverWifi);  
}
```

Broadcast Receiver pour le traitement de données

Étant donné que la réception des informations réseau renvoyées par le processus de balayage wifi se fait via le système Broadcast d'Android, nous devons intégrer la classe `BroadcastReceiver` enregistrée ci-dessus avec le système d'exploitation.

Les *BroadcastReceivers* dans Android prennent la forme de classes dérivées de la classe principale `BroadcastReceiver`. Dans notre exemple, sans tenir compte des variables de l'interface graphique, l'en-tête se présente comme suit (nous n'incluons pas le constructeur nécessaire pour remplir les champs) :

```
class WifiReceiver extends BroadcastReceiver {  
    WifiManager wifiManager;  
    ListView wifiDeviceList;  
    MainActivity myParentAct;
```

Maintenant, la question importante est de savoir comment traiter les informations réseau renvoyées par les émetteurs wifi. La réponse se trouve dans la méthode `onReceive()`, que le système d'exploitation activera à l'arrivée des informations diffusées sous forme d'intentions (*Intents*).

```
public void onReceive(Context context, Intent intent) {  
    String action = intent.getAction();  
    if (WifiManager.  
        SCAN_RESULTS_AVAILABLE_ACTION.equals(action)) {
```

```
    List<ScanResult> wifiList =  
        wifiManager.getScanResults();  
    ArrayList<String> deviceList =  
        new ArrayList<>();  
    for (ScanResult scanResult : wifiList) {  
        if(scanResult.SSID.  
            contains("NAMEOFTHING"))  
            deviceList.add(scanResult.SSID );  
    }  
    myParentAct.myArrayAdapter=  
        new ArrayAdapter(context,  
            android.R.layout.simple_list_item_1,  
            deviceList.toArray());  
    wifiDeviceList.  
        setAdapter(myParentAct.myArrayAdapter);  
}
```

Dans la première étape, le code appelle `intent.getAction()` pour vérifier le nom de l'*Intent*. Le Broadcast Receiver peut théoriquement contenir n'importe quelle intention ; en vérifiant la chaîne `WifiManager.SCAN_RESULTS_AVAILABLE_ACTION`, nous nous assurons que nous traitons avec une intention « compatible ».

Puisque l'exemple de code dont l'auteur a tiré les extraits suivants contient une liste pour l'affichage, il est nécessaire de faire certaines modifications à l'étape suivante. Le but de cette démarche est de peupler la classe `deviceList`.

À ce stade, dans une application réelle, une liste serait remplie d'informations, mais vous pouvez procéder différemment, bien entendu, si vous ne considérez qu'un seul réseau wifi cible comme valide.

Un exemple de l'autre approche serait un appareil qui établit le contact avec son application de base lors de la configuration et qui l'indique par un nom wifi spécifique. Dans ce cas, nous pourrions immédiatement continuer après avoir détecté ce réseau.

Une configuration de connexion bien pensée

Une longue période d'attente dans les applications pour smartphones n'est pas du goût des utilisateurs. Il est donc logique d'afficher une barre de progression ou des symboles similaires qui assurent à l'utilisateur que l'application n'est pas plantée.

Dans l'application du client qui sert de base à cet article, elle a été implémentée sous la forme d'une activité dédiée : elle affiche le logo de l'entreprise et une barre de progression, ce qui rassure l'utilisateur. Elle est activée par l'envoi d'une intention :

```
public void onItemClick(AdapterView<?>  
    adapterView, View view, int pos, long anID) {  
    String y = (String) myArrayAdapter.getItem(pos);  
    Intent intent =  
        new Intent(MainActivity.this,ConnActivity.class);  
    intent.putExtra("WIFINAME", y);  
    startActivity(intent);  
}
```



Attention ! Google peut bloquer les demandes d'autorisation

À partir d'Android 11, Google a introduit les « autorisations uniques », qui permettent à l'utilisateur d'accorder à une application l'accès à une certaine fonctionnalité ou à des données une seule fois. Cela modifie également la façon dont les applications d'arrière-plan demandent des autorisations, ce qui peut entraîner le blocage de certaines demandes. L'objectif est de permettre aux utilisateurs de mieux contrôler leurs données et de renforcer la protection de la vie privée sur Android. Les développeurs doivent être conscients de ces problèmes éventuels.

Le code commence par déclarer un `AsyncTask` qui effectue un test de présence. Il s'agit d'une vérification (non décrite ici) que le pair de connexion fait partie de l'écosystème du client :

```
public class ConnActivity extends AppCompatActivity {  
    PresenceTestAsynctask aT;  
    ConnActivity mySelf;
```

Dans `onCreate()`, l'établissement de la connexion commence sans aucune intervention de la part de l'utilisateur. La première action officielle consiste à configurer certaines variables membres et à appeler les méthodes `getIntent().getExtras()` pour rendre les informations - « mises en paquet » plus haut par *Intent* - accessibles avec le nom wifi et d'autres données de support :

```
@Override  
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_conn);  
    mySelf=this;  
    Bundle b = getIntent().getExtras();  
    String networkPass = "sdsds";
```

L'étape suivante est un autre changement de version, qui vérifie l'« état de la version » du matériel cible. Si nous travaillons avec une version antérieure d'Android, l'étape suivante est le démarrage de la configuration de la connexion selon la procédure suivante :

```
if (android.os.Build.VERSION.SDK_INT <  
Build.VERSION_CODES.Q) {  
    WifiConfiguration conf =  
    new WifiConfiguration();  
    conf.SSID = "\"" +  
    b.getString("WIFINAME") + "\"";  
    conf.preSharedKey = "\"" + networkPass + "\"";  
    WifiManager wifiManager =  
    (WifiManager) getApplicationContext().  
    getSystemService(Context.WIFI_SERVICE);  
    wifiManager.addNetwork(conf);
```

La méthode `addNetwork()` prend en charge un objet `WifiConfiguration`. Sa tâche consiste à fournir un ensemble complet d'informations sur le réseau local sans fil, auquel le téléphone peut ensuite se connecter.

Dans cette dernière étape, nous devons établir la connexion selon le mécanisme suivant, avec trois méthodes : `disconnect()`,

`enableNetwork()` et `reconnect()`. Ensuite vient la phase d'activation de l'`AsyncTask` - si la station distante est authentifiée avec succès, l'application client commence à configurer le matériel connecté :

```
/Can only come here from permission granted state  
@SuppressLint("MissingPermission")  
List<WifiConfiguration> list =  
    wifiManager.getConfiguredNetworks();  
for (WifiConfiguration i : list) {  
    if (i.SSID != null && i.SSID.equals("\"" +  
        b.getString("WIFINAME") + "\"")) {  
        final ConnectivityManager connectivityManager =  
            (ConnectivityManager) getApplicationContext().  
            getSystemService(Context.CONNECTIVITY_SERVICE);  
        ConnectivityManager.NetworkCallback  
            networkCallback =  
            new ConnectivityManager.NetworkCallback() {  
                @Override  
                public void onAvailable(@NonNull Network network) {  
                    super.onAvailable(network);  
                    //ONLY IF ANDROID 9  
                    if (android.os.Build.VERSION.SDK_INT ==  
                        Build.VERSION_CODES.Q) {  
                        connectivityManager.  
                            bindProcessToNetwork(network);  
                    }  
                }  
            };  
        connectivityManager.  
            registerDefaultNetworkCallback(networkCallback);  
  
        wifiManager.disconnect();  
        wifiManager.enableNetwork(i.networkId, true);  
        wifiManager.reconnect();  
        aT = new PresenceTestAsynctask  
            (getApplicationContext());  
        aT.myAct=mySelf;  
        aT.execute("...");  
        break;  
    }  
}
```

Avec les versions plus récentes d'Android, l'intervention de l'utilisateur est à nouveau nécessaire à ce stade. On commence par la création d'un objet `NetworkSpecifier` :

```
}else{  
    final NetworkSpecifier specifier =  
    new WifiNetworkSpecifier.Builder()  
        .setSsidPattern(new PatternMatcher(  
            b.getString("WIFINAME"),  
            PatternMatcher.PATTERN_PREFIX))  
        .setWpa2Passphrase(networkPass)  
        .build();
```

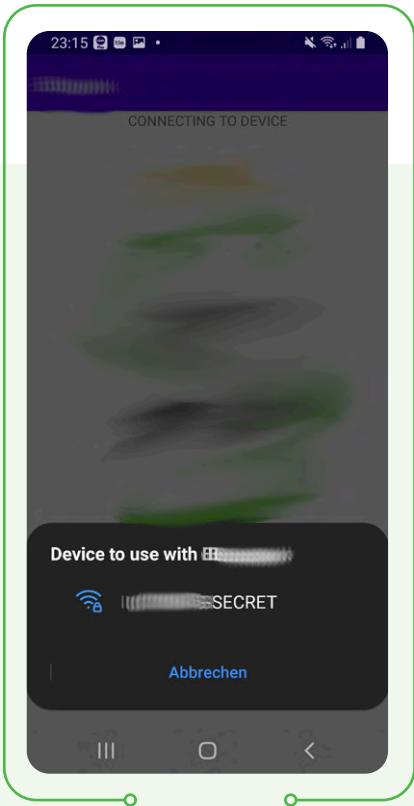


Figure 5. Android Q invite l'utilisateur à établir une connexion.

Les objets `NetworkSpecifier` servent de filtre de recherche pour aider l'interface graphique d'Android à identifier les pairs du réseau sans fil correspondant au cas d'utilisation.

L'étape suivante consiste à créer un élément `NetworkRequest`, comme suit :

```
final NetworkRequest request =
    new NetworkRequest.Builder()
        .addTransportType(NetworkCapabilities.
TRANSPORT_WIFI)
        .removeCapability(NetworkCapabilities.
NET_CAPABILITY_INTERNET)
        .setNetworkSpecifier(specifier)
        .build();
```

La configuration de la connexion proprement dite se déroule comme le montre la **figure 5**. L'utilisateur doit saisir le nom du réseau wifi qui sera la cible de la configuration.

Un inconvénient de cette procédure est que la connexion wifi doit être établie de manière asynchrone. Le retour d'information sur le succès ou l'échec est fourni via un rappel qui doit être créé dans l'application responsable de l'activation :

```
final ConnectivityManager connectivityManager =
(ConnectivityManager)getApplicationContext().
getSystemService(Context.CONNECTIVITY_SERVICE);
final ConnectivityManager.NetworkCallback
networkCallback =
new ConnectivityManager.NetworkCallback() {
    @Override
    public void onAvailable(@NonNull Network network) {
```

```
super.onAvailable(network);
connectivityManager.
bindProcessToNetwork(network);
aT = new PresenceTestAsynctask
(getApplicationContext());
aT.myAct=mySelf;
aT.execute("...");}
}
@Override
public void onUnavailable() {
    super.onUnavailable();
}
};
```

Si la méthode `onAvailable()` est appelée, la connexion a été établie avec succès. Dans ce cas, l'application relance l' `AsyncTask` afin d'authentifier la partie distante et d'effectuer d'autres actions si nécessaire.

La dernière action de notre méthode de balayage consiste à appeler `requestNetwork()`, qui active l'interface utilisateur illustrée ci-dessus.

```
    connectivityManager.
requestNetwork(request, networkCallback);
}
}
```

L'effort en vaut-il la peine ?

En informatique, il y a souvent plusieurs façons de résoudre un problème. Certes, les systèmes d'exploitation en temps réel - je pense ici au RTOS Azure - ainsi que leurs piles d'interfaces graphiques intégrées, fonctionnant sur un microcontrôleur, permettront effectivement de réaliser des interfaces graphiques puissantes dans une solution à appareil unique qui conviendra à la plupart des cas d'utilisation avancés. Pour obtenir la même fonctionnalité que celle fournie par la configuration Android décrite ici, les coûts matériels seraient plus élevés - vous ne bénéficieriez pas d'un écran tactile pour l'interface graphique, car la logique associée, telle qu'une mémoire tampon, augmenterait le coût. En outre, nous devons tenir compte du temps et des efforts nécessaires pour développer le système : si vous souhaitez réaliser, par exemple, le transfert d'images et de relevés de capteurs par courrier électrique dans un système d'exploitation en temps réel, vous devez prévoir quelques jours pour le codage et être conscient que la charge supplémentaire peut avoir un impact sur la latence du système. Android offre une interface graphique élégante sur un appareil que la plupart d'entre nous portent déjà dans leur poche. L'idée de sortir son téléphone et de se connecter à un point d'accès local est plutôt cool.

Si vous décidez de développer ce projet, j'espère que certaines des expériences et des ajustements décrits ici vous seront utiles. Si vous réussissez ou si vous trouvez des améliorations, nous serions ravis de le savoir ; envoyez-nous un courriel. ↗

Des questions, des commentaires ?

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



Produits

- **ESP32-PICO-Kit V4 (avec des connecteurs) (SKU 20323)**
<https://elektor.fr/20323>

- **ESP32-DevKitC-32D (SKU 18701)**
<https://elektor.fr/18701>



LIENS

- [1] Page web de cet article : <https://elektormagazine.fr/210229-04>
- [2] ESP-IDF : <https://idf.espressif.com/>
- [3] Android Studio: <https://developer.android.com/studio>
- [4] Android Documentation: Request permission to access nearby Wi-Fi devices : <https://elektor.link/AndroidWiFiPermission>
- [5] Zen of Palm (2003) [PDF] : <https://cs.uml.edu/~fredm/courses/91.308-fall05/palm/zenofpalm.pdf>

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 pour les membres
- ✓ accès à plus de 5000 fichiers Gerber



Devenez membre
maintenant !



www.elektormagazine.fr/membres

elektor
design > share > earn

filtre actif 1 kHz pour mesures de distorsion

mesures améliorées par l'optimisation du signal de mesure

Alfred Rosenkränzer (Allemagne)

La détection de faibles distorsions dans les signaux exige de fournir un signal de mesure de très bonne qualité au dispositif à tester et un appareil de mesure dont la gamme dynamique et la linéarité sont très étendues. Un filtre en aval d'un générateur de qualité moyenne permet toutefois de fournir des signaux de mesure de haute qualité.

Dans l'article « Filtre coupe-bande de Fliege pour les mesures audio » paru dans Elektor 09/10 2022 [1], il a été montré comment l'utilisation d'un filtre coupe-bande étend la plage de mesure des distorsions avec un numériseur. De cette façon, on élimine le signal de test, ce qui empêche le numériseur d'y ajouter ses propres distorsions. Une condition nécessaire à la détection correcte de faibles niveaux de distorsion du dispositif testé (DUT) est que le générateur de signaux alimentant le DUT produise un signal de haute qualité. Sinon, à la distorsion du circuit testé, on ajoute celle du générateur ce qui constitue un biais de mesure.

Cependant, la qualité du signal généré par les générateurs de signaux abordables donc non-professionnels laisse souvent à désirer. En plaçant un filtre en aval du générateur on peut atténuer les distorsions de celui-ci et faire un petit miracle en fournissant un signal de mesure très propre.

Exigences

La plupart des mesures audio étant effectuées à la fréquence type d'1 kHz, un filtre à bande passante fixe ou à fréquence de coupure (F_c) de 1 kHz est généralement suffisant. Le filtre doit laisser passer la fréquence d'essai sans l'atténuer, mais atténuer ses harmoniques (les multiples de celle-ci) autant que possible. Une autre condition sine qua non est que le filtre lui-même ne doit pas produire de distorsion significative.

Pour faciliter les essais, il est utile que ce 1^{er} filtre ait une certaine largeur de bande, pas trop étroite. Cela permet en effet d'adapter la fréquence du signal de mesure à la fréquence exacte du 2^e filtre coupe-bande à bande étroite sans variations de niveau et faire que ce dernier élimine le signal de mesure avant la mesure. La **figure 1** montre le principe de cette mesure avec les deux filtres en amont et aval du DUT.

Filtre

En principe, la réduction de la distorsion du générateur par atténuation des harmoniques peut être obtenue avec un filtre passe-bande, mais aussi avec un filtre passe-bas approprié. Le mieux est de caractériser l'atténuation des harmoniques pour divers types et configuration de

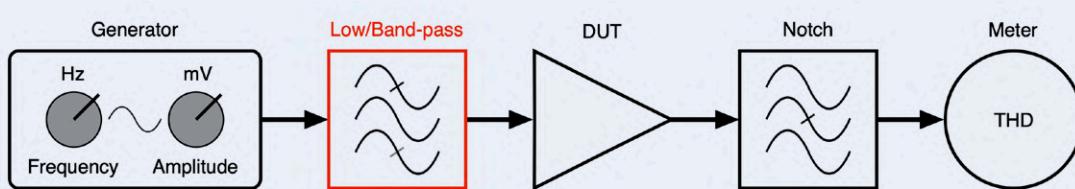


Figure 1. Principe de mesure optimisé pour la distorsion avec le DUT entre deux filtres. Le filtre amont (rouge) passe-bas ou passe-bande élimine la distorsion du signal du générateur.

filtres à 1 kHz et réunir les résultats (**tableau 1**) afin de comparer un filtre passe-bande d'ordre 4 à différents filtres passe-bas) et retenir les combinaisons ayant le meilleur rapport atténuation/coût. L'atténuation à 2 et 3 kHz est particulièrement pertinente car ces harmoniques sont les plus marquées sur la plupart des générateurs.

Tableau 1. Amortissement du filtre

Fréquence (kHz)	BP ord. 4. (dB)	LP Bw. ord. 4. (dB)	LP Ch. ord. 4. (dB)	LP Bw. ord. 8. (dB)
1,0	0,0	0,1	0,3	0,0
2,0	-45,0	-11,7	-26,0	-32,0
3,0	-56,0	-25,2	-42,0	-60,0
4,0	-62,0	-35,0	-52,0	-80,0
5,0	-66,0	-42,8	-60,0	-90,0
6,0	-69,0	-49,5	-67,0	-90,0
7,0	-72,0	-54,0	-72,0	-90,0
8,0	-74,0	-59,0	-77,0	-90,0
9,0	-75,0	-63,0	-81,0	-90,0
10,0	-76,0	-67,0	-85,0	-90,0

Un filtre passe-bas de Butterworth (Bw) d'ordre 4 n'atténue même pas de 12 dB à 2 kHz et ne convient donc pas à ce type d'application. Un filtre passe-bas de Chebyshev (Ch) d'ordre 4 atteint déjà -26 dB ici. Un filtre passe-bas de Butterworth d'ordre 8 et un filtre passe-bande d'ordre 4 avec une BP de 100 Hz font encore mieux. Voyez la compa-

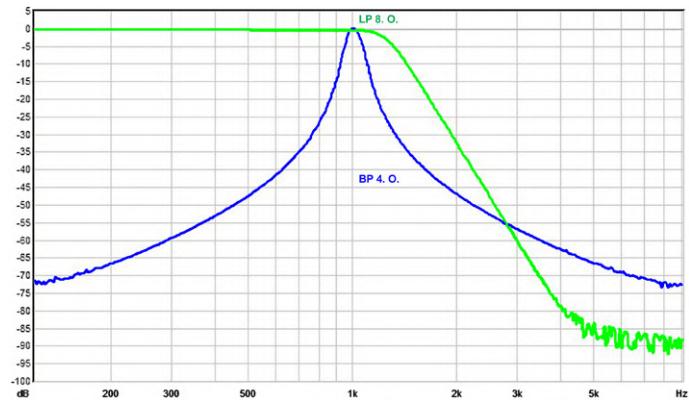


Figure 2. Comparaison des réponses en fréquence d'un filtre passe-bas de Bw d'ordre 8 (vert) et d'un filtre passe-bande d'ordre 4 (bleu).

raison des réponses en fréquence de ces deux filtres **figure 2**. Comme ces deux filtres sont à privilégier, nous avons développé des cartes à circuit imprimé pour eux (même taille que le filtre coupe-bande de [1]). La carte régulateur de tension d'alors convient ici aussi.

Schémas

La **figure 3** montre le schéma du filtre passe-bas de Bw d'ordre 8. La valeur spécifiée des composants donne une F_c d'1,28 kHz et évite toute atténuation indésirable sur une plage de quelques dizaines de Hz autour de 1 kHz. Les **figures 4** et **5** montrent l'implantation des composants et le prototype du filtre passe-bas. Pour chaque résistance et condensateur déterminant la fréquence, un composant additionnel est prévu en parallèle (aucune valeur sur la **figure 3**).

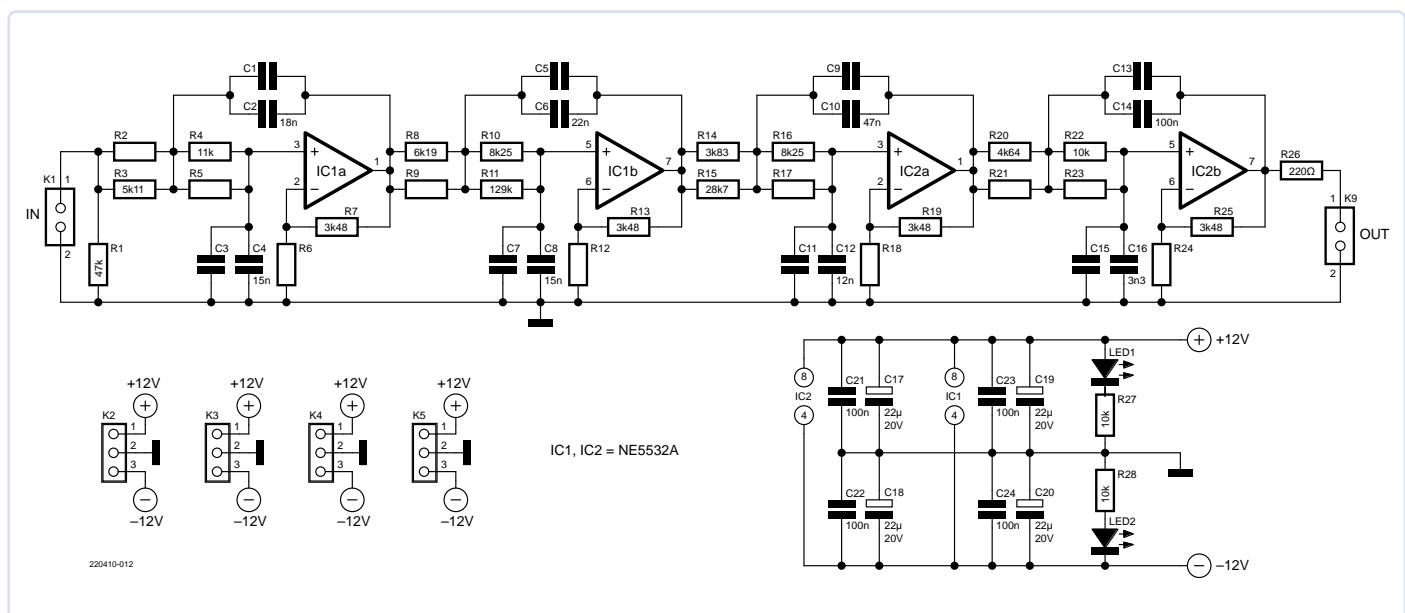


Figure 3. Schéma du filtre passe-bas de Bw d'ordre 8.

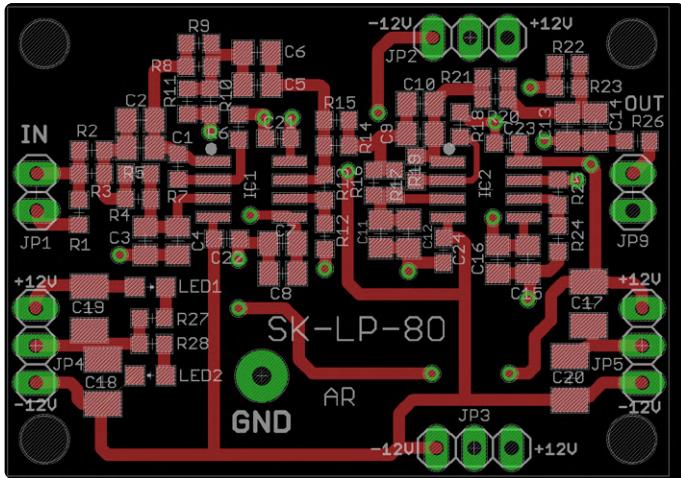


Figure 4. Implantation de la carte pour le filtre passe-bas de Bw d'ordre 8.



Figure 5. Le prototype passe-bas terminé de l'auteur.

Ainsi, la série E disponible dans le commerce permet d'atteindre avec précision la valeur calculée. Mesurez avec précision les condensateurs choisis avant de les souder sous peine de remplacement ultérieur de composants. La carte prévoit également de la place pour des résistances supplémentaires (R6, R12, R18 et R24, réglage éventuel du gain du filtre), mais elles ne sont pas nécessaires ici et sont donc omises. Mieux vaut lire l'article sur le passe-bande de Fliege [1] et, si nécessaire, celui sur le logiciel de filtrage [2], car ils expliquent en détail la procédure du réglage exact des fréquences du filtre et le logiciel de calcul des composants utilisé à cet effet. Exemple d'optimisation : La valeur mesurée de C14 de 100 nF est de 103 nF, c'est dans la tolérance de 5 % du composant. Dès lors, en entrant une valeur de 103 nF dans

le logiciel, on obtient les valeurs ajustées des autres composants de l'étage à une fréquence donnée, or les résistances se trouvent facilement et à bon marché, même avec une tolérance de 1 %.

La procédure vaut aussi pour le filtre passe-bande (schéma : **figure 6**, implantation : **figure 7**, prototype : **figure 8**). Pour les composants déterminant la fréquence, des composants supplémentaires ont été prévus pour un éventuel montage en parallèle, et une amplification optionnelle utilise R8, R9, R18 et R19. Les valeurs des condensateurs déterminant la fréquence étant identiques, mieux vaut utiliser des composants du même lot et compenser les différences – qui seront probablement très faibles – par des condensateurs en parallèles. Là aussi, il est recommandé de saisir la valeur obtenue dans le logiciel.

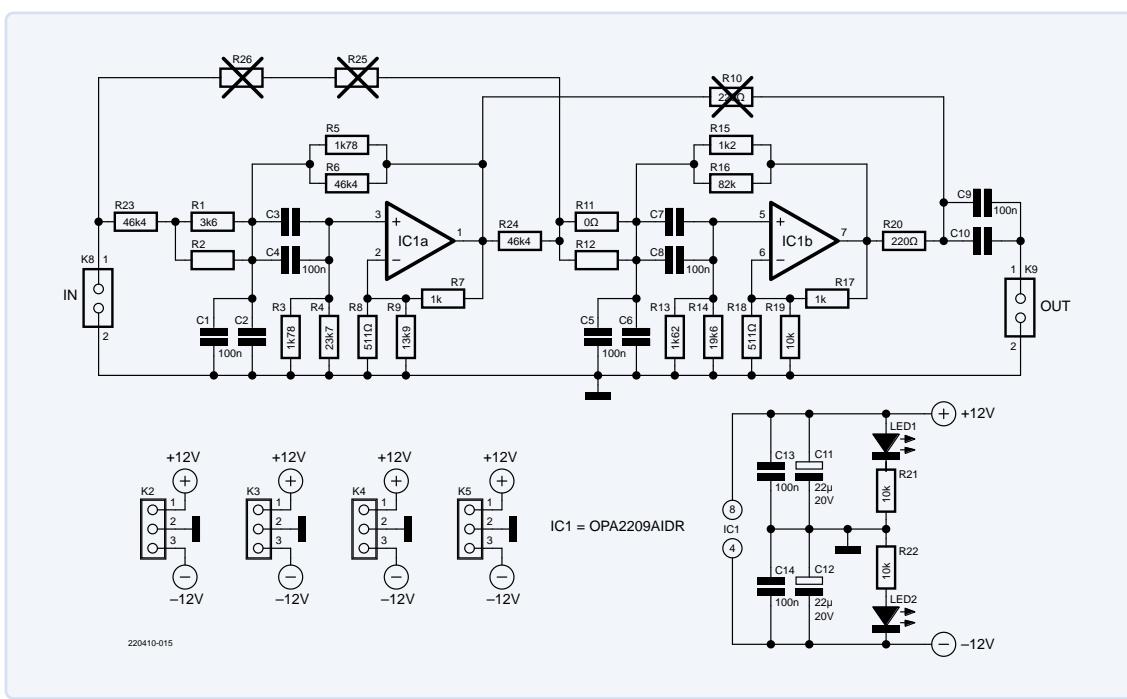


Figure 6. Schéma du filtre passe-bande d'ordre 4 avec une BP de 100 Hz.

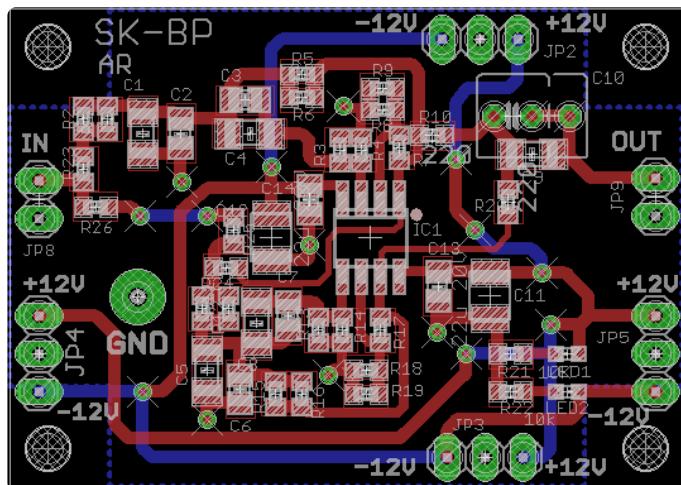


Figure 7. Disposition de la carte pour le filtre passe-bande d'ordre 4.

Pour atteindre la meilleure précision possible pour les résistances en entrée de chaque étage, une combinaison d'une résistance en série avec deux résistances en parallèle a été prévue dans chaque cas. R10, R25 et R26 ne sont utilisées que pour mesurer les caractéristiques des différents étages comme décrit dans [1] et sont donc omises. Pour un réglage fin, il est recommandé de simuler le circuit avec les valeurs réelles mesurées.

Quel filtre ?

Ayant maintenant le choix d'utiliser un filtre passe-bas ou un filtre passe-bande, la question se pose naturellement : quel type de filtre est le mieux adapté au besoin ?

Pour une atténuation correcte, un filtre passe-bas doit être d'ordre plus élevé, il utilise donc quatre AOP. Cela augmente le bruit intrinsèque et, bien sûr, la distorsion produite par le filtre. En revanche, à partir d'environ 3 kHz, il atténue plus que le passe-bande les harmoniques d'ordre supérieur. Il laisser passer toutes les fréquences $< F_c$, cela peut favoriser d'autres applications.

Le passe-bande d'ordre 4, s'en sort avec seulement deux AOP, il doit donc produire moins de bruit et de distorsion intrinsèques. Ne laissant passer que les fréquences proches d'1 kHz avec une bande passante de 100 Hz, il ne convient guère à d'autres applications. Mais, atténuant toutes les fréquences inférieures à 1 kHz et donc des signaux indésirables comme le ronflement secteur, le bruit BF intrinsèque est aussi réduit, ce qui peut profiter aux mesures du rapport S/B.

Divers

À cause des minuscules composants CMS utilisés, le soudage des cartes n'est pas adapté aux débutants en soudage manuel. Les électroniciens expérimentés ne devraient cependant pas rencontrer de problèmes. Les fichiers d'implantation au format Eagle sont téléchargeables gratuitement sur la page de cet article [3]. Toutes les résistances impliquées dans le signal doivent être de haute qualité et de type couche mince. Pour garantir des caractéristiques stables (dont une faible distorsion intrinsèque), seuls les condensateurs céramiques CMS de type COG sont utilisables pour véhiculer le signal.

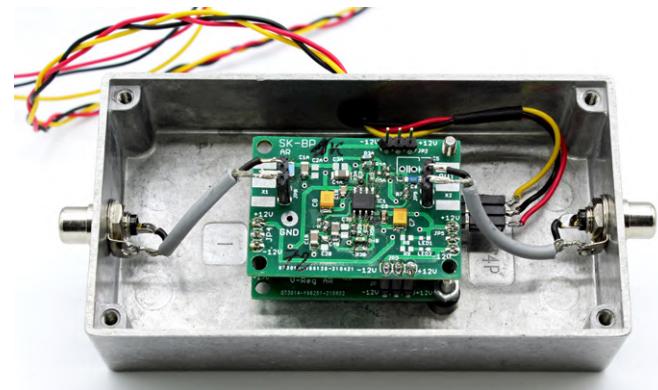


Figure 8. Prototype de passe-bande terminé, monté dans un boîtier en alu.

L'auteur dispose encore de quelques cartes nues. Au besoin, vous pouvez le contacter par courriel (voir « À propos de l'auteur »). Les deux filtres présentés ici ne sont pas limités à 1 kHz. En appliquant les considérations de conception de [1], on peut facilement réaliser un ensemble de filtres passe-bas, passe-bande et coupe-bande adapté à différentes fréquences, et l'utiliser pour étudier la distorsion de circuits audio à différentes fréquences. ↗

VF : Yves Georges — 220410-04

À propos de l'auteur

Alfred Rosenkränzer a longtemps travaillé pendant comme ingénieur de développement, au début dans les techniques de la télévision professionnelle. Peu avant l'an 2000, pour tester des CI, il développe des circuits numériques à haut débit et analogiques. Il n'a cessé de se passionner pour l'audio.

Des questions, des commentaires ?

Envoyez un courriel à l'auteur (alfred_rosenkraenzer@gmx.de) ou contactez Elektor (redaction@elektor.fr).

LIENS

- [1] Alfred Rosenkränzer, « Filtre coupe-bande de Fliege pour les mesures audio », Elektor 9-10/2022 : <https://www.elektormagazine.fr/magazine/elektor-275/61027>
- [2] Alfred Rosenkränzer, « logiciel de calcul de filtres », Elektor 11-12/2022 : <https://www.elektormagazine.fr/magazine/elektor-283/61165>
- [3] Page web de cet article : <https://elektormagazine.fr/220410-04>



LISTE DES COMPOSANTS : PASSE-BAS

Résistances :

(Film mince, 1 %, SMD 0603)

R1 = 47 k

R2, R5, R9, R17, R21, R23 = connexion parallèle *

R3 = 5,11 k

R4 = 11 k

R6, R12, R18, R24 = gain, non applicable *

R7, R13, R19, R25 = 3,48 k

R8 = 6,19 k

R10, R16 = 8,25 k

R11 = 129 k

R14 = 3,83 k

R15 = 28,7 k

R20 = 4,64 k

R22, R27, R28 = 10 k

R26 = 220 Ω

Condensateurs :

(Sauf indication contraire : 5 %, COG, SMD0805)

C1, C3, C5, C7, C9, C11, C13, C15 = connexion parallèle *

C2 = 18 n

C4, C8 = 15 n

C6 = 22 n

C10 = 47 n

C12 = 12 n

C14 = 100 n

C16 = 3,3 n

C17, C18, C19, C20 = Condensateur électrolytique 22 μ / 20 V, SMD-B

C21, C22, C23, C24 = 100 n, SMD0603

Semi-conducteur :

LED1, LED2 = LED, green, SMD0805

IC1, IC2 = NE5532A, SO08

Aussi :

K1, K9 = connecteur mâle, 2 broches, RM 1/10"

K2..K5 = connecteur mâle, 3 broches, RM 1/10"

* Voir texte



LISTE DES COMPOSANTS : PASSE-BANDE

Résistances :

(Film mince, 1 %, SMD 0603)

R1 = 3,6 k

R2, R12 = connexion parallèle *

R3, R5 = 1,78 k

R4 = 23,7 k

R6, R23, R24 = 46,4 k

R7, R17 = 1 k

R8, R18 = 511 Ω

R9 = 13,9 k

R10, R25, R26 = non applicable *

R11 = 0 Ω

R13 = 1,62 k

R14 = 19,6 k

R15 = 1,2 k

R16 = 82 k

R19, R21, R22 = 10 k

Condensateurs :

(Sauf indication contraire : 5 %, COG, SMD0805)

C1, C4, C5, C8, C9 = 100 n, SMD1206

C2, C3, C6, C7 = connexion parallèle *

C10 = non applicable

Semi-conducteurs :

LED1, LED2 = LED, verte, SMD0603

IC1 = OPA2209AIDR, SO08

Aussi :

K8, K9 = Connecteur mâle, 2 broches, RM 1/10"

K2..K5 = Connecteur mâle, 3 broches, RM 1/10"

* Voir texte



Produits

➤ OWON AG051 générateur de forme d'ondes arbitraires (5 MHz) (SKU 18874)

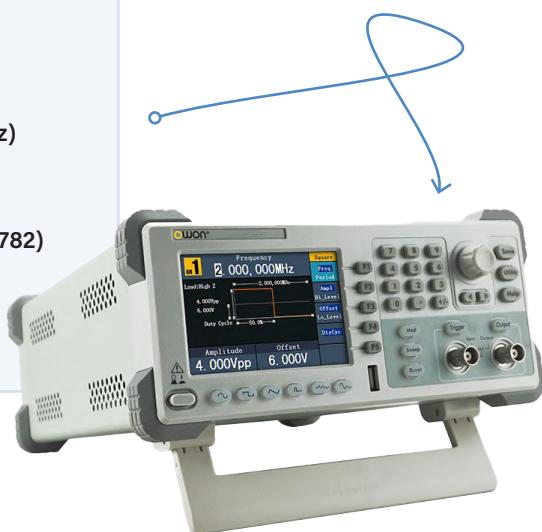
<https://elektor.fr/18874>

➤ OWON SDS1102 oscilloscope à deux voies (100 MHz) (SKU 18782)

<https://elektor.fr/18782>

➤ OWON XSA810 analyseur de spectre (1 GHz) (SKU 19714)

<https://elektor.fr/19714>



démarrer en électronique

... les multivibrateurs

Eric Bogers (Elektor)

Dans l'épisode précédent, nous avons (enfin) abordé les transistors, que l'on peut facilement qualifier de « cœur battant de notre société ». Sans les transistors, notre vie moderne serait tout simplement impossible. Il est maintenant temps d'examiner les multivibrateurs, puis d'aborder les transistors en tant qu'amplificateurs.

Tout d'abord, une brève remarque, destinée principalement aux nouveaux lecteurs d'Elektor qui découvrent leur premier épisode de la série « démarrer en électronique » : cette série ne prétend en aucun cas être un cours théorique approfondi d'électronique. Notre seul objectif est de donner aux novices en électronique suffisamment de connaissances de base pour qu'ils puissent concevoir et réaliser des circuits avec une chance raisonnable de succès, ou modifier un circuit existant et obtenir un résultat exploitable. Pour ce faire, nous faisons de notre mieux pour éviter les calculs sérieux, et nous évitons parfois les détails et les aspects théoriques. Nous sommes bien conscients qu'avec cette approche, nous provoquerons parfois de vives critiques de la part des perfectionnistes et des théoriciens parmi nos lecteurs. Nous nous en excusons, sans trop en faire... À la fin du dernier épisode, nous avons parlé du multivibrateur bistable, qui est aussi appelé bascule (flip-flop). Pour vous rafraîchir la mémoire, nous reproduisons ici le schéma de base de ce multivibrat-

teur à la **figure 1**. Comme le suggère son nom, ce circuit possède deux états, chacun avec l'un des deux transistors conducteur (mais pas les deux). Nous pouvons utiliser les boutons-poussoirs pour passer d'un état à l'autre. Il est impossible de prédir lequel des deux transistors commencera à conduire immédiatement après la mise sous tension - la seule chose dont nous pouvons être sûrs, c'est que l'un d'entre eux sera passant et l'autre bloqué.

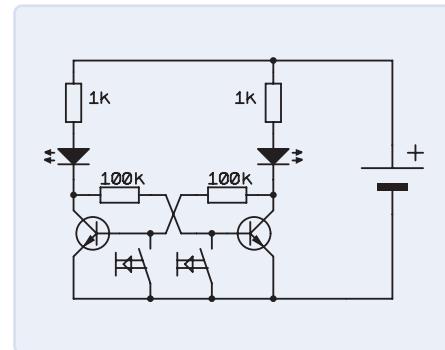


Figure 1. Multivibrateur bistable ou bascule.

Multivibrateur monostable

Le multivibrateur monostable (ou *monoflop*) est dérivé du bistable et possède un seul état stable, comme vous pouvez le deviner d'après son nom. Le schéma de base est illustré à la **figure 2**. Bien que ce circuit puisse être mis dans un état différent lorsqu'il est déclenché par une impulsion externe, après un certain temps, il reviendra automatiquement à son état initial stable. Vous connaissez peut-être ce type de circuit grâce aux circuits d'éclairage d'escalier pratiques, qui s'éteignent automatiquement au bout d'un certain temps.

Dans le circuit illustré à la **figure 2**, le transistor de gauche est généralement conducteur. Tout comme la bascule (**figure 1**), nous pouvons provoquer le blocage de ce transistor en mettant brièvement sa base à la masse en appuyant sur le bouton-poussoir. Cela entraîne également la décharge du condensateur électrolytique. Ensuite, le transistor de droite commence à conduire, mais temporairement, car le condensateur électrolytique est rechargeé

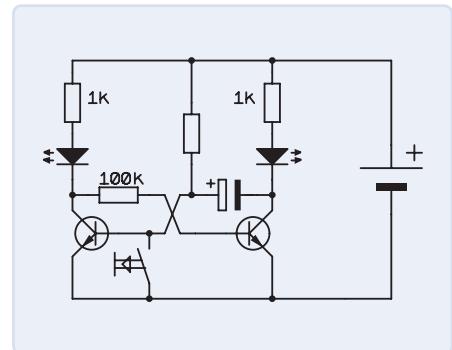


Figure 2. Multivibrateur monostable.

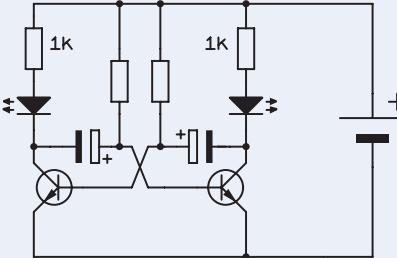


Figure 3. Multivibrateur astable.

à travers sa résistance en série. Lorsque la tension sur ce condensateur augmente jusqu'à environ 0,7 V, le transistor de gauche recommence à conduire, ce qui fait chuter sa tension collecteur-émetteur à environ 0. En conséquence, le transistor de droite est à nouveau bloqué, et le circuit revient à son état initial stable.

Le temps nécessaire au monoflop pour revenir à son état stable dépend, comme prévu, des valeurs de la résistance en série et du condensateur, ainsi que de la valeur de la tension d'alimentation. Essayons d'estimer grossièrement ce temps (l'intervalle monoflop) en réfléchissant un peu. Nous supposons ici que : $R = 100 \text{ k}\Omega$, $C = 100 \mu\text{F}$, and $U_V = 12 \text{ V}$. La tension d'un condensateur est donnée par la formule suivante :

$$U_C = \frac{Q}{C} = \frac{I \cdot t}{C}$$

Le courant dépend de la tension aux bornes de la résistance et de sa valeur. Cette tension baisse progressivement au fur et à mesure que la tension du condensateur augmente. Puisque le circuit change d'état à une tension de condensateur de seulement 0,7 V, nous pouvons ignorer le comportement non linéaire ici et utiliser la valeur moyenne arithmétique dans notre calcul.

$$U_C = \frac{U_V \cdot t}{R \cdot C} \Rightarrow t = \frac{U_C \cdot R \cdot C}{U_V}$$

Cela signifie que la tension moyenne sur la résistance est d'environ 11,65 V, puisque le circuit bascule lorsque la tension du condensateur atteint 0,7 V.

$$t = \frac{0,7 \text{ V} \cdot 100 \text{ k}\Omega \cdot 100 \mu\text{F}}{11,65 \text{ V}} = 0,6 \text{ s}$$

Multivibrateur Astable

Un multivibrateur astable n'a pas d'état stable. Il bascule donc entre deux états possibles, d'où son nom d'astable. Le circuit de la **figure 3** ressemble un peu à un double monoflop, vous pourriez donc envisager d'utiliser les mêmes formules que celles dérivées ci-dessus pour le monoflop afin de calculer le timing - mais dans ce cas, vous vous tromperiez d'un facteur d'environ 10. Cela

soulève naturellement la question de savoir comment ce circuit fonctionne réellement. Supposons que le transistor de gauche soit conducteur : alors la tension U_{CE} du transistor de gauche est presque nulle et la tension collecteur-émetteur du transistor de droite est égale à la tension d'alimentation. Le condensateur de gauche est progressivement chargé à travers sa résistance en série, et à un certain moment, la tension sur ce condensateur atteindra 0,7 V et le transistor de droite sera passant.

Lorsque le transistor de droite est passant, la tension de U_{CE} est presque égale à la valeur de la tension d'alimentation, ce qui signifie que la tension du condensateur diminue d'autant. En conséquence, la tension sur la base du transistor de gauche devient négative et le transistor se bloque.

Cela signifie également que le condensateur doit être chargé par une tension négative presque égale à la valeur de la tension d'alimentation, au lieu de commencer par 0 V. Cela prend bien sûr beaucoup plus de temps. Essayons de calculer cela. Le condensateur doit être chargé de -12 V à +0,7 V. Si nous supposons que la courbe de charge est linéaire, nous introduisons une erreur d'environ 10 %, ce qui correspond à la tolérance d'un condensateur électrolytique de bonne qualité. Mais, essayons d'être un peu plus précis. La tension aux bornes d'un condensateur est donnée par la formule suivante :

$$U_C = U_0 \cdot \left(1 - e^{-\frac{t}{RC}} \right)$$

Or, dans ce cas, U_0 est égale au double de la valeur de la tension d'alimentation et le condensateur est chargé jusqu'à $U_V + 0,7 \text{ V}$. Nous pouvons donc écrire l'équation suivante :

$$U_C + U_V = 2 \cdot U_V \cdot \left(1 - e^{-\frac{t}{RC}} \right)$$

Nous pouvons résoudre cette équation en fonction de la variable t . Ici, ne vous préoccupez pas de la dérivation, nous vous donnons simplement le résultat :

$$t = -R \cdot C \cdot \ln \left(1 - \frac{U_C + U_V}{2 \cdot U_V} \right)$$

Si l'on calcule cela avec les mêmes valeurs de composants que pour le monoflop ($R = 100 \text{ k}\Omega$ et $C = 100 \mu\text{F}$), le résultat est $t = 7,5 \text{ s}$, soit une durée 12 fois plus longue qu'avec la monoflop.

Transistor bipolaire comme amplificateur

L'utilisation d'un transistor bipolaire pour réaliser un amplificateur n'est pas réellement facile. En effet, tous les paramètres, dont aucun n'est vraiment linéaire, dépendent les uns des autres et malheureusement de la température. Si vous ne prenez pas les mesures adéquates, le point de fonctionnement sera très différent de ce que vous aviez prévu.

C'est pourquoi les transistors sont rarement utilisés pour les applications à petits signaux, pour lesquelles les amplificateurs opérationnels (ampli-op) constituent le composant idéal. Bien que les ampli-op soient constitués de transistors, la plupart des problèmes ont déjà été résolus par les concepteurs. Il est donc très facile de construire des amplificateurs stables avec des ampli-op.

Cependant, les ampli-op ne sont adaptés qu'au fonctionnement en petits signaux, car leur tension et/ou leur courant de sortie maximal est trop faible pour les amplificateurs de puissance. Pour de telles applications, vous pouvez combiner les ampli-op avec des transistors (si nécessaire des transistors de puissance), ou vous pouvez construire tout le circuit avec des transistors discrets.

Nous aborderons les amplificateurs dans le prochain épisode. Pour faciliter la compréhension, nous examinons ici brièvement les transistors comme composants d'amplification.

À ce stade, dans la plupart des manuels, vous trouverez beaucoup de théories et d'équations, dont la plupart ne sont pas pertinentes dans la pratique. La seule relation vraiment importante est la caractéristique

U/I de la jonction base-émetteur - et nous l'avons déjà abordée dans le chapitre sur les diodes (voir **figure 4**). Cependant, une chose que nous n'avons pas mentionnée dans cette étude est que la caractéristique U/I dépend de la température. Lorsqu'une diode se réchauffe, le courant augmente (en supposant que la tension reste la même). La diode devient alors encore plus chaude. Ce comportement entraîne rarement des problèmes avec les diodes ordinaires, mais avec les transistors, il est gênant parce que le courant de collecteur change également - et le courant de collecteur peut augmenter d'un facteur de 100 que le courant de base. En effet, la formule suivante s'applique au courant de collecteur :

$$I_C = I_B \cdot h_{FE} = I_B \cdot \beta$$

Ici, h_{FE} (également appelé β) est le gain en courant du transistor. Ce gain en courant est non seulement sujet à une large variation entre les composants du même type, mais (pour compliquer encore les choses) il dépend également du courant du collecteur. Pour les transistors à petits signaux, la valeur de h_{FE} est comprise entre 100 et 800, et pour les transistors de puissance, elle est comprise entre 10 et 100. Le gain en courant étant un paramètre si peu fiable, vous devez essayer de concevoir vos circuits

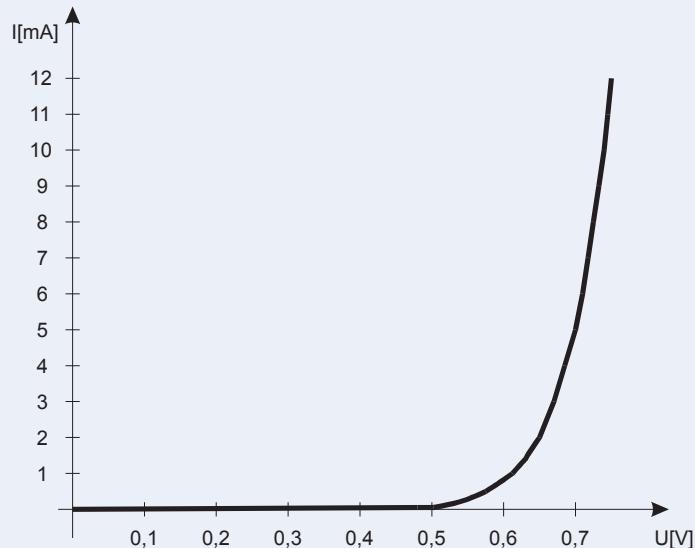


Figure 4. Caractéristique U/I d'une diode.

à transistors de manière que la valeur exacte de h_{FE} ne soit pas un facteur important. Nous verrons comment y parvenir dans le prochain épisode. ↗

230033-04

Note de l'éditeur : La série d'articles « démarer en électronique » est basée sur le livre « Basiskurs Elektronik » de Michael Ebner, publié par Elektor.



Produits

➤ **B. Kainka, Basic Electronics for Beginners (Elektor, 2020) (SKU 19212)**
www.elektor.fr/19212

➤ **B. Kainka, Basic Electronics for Beginners (Elektor, 2020) (E-Book, SKU 19213)**
www.elektor.fr/19213

Des questions, des commentaires?

Contactez Elektor (redaction@elektor.fr).

YOUR KEY TO CELLULAR TECHNOLOGY



WÜRTH
ELEKTRONIK
MORE THAN
YOU EXPECT

WE meet @
PCIM Europe
Hall 6-217

Adrastea-I is a Cellular Module with High Performance, Ultra-Low Power Consumption, Multi-Band LTE-M and NB-IoT Module.

Despite its compact size, the module has integrated GNSS, integrated ARM Cortex M4 and 1MB Flash reserved for user application development. The module is based on the high-performance Sony Altair ALT1250 chipset. The Adrastea-I module, certified by Deutsche Telekom, enables rapid integration into end products without additional industry-specific certification (GCF) or operator approval. Provided that a Deutsche Telekom IoT connectivity (SIM card) is used. For all other operators the module offers the industry-specific certification (GCF) already.

www.we-online.com/gocellular

- Small form factor
- Security and encryption
- Long range/worldwide coverage
- Multi-band support

#GOCELLULAR

projet 2.0

corrections, mises à jour et messages de lecteurs

Compilé par Jens Nickel (Elektor)



Cultivez-le vous-même

Elektor : édition spéciale (Arduino) 2022, p. 54 (220414)

J'ai lu votre article dans Elektor avec intérêt. J'ai planté des graines dans le passé, mais ça n'a pas eu beaucoup de succès parce que je n'ai pas donné l'attention que les graines demandent. J'ai fini par faire don de mes serres de bureau à ma petite amie, qui s'intéresse davantage aux plantes et à leur entretien. Quoi qu'il en soit, j'ai beaucoup aimé votre article et je veux construire quelque chose de similaire pour ma copine. Vous mentionnez que les plantes préfèrent la lumière de 450 nm et 650 nm. Les LED RVB produisent des longueurs d'onde de 460, 525 et 625 nm (recherche rapide sur Google). Ces valeurs ne suivent pas tout à fait les préférences des plantes. Sont-elles assez proches ? Elektor Magazine a déjà publié en 2019, en collaboration avec Würth Elektronik, un article sur une serre éclairée par des LED (elektormagazine.fr/magazine/elektor-111/51064). Votre article m'a incité à approfondir ce sujet. J'ai un générateur de brouillard à ultrasons qui traîne ici depuis de nombreuses années, il pourrait donc enfin devenir utile.

Michael Hompus



Testeur audio à bas prix

Elektor 7-8/2022, p. 6 (200604)

Le logiciel REW (roomeqwizard.com) est très largement utilisé dans la communauté DIY. REW est essentiellement un logiciel gratuit, bien que vous puissiez (ou devez) faire un don à l'auteur. Il devrait être capable de faire tout ce qui est mentionné dans l'article. Pour ceux qui ont des exigences nettement plus élevées, il existe également une version Pro, mais elle coûte 100 dollars. Il y a un long fil de discussion sur REW sur diyaudio.com, qui traite également des différentes interfaces et de la façon de les modular.

Frank von Zeppelin



Merci pour votre question ! Il y a quelque temps, après avoir publié le projet GIY en ligne, j'ai commencé à faire des recherches sur les lumières LED pour la croissance des plantes, et vous êtes invités à lire mon résumé à ce sujet ici :

instructables.com/DIY-Grow-LED-Light-Designing-a-Better-Sun

Pour répondre à votre question, et pour vous aider à appliquer concrètement le dernier savoir-faire du milieu : utilisez des LED de couleur blanche combinées à des LED à spectre rouge ou rouge lointain. L'idée est de recréer un spectre complet similaire à celui du soleil, puis d'en augmenter l'efficacité en ajoutant quelques LED rouges/rouges lointains/UV supplémentaires.

Pour le blanc, je recommande les diodes Samsung LM301B ou LM301H. Pour le rouge, des LED Osram hyper-rouges. En pratique, cela donnera à peu près ceci : 6 à 10 LED blanches plus deux LED rouges sur une barre lumineuse phyto. Il ne vous reste plus qu'à jouer avec la température de couleur des LED blanches, si vous choisissez le blanc froid, vous obtiendrez plus de bleu dans le spectre, si vous choisissez le blanc chaud, vous obtiendrez plus de rouge dans le spectre, et ainsi de suite. J'espère que l'idée de base est claire.

Consultez cityfarm.md pour voir la version finale du projet GIY.

Dmitrii Albot (auteur de l'article)



The Tube

Elektor 1-2/2023, p. 70 (220089)

Je suis abonné depuis de nombreuses années. Mon premier projet Elektor a été un tuner FM avec des bobines imprimées en 1975, alors que j'avais environ 20 ans. Je suis heureux de constater qu'au fil des décennies, vous avez adapté le concept de votre magazine et toutes les offres connexes aux besoins réels des gens. Continuez à faire du bon travail ! Dans le dernier numéro, cependant, je me suis interrogé sur certains points de l'article « The tube », peut-être s'agit-il simplement de fautes de frappe ? Ou peut-être ai-je mal compris quelque chose ?

a (negative feedback over all)

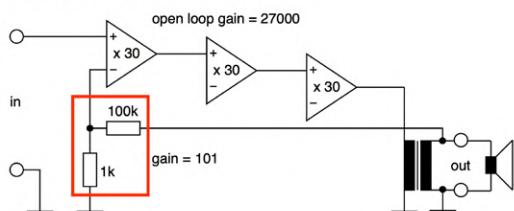


Figure 1a : l'explication dans le texte suggère (indirectement) qu'aucune rétroaction négative n'est appliquée dans les différents étages (par exemple, par des résistances de cathode). J'ai du mal à imaginer que cette approche soit réellement possible. Sinon, l'auteur aurait dû signaler ce cas particulier, non ?

b (local negative feedback only)

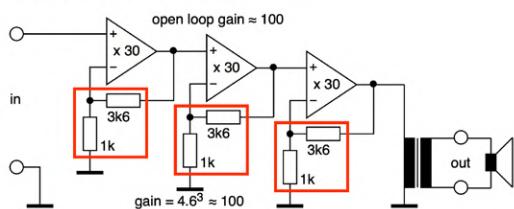


Figure 1b : avec les valeurs de résistance indiquées, l'amplification globale (gain) devrait en fait être de 4,3 au cube (79,5), n'est-ce pas ? (Et d'ailleurs, 3,3 au cube fait environ 36, ce qui est loin de 100.) Et, est-il même possible de parler de gain en boucle ouverte dans cette situation ? Je pense que le terme « gain global du chemin de l'amplificateur » serait plus approprié.

Harald Sonnemann

Vous avez tout à fait raison de dire qu'aucun amplificateur audio viable ne peut se passer de rétroaction négative, car les composants d'amplification ne sont pas suffisamment linéaires. Les amateurs de tubes utilisent souvent l'expression « pas de rétroaction négative » comme synonyme de « pas de rétroaction négative globale ».

Ce que j'ai fait, c'est opposer spécifiquement la rétroaction négative globale au principe de la rétroaction négative locale par étage d'amplification ; voir la figure 1 de l'article.

Dans le feu de l'action, cependant, je me suis effectivement trompé dans les calculs. Comme vous le dites, 4,3 au cube fait 79 et non 100, et 3,3 au cube ne fait certainement pas 100. Je n'ai pas remarqué cette erreur lors de la relecture du texte, pas plus que l'auteur lors de l'acceptation de l'article. Un schéma de circuit révisé est présenté ici.

Et bien sûr, « gain en boucle ouverte » dans la figure 1b signifie en fait un gain en boucle quasi ouverte.

Thomas Scherer (rédacteur de l'article)



Démarrer en électronique

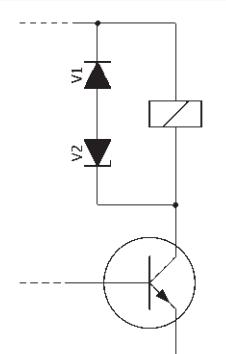
Elektor 9-10/2022, p. 34 (220256)

Projet 2.0

Elektor 1-2/2023, p. 110 (220601)

Malheureusement, une erreur s'est glissée dans ma proposition, publiée sous forme de message de lecteur dans le numéro de janvier 2023.

J'y suggérais de connecter une diode Zener en série avec la diode en roue libre lorsqu'on pilote un relais avec un transistor. Cela permet non seulement de raccourcir considérablement le temps de coupure du relais, mais aussi d'augmenter la durée de vie des contacts du relais, car la séparation des contacts est nettement plus rapide. Malheureusement, dans le dessin associé, la diode Zener a été représentée dans le mauvais sens, de sorte que seules les tensions directes des deux diodes sont superposées, ce qui n'apporte pas grand-chose. Sur la figure ci-contre, la diode Zener est connectée dans le bon sens.



Lorsque le transistor s'éteint, la tension inverse V2 de la diode Zener s'ajoute à la tension directe V1 de la diode de redressement. Cela entraîne une chute beaucoup plus rapide du courant à travers le relais. Ce phénomène peut non seulement être mesuré, mais aussi être clairement entendu.

À cet égard, il est important de se rappeler que la tension entre le collecteur et l'émetteur du transistor est la somme de la tension d'alimentation, de la tension inverse de la diode Zener et de la tension directe de la diode de redressement. Avec une tension d'alimentation de 12 V et une diode Zener de 10 V, au moment de l'extinction, le transistor voit une tension d'environ 22,6 V, ce qui doit être pris en compte lors du choix du transistor et de la diode Zener.

Thomas Klingbeil

le nouveau protocole I₃C

effet d'annonce ou vrai progrès ?

Tam Hanna (Slovaquie)

Aucun protocole de bus n'a connu un succès aussi grand que l'I²C lancé dans les années 1980 par Philips Semiconductor. Cependant, les faiblesses de ce système de bus sont apparues ces dernières années. L'I₃C, son successeur est en embuscade et cet article souhaite vous le présenter en détail.

Pour comprendre les possibilités et limites du bus I²C (*Inter Integrated Circuits*), il faut garder en mémoire qu'il est dans le domaine public depuis l'expiration des brevets détenus par *Philips*. Bien que *NXP* (successeur de la division semi-conducteurs de *Philips*) détienne les droits de propriété intellectuelle sur le logo et le nom, l'exploitation du protocole I₂C est désormais libre.

Le débat I²C/TWI (*Two Wire Interface* = interface bifilaire) est maintenant pratiquement éteint, mais des commentaires intéressants sont encore en ligne (par ex. sur [1]), notamment : « L'idée derrière le droit de licence était de s'assurer que les entités qui souhaitaient fabriquer du silicium I²C (et le vendre en tant que tel) soutenaient la norme et contribuaient ainsi à son développement ». Les résultats ainsi que l'expérience de l'auteur montrent que, pour *Philips*, l'objectif de cette redevance a toujours été d'encourager les détenteurs de licence à mettre la norme I²C intégralement en œuvre. Cette norme incluait explicitement le filtre antiparasite de 50 ns qui fut jadis une pomme de discorde fréquente entre *Philips Semiconductor* et les fabricants tiers. Ce filtre (nous y reviendrons plus tard) pourrait être l'une des raisons qui poussèrent *Philips* à percevoir des droits de licence.

Level Annual Turnover	← Prev	1st Quarter Fees (Jan, Feb, Mar)	Next →	Annual Fee
Adopter > USD \$250m		\$8,000		\$8,000
Adopter¹ < USD \$250m		\$4,000		\$4,000
Contributor > USD \$250m		\$32,000		\$32,000
Contributor¹ < USD \$250m		\$16,000		\$16,000
Contributor² < USD \$10m		\$8,000		\$8,000
Board (and Promoter)		\$40,000		\$40,000

Figure 1. L'adhésion à l'alliance MIPI est incontestablement coûteuse. (Source : [3])

Toutefois, avec l'expiration des brevets de *Philips* tout est désormais ouvert. Malheureusement, l'interopérabilité des dispositifs I²C dans un système de bus I₃C n'apparaît que comme un procédé recommandable ou un vœu pieux. On ne peut pas se passer d'essais pratiques sur ces systèmes mixtes. Voyons d'abord qui ou quoi se cache derrière la norme I₃C.

I₃C et l'Alliance MIPI

L'Alliance MIPI [2] (*Mobile Industry Processor Interface Alliance*) est un comité de normalisation proposant diverses normes d'électronique générale et la norme I₃C. L'adhésion à l'Alliance est chère, comme le montre la liste des redevances (fig. 1) [3]. Pour stimuler une adoption maximale, l'Alliance MIPI offre une version de base entièrement

gratuite de la spécification I3C. Les fonctions avancées (seuls les membres de l'Alliance y ont accès) sont listées ici [4]. Vous pouvez y voir que l'abréviation I3C signifie *Improved Inter Integrated Circuits* (=I2C amélioré). Selon l'auteur, une exploitation commerciale d'I3C ne pose pas de « problème » à l'Alliance MIPI sauf si vous implémentez peu ou prou *vous-même* l'interface, par ex. dans un FPGA ou par *bit-banging* (c.-à-d. l'exécution d'un protocole série par logiciel). En revanche, si vous n'utilisez que des composants achetés (sans utiliser le logo ni le nom I3C dans votre publicité), vous ne devriez pas rencontrer de problèmes.

Adieu le drain ouvert

Dès le début, le bus I²C réserva des désagréments aux développeurs. Primo, il y a toujours la question de savoir où placer les résistances de polarisation (*pull-up*) pour décharger la capacité du bus. Secundo, généralement il n'est pas possible de déclencher des interruptions depuis un composant qui ne délivre pas le signal d'horloge. Avec l'I²C, pour qu'un circuit « de détection d'horloge » puisse déclencher des interruptions, il faut toujours une connexion en plus. S'il y a plusieurs capteurs sur le terrain, c'est au détriment de la capacité GPIO du circuit émetteur du signal d'horloge. Tertio, la vitesse reste relativement faible : selon Philips, le débit de données est de 3 Mbps avec une horloge à 3,4 MHz. La conséquence est que la faible réactivité du transmetteur entraîne une consommation élevée d'énergie, l'allongement de l'échange de données entre deux appareils retarder la mise en veille à faible consommation. Selon l'auteur, les histogrammes MIPI sont plausibles (fig. 2).

Quarto, les dispositifs I²C ont la capacité fatale de désactiver « pour toujours » d'autres dispositifs du bus. La figure 3 illustre ce fait tel qu'on peut l'observer dans un certain nombre de systèmes I²C.

Saluons les améliorations de l'I3C

Le 1^{er} changement d'I3C est le mode *push-pull* des broches SCL (toujours) et SDA (en général). La broche SDA passe en mode drain ouvert (*open drain*) sous certaines conditions. Pour éviter les problèmes de placement des résistances de *pull-up*, la norme spécifie

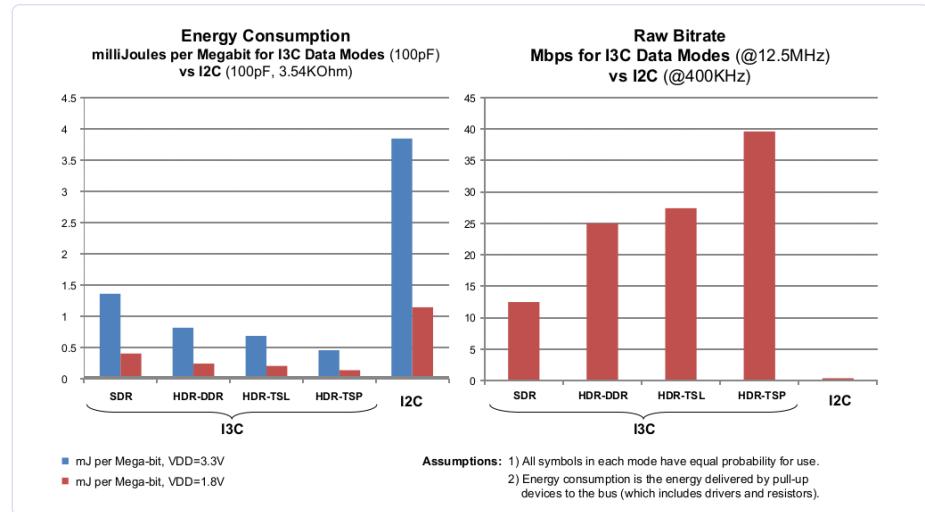


Figure 2. À l'instar des processeurs pour mobiles, la vitesse économise l'énergie. (Source : [8])

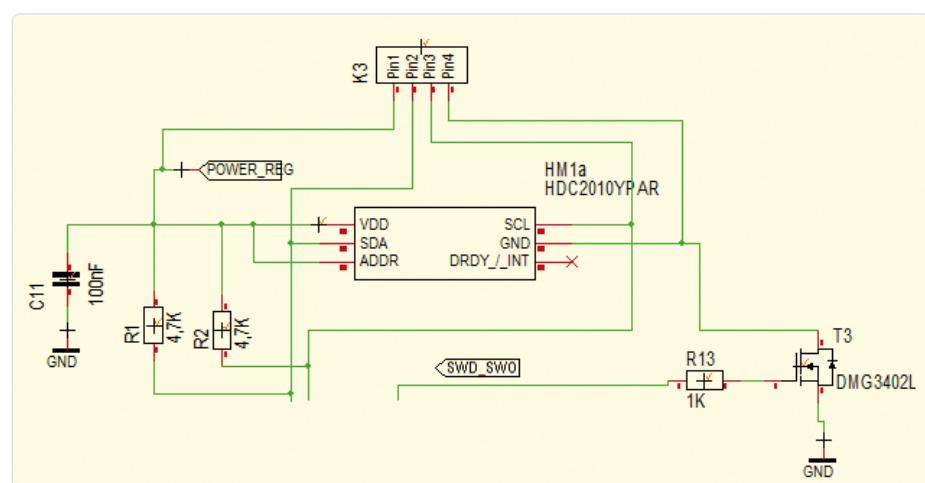


Figure 3. Le transistor T3 « d'arrêt d'urgence » permet l'extinction définitive de tous les capteurs connectés au bus pour remettre un dispositif « de blocage » dans son état par défaut.

qu'elles doivent être situées dans le circuit qui délivre le signal d'horloge.

Autre aspect important (laissant de côté des cas d'espèce) : la broche SCL est toujours et exclusivement commandée par le circuit générateur de signal d'horloge. Si le bus comporte plusieurs circuits susceptibles d'être générateurs du signal d'horloge, un seul est actif à un instant donné et responsable de la commutation de la broche SCL.

L'étirement de l'horloge par un dispositif esclave, possible sur un bus I²C classique, n'est pas pris en charge par la spécification I3C. Le fonctionnement de tels dispositifs n'est donc pas possible sur un bus mixte. Là encore, l'auteur ne connaît que très peu de cas où l'étirement de l'horloge se produit en pratique.

Cette simplification implique une simplification des broches d'E/S du MCU ou du circuit qui produit le signal d'horloge. À cause du filtre

de 50 ns mentionné ci-dessus, l'I²C exige des broches GPIO spécialisées. La norme MIPI n'exige que des broches GPIO normales, supportant un courant de 4 mA. Elle indique aussi explicitement que les filtres de 50 ns, souvent problématiques avec l'I²C, ne sont pas nécessaires.

Les avantages de la vitesse

Examinons maintenant les formes d'onde de l'I3C. Le filtre antiparasite mentionné ci-dessus n'est significatif avec l'I3C que si le signal SCL n'a pas un rapport cyclique de 50 % (comme c'est en général le cas avec l'I²C). Avec l'I3C, la période haute est définie comme étant inférieure à 45 ns (fig. 4). Par conséquent, un dispositif I²C conforme à la norme ne pourra rien faire de la forme d'onde SCL.

Après réception de cette sorte de transaction SCL « invisible », le dispositif « de génération d'horloge » peut également commuter sa

MIPI I3C Bus signal in SDR mode after dynamic address assignment

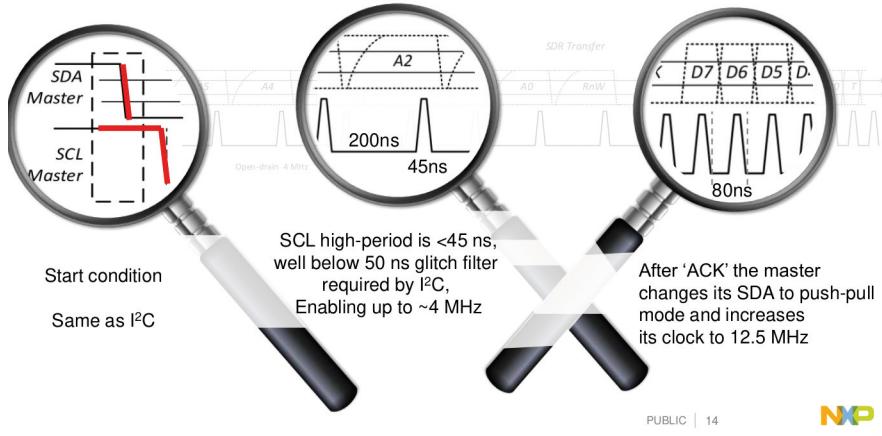


Figure 4. Les composants I²C considèrent ce front d'horloge comme « bas ». (Source : présentation NXP)

5 Pin Configuration and Functions

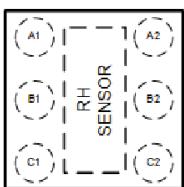


Figure 5-1. WLCS (DSBGA) 6-Pin YPA Top View

Table 5-1. Pin Functions

PIN		I/O TYPE ⁽¹⁾	DESCRIPTION
NAME	NO.		
VDD	A1	P	Positive Supply Voltage
ADDR	B1	I	Address select pin – hardwired to VDD or GND. GND: slave address: 1000000 VDD: slave address: 1000001
GND	C1	G	Ground
SDA	A2	I/O	Serial data line for I ² C, open-drain; requires a pullup resistor to VDD
SCL	B2	I	Serial clock line for I ² C, open-drain; requires a pullup resistor to VDD
DRDY / INT	C2	O	Data ready/Interrupt. Push-pull output

(1) P=Power, G=Ground, I=Input, O=Output

Figure 5. Le HDC2010 n'a qu'une broche d'adresse, donc impossible d'en mettre plus de deux sur un bus I²C. (Source : [9])

broche/ligne SDA en mode *push-pull* et augmenter la fréquence jusqu'à 12,5 MHz. Dans ce mode pleine vitesse, il est possible d'atteindre un débit de transmission allant jusqu'à 12,5 Mbps conformément à la norme MIPI I3C.

Des débits de transmission plus élevés peuvent être obtenus avec l'I3C dans l'un des trois modes *HDR* (*High Data Rate*), ils ne sont utilisables que dans deux cas. Le 1^{er} cas est un système de bus appelé *Pure I3C Bus*, composé exclusivement de composants I3C, et le 2^e est baptisé *Mixed Fast Bus* et peut inclure des composants I²C, mais seulement s'ils ont un filtre de 50 ns implémenté selon la spécification.

Les modes HDR sont lancés par une séquence spéciale d'impulsions SCL alors

que la ligne SDA est au repos. Le mode le plus simple est le *HDR-DDR* (*Double Data Rate*), qui peut atteindre une vitesse de 25 Mbps. À cet effet, chaque front du signal SCL devient un déclencheur valide de réception de données. Le mode *HDR-TSP* (*Ternary Symbol Pure*) permet de dépasser une vitesse de 30 Mbps. Il utilise les lignes SCL et SDA en combinaison pour la transmission des données. Le mode *HDR-TSL* (*Ternary Symbol Legacy-inclusive*) est un mode qui rend le TSP plus accommodant pour les bus mixtes.

Attribution automatique des adresses

Un autre inconvénient de l'I²C est l'attribution physique quasi obligatoire d'une adresse à chaque composant. Pour de nombreux

composants (par ex. le HDC2010 illustré à la **figure 5**), cela implique que l'espace d'adresse prévu par la norme ne peut pas être pleinement utilisé.

Au lieu d'un adressage sur 10 bits possible avec l'I²C, l'I3C permet une autoconfiguration dynamique. En outre, la fonction I3C *Hot Join* permet (comme l'USB) le *branchement à chaud* de dispositifs I3C sur le bus.

En arrière-plan, il existe aussi une procédure dite *d'arbitrage d'adresse* qui s'occupe d'attribuer des adresses aux dispositifs esclaves. À cette fin, chaque périphérique I3C se voit doté de trois paramètres : deux champs de huit bits d'information d'état, et un identifiant unique de 48 bits (*UID*). La spécification implique que chaque *UID* (appelé *Provisional ID*) est unique sur le bus. Au démarrage du bus I3C, le dispositif de génération d'horloge doit scruter tous les dispositifs du bus et leur attribuer des adresses dynamiques. À cet effet, il utilise une méthode similaire au processus d'arbitrage I²C : une adresse dynamique est d'abord attribuée au composant I3C ayant la valeur *UID* la plus faible. Le dispositif de génération d'horloge exécute autant de cycles d'affectation que nécessaire pour que chaque dispositif reçoive un ID temporaire.

Adoption de la norme par les fabricants

La liste des membres de l'Alliance MIPI [5] est un who's who des fondateurs, de *GigaDevice* à *Solomon Systech* en passant par *ST*. La quasi-totalité des entreprises concernées y figure. Cependant, en pratique si vous cherchez des composants disponibles, le choix est très limité. Chez *Mouser* par ex., on ne trouve actuellement que des composants tels que le *PI3CSW12* de *Dialog*. Il s'agit en général de circuits intégrés de multiplexage et de conditionnement de signaux, voir la **figure 6**. *Renesas* a, du moins pour l'instant, adopté une approche similaire avec des CI tels que l'*IMX3102*. En outre, la déclaration suivante est incluse dans l'annonce : « Les vitesses de 12,5 MHz de l'I3C dépassent considérablement les solutions actuelles et anciennes, comme l'I²C à 1 MHz et les commutateurs passifs analogiques rapides ». Si vous cherchez des MCU avec prise en charge de l'I3C, la liste est très courte : le *i.MX RT685* de NXP est le seul disponible. Il existe une note d'application (AN12797)

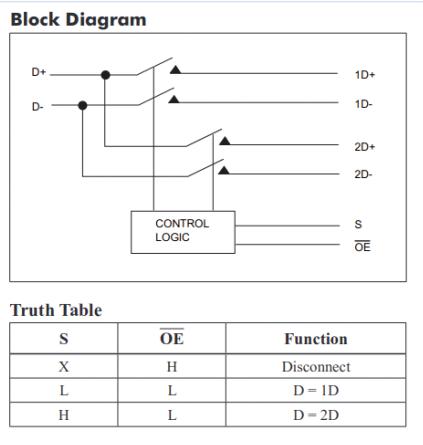


Figure 6. Le matériel I3C disponible à ce jour se limite avant tout au conditionnement de signal.
(Source : [10])

qui fournit un exemple de logiciel. NXP propose un noyau logiciel (gratuit dans certains cas) et une licence spéciale pour les utilisateurs non-membres de l'alliance MIPI [6]. Si vous êtes prêt à payer pour la propriété intellectuelle (PI) du noyau logiciel, des fournisseurs comme *Silvaco* ou *Arasan Chip Systems* proposent des modules VHDL I3C clés en main.

Ne cherchez pas d'écrans avec support I3C (cas d'utilisation souvent mentionné dans la norme), il n'y en a pas encore. Les seuls CI actuellement disponibles sont des accéléromètres (*BMI263* de *Bosch* et *LSM6DSO* de *ST*) et le capteur de pression *LPS22HH* de *Bosch*.

Pour le logiciel, c'est plus encourageant : une description relativement détaillée du pilote d'interface mis en œuvre dans le noyau *Linux* est disponible [7].

Combien de temps encore ?

Le débit plus élevé et les interruptions sans recours à des entrées GPIO spécifiques conduiront peut-être à l'adoption de l'I3C par l'ensemble des fondeurs, mais combien de temps faudra-t-il ? Comme, à ce jour presque aucun MCU n'est disponible avec les périphériques I3C appropriés, l'auteur pense que cela pourrait prendre du temps. ↗

210640-04 – VF : Yves Georges

Pourquoi pas maître et esclave ?

Très conservateur en matière de sémantique, l'auteur préfère les locutions *dispositif de génération d'horloge* et *dispositif de détection d'horloge*. L'expérience acquise en enseignant à des générations d'étudiants prouve que cette façon de penser induit une meilleure compréhension.

Combien de dispositifs ?

L'I3C limite davantage le nombre de dispositifs sur un seul bus que l'I²C, probablement en raison de la vitesse plus élevée et de la capacité plus faible des broches à fournir ou absorber du courant. Dans les versions précédentes de la norme, l'Alliance MIPI indiquait la prise en charge de onze dispositifs au plus par bus. Toutefois, cette restriction, résultant d'une simulation des caractéristiques électriques, ne figure plus dans la norme. Sur les 2⁷ (128) adresses dynamiques possibles en théorie, environ 90 sont disponibles (selon la configuration du bus).

Des questions, des commentaires ?

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



PRODUITS

➤ Livre en anglais, « Mastering the I²C Bus », Elektor 2011

Version papier : www.elektor.fr/15385

Version numérique : www.elektor.fr/16193

LIENS

- [1] Discussions sur l'I²C : www.verycomputer.com/31_b0b5a85949fcce82_1.htm
- [2] Page web MIPI I3C : www.mipi.org/specifications/i3c-sensor-specification
- [3] L'adhésion MIPI est coûteuse : www.mipi.org/join-mipi
- [4] Spécification de base de l'I3C : <https://resources.mipi.org/blog/mipi-alliance-delivers-new-i3c-basic-specification>
- [5] Liste des membres de l'Alliance MIPI : www.mipi.org/devcon//membership/all-member-directory
- [6] NXP soft core IP : <https://bit.ly/3qWAvD3>
- [7] API dans le noyau Linux : www.kernel.org/doc/html/latest/driver-api/i3c/protocol.html
- [8] Livre blanc MIPI I3C : <http://bit.ly/2gld6BL>
- [9] Fiche technique HDC2010 de Texas Instruments : www.ti.com/lit/gpn/hdc2010
- [10] Fiche technique PI3CSW12 de Diodes Inc. : www.diodes.com/assets/Datasheets/PI3CSW12.pdf

BlueRC :

télécommande IR avec smartphone et ESP32

adaptative et universelle

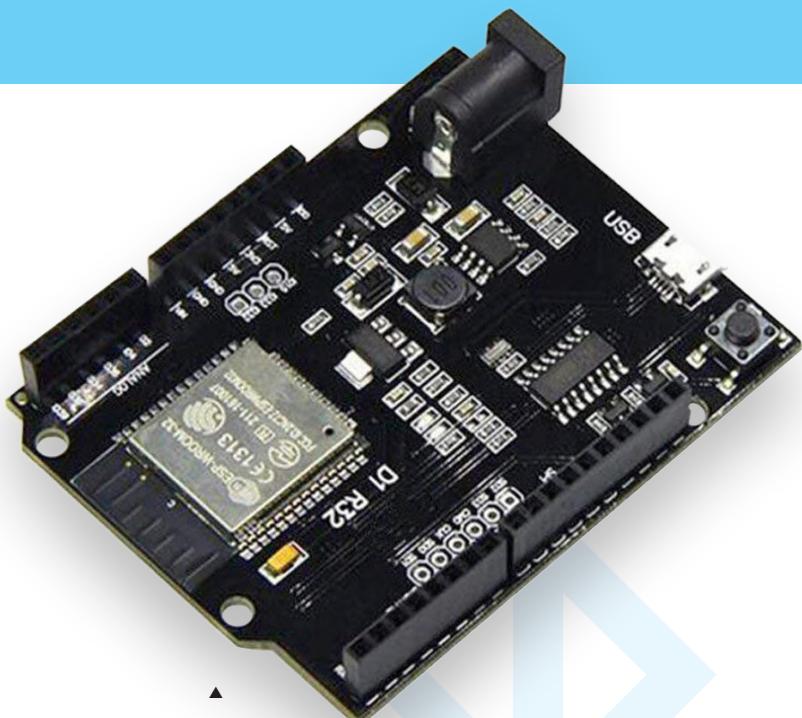


Figure 1. Carte de développement WeMos D1 R32 ESP32 Wi-Fi/Bluetooth.

Xin Wang (Allemagne)

Il fut un temps où les émetteurs infrarouges sur les téléphones portables étaient courants, mais ce n'est plus le cas. Ce projet propose une solution : utilisez votre téléphone portable avec une connectivité Bluetooth pour contrôler une carte ESP32 qui peut à la fois envoyer des commandes infrarouges et en apprendre de nouvelles.

Je souhaitais commander mes appareils à infrarouge avec mon téléphone. J'ai fait quelques recherches pour comparer plusieurs projets sur internet. La plupart sont basés sur des codes prédéfinis, ce qui signifie que les codes IR devaient être prédéfinis et stockés dans un tableau. Je voulais créer une télécommande à apprentissage qui ne nécessiterait aucune connaissance préalable de codes. Les codes doivent progressivement être mis à jour par le circuit en temps réel.



Figure 2. Interface utilisateur de l'application Android BlueRC V1.1.



Figure 3. Le shield Keyestudio Easy Module V1 pour Arduino. Source : flyrobo.in.

Matériel

Ce projet repose sur deux éléments principaux :

- Une carte à microcontrôleur (**figure 1**) avec un récepteur infrarouge connecté pour apprendre les codes existants d'autres télécommandes, et un émetteur infrarouge pour effectuer la tâche principale en envoyant des commandes à votre téléviseur, boîtier décodeur, etc. La carte stocke également les codes appris.
- Un téléphone Android pour exécuter une application, qui sert simplement d'interface utilisateur pour la carte à microcontrôleur. L'application vous offre une interface de télécommande (**figure 2**), qui vous permet d'appuyer sur les boutons dont vous avez besoin pour contrôler vos appareils infrarouges. Vous pouvez également lancer la séquence d'apprentissage en appuyant sur une touche de cette interface utilisateur, mais la procédure d'apprentissage elle-même est exécutée sur la carte à microcontrôleur.

J'ai choisi le Bluetooth pour la communication entre le téléphone et l'émetteur IR, pour sa faible consommation d'énergie et sa faible latence par rapport au wifi.

J'ai cherché une solution disponible sur le marché qui répondrait à toutes les exigences matérielles, mais sans succès. Donc, en ce qui concerne le microcontrôleur, j'ai choisi une carte ESP32, puisqu'elle dispose à la fois de Bluetooth et de wifi. La carte de développement D1 R32

de WeMos a une disposition des broches compatible avec Arduino UNO et fonctionne bien avec l'EDI Arduino. J'ai ajouté un *shield* à capteurs multiples (**figure 3**). Ce *shield* a une pléthore d'autres capteurs : un capteur d'humidité, un buzzer, un potentiomètre, une cellule photoélectrique, une LED RGB, et quelques boutons, entre autres — tous ces éléments offrent une grande polyvalence pour de futures idées de domotique. J'ai utilisé les capteurs pour afficher la température et l'humidité dans l'application Android, mais notre capteur intégré le plus important est le récepteur infrarouge !

Alors que le *shield* a un récepteur IR intégré, il ne dispose pas d'un émetteur, j'ai donc dû construire mon propre module de commande/émetteur rapide et pratique. Il faut seulement 3 composants - une douille à 3 broches, un transistor BC547 NPN, et une LED IR, montrés très simplement assemblés dans la **figure 4**. Après soudure,

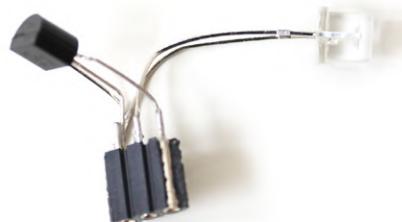
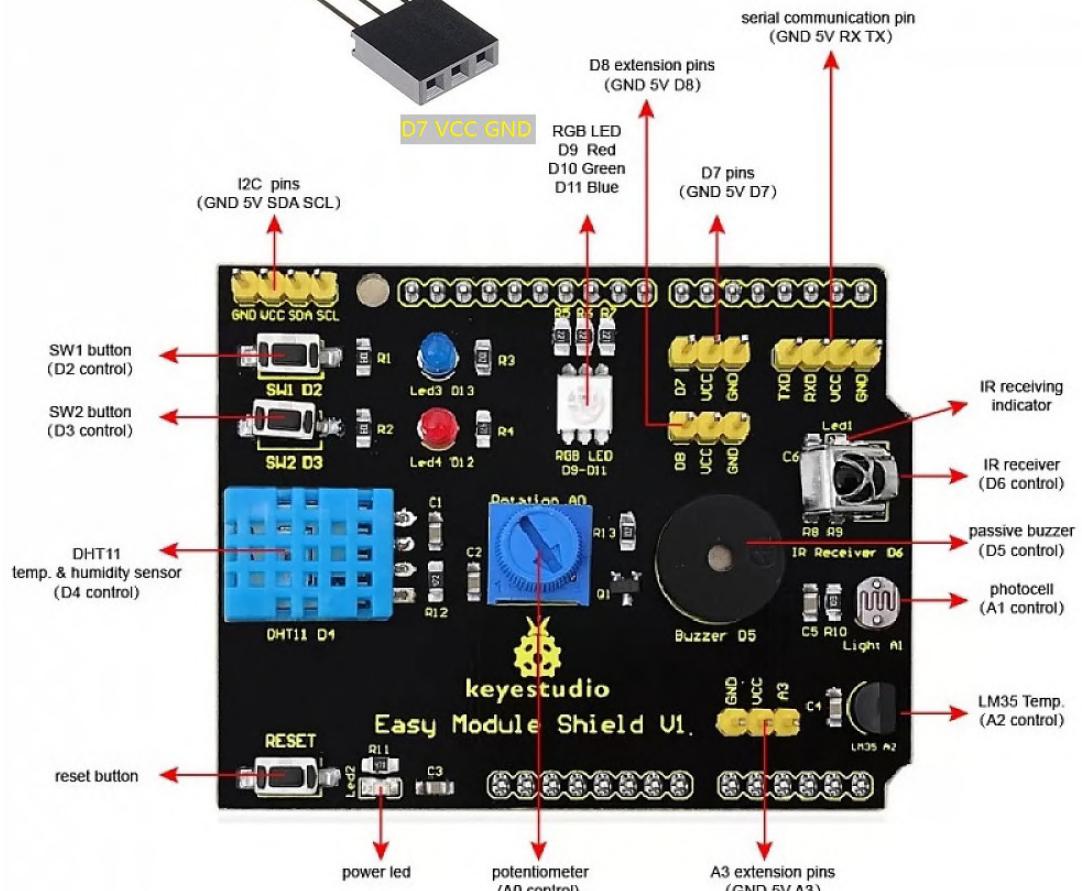


Figure 4. Mon module de commande/transmission infrarouge «maison».

▶
 Figure 5. Le shield et la façon dont j'y ai connecté mon module de commande/transmission IR.
 Source : flyrobo.in.

BC547 IR transmitter



j'ai branché le connecteur dans la broche du port D7 sur le *shield* du capteur, qui utilise la broche du port D14 comme sortie pour contrôler mon émetteur IR, alors que le récepteur IR du *shield* utilise la broche du port D27 comme entrée par défaut.

La **figure 5** présente le schéma de base.

Logiciel

Mon logiciel est relativement simple : l'ESP32 sert de serveur Bluetooth, et l'application Android sert de client. L'ESP32 attend simplement les messages Bluetooth de l'application. La communication Bluetooth entre les deux appareils est limitée, car les pressions sur les touches de la télécommande – ainsi que la commande pour commencer la séquence d'apprentissage (^) – de l'appli-

cation Android sont envoyées à l'ESP32 sous la forme de caractères uniques.

J'ai d'abord utilisé la bibliothèque IR standard pour Arduino. Elle fonctionnait bien, mais n'était pas totalement compatible avec l'ESP32, car ce dernier utilise un émetteur-récepteur de télécommande (RMT) propriétaire avec beaucoup de protocoles et de logique de modulation intégrés pour les signaux infrarouges. Après de nombreux essais et erreurs, j'ai finalement trouvé une bibliothèque IR entièrement fonctionnelle pour l'ESP32, *IRremoteESP8266*, par Ken Shirriff et al [1]. Merci à Ken. À l'origine, je voulais stocker les codes IR dans l'EEPROM, mais lorsque j'ai écrit mon micrologiciel, j'ai réalisé que l'ESP32 n'avait pas d'EEPROM. Cependant, il a la possibilité de sauvegarder des données de façon permanente dans



la mémoire flash grâce à la bibliothèque *Preferences.h* [5]. Comme ma télécommande d'origine avait un codage RC5, je n'ai implémenté que le schéma de codage RC5. Vous devriez peut-être modifier le croquis Arduino si votre télécommande utilise un autre protocole.

Programmation

Pour la carte WeMos, téléchargez le croquis Arduino *BlueRC* sur [2]. Il s'agit d'un seul fichier *.ino* que vous pouvez télécharger avec l'IDE Arduino.

Pour configurer l'application Android, suivez les étapes suivantes :

1. Sur votre appareil Android, téléchargez *BlueRC-app-V1.1.apk* depuis la page GitHub [3]. Vous devez configurer votre téléphone pour qu'il autorise le téléchargement d'applications à partir de sources externes (autres que Google Play Store).
2. Mettez sous tension la carte WeMos sur laquelle est installé le micrologiciel *BlueRC.ino*.
3. Lancez l'application Android. Si le Bluetooth n'est pas actif, l'application demande une autorisation ; appuyez sur *Accept*.
4. Tapez sur le « kebab menu » en haut à droite, puis sur *Connect BT Device*. S'il s'agit de votre premier appairage avec l'ESP32, tapez sur *Scan for devices* et attendez que *ESP32BlueRC-Shield* apparaisse sous *Other Available Devices*. Tapez dessus, puis confirmez l'appairage.
5. Dans la barre d'état en haut, vous devriez maintenant voir *connected:ESP32BlueRC-Shield*.

Une fois que vous avez apparié votre matériel, il n'est pas nécessaire d'effectuer à nouveau le processus de recherche – il suffit d'appuyer sur *ESP32BlueRC-Shield* sous *Paired Devices* à l'avenir.

Une fois l'application installée et fonctionnelle, vous pouvez maintenant suivre le processus d'apprentissage :

1. Appuyez sur la touche LEARN dans l'interface utilisateur, et l'application répondra par « *Learning begins, please press a button within 10 seconds...* » L'application Android envoie alors le caractère spécial « ^ » à l'ESP32, puis la LED bleue du *shield* s'allume.
2. Sur l'interface utilisateur, appuyez sur la touche que vous voulez indiquer à la carte ESP32. Le téléphone

envoie le caractère correspondant à l'ESP32, qui attend alors l'étape suivante.

3. Pointez votre ancienne télécommande infrarouge vers la carte ESP32 et appuyez sur le bouton correspondant de la télécommande. Le code IR de ce bouton sera automatiquement stocké dans la mémoire de l'ESP32.
4. Répétez les étapes 1 et 2 pour toutes les touches que vous souhaitez enregistrer.
5. Fermez l'application en appuyant sur *Menu → Exit program*.

Voilà, c'est fait. Vous pouvez maintenant contrôler vos appareils infrarouges à partir de votre appareil BlueRC, grâce à l'application sur votre téléphone Android.

Pour résoudre les problèmes liés à l'ESP32, vous pouvez l'éteindre et le rallumer.

Vous pouvez visionner une vidéo de démo [4] sur ma chaîne YouTube.

220103-04



Des questions, des commentaires ?

Contactez Elektor (redaction@elektor.fr).



Produits

- **Dogan and Ahmet Ibrahim, *The Official ESP32 Book* (PDF, en anglais, SKU 18330)**
<https://elektor.fr/18330>
- **Koen Vervloesem, *Getting Started with ESPHome* (PDF, en anglais, SKU 19739)**
<https://elektor.fr/19739>

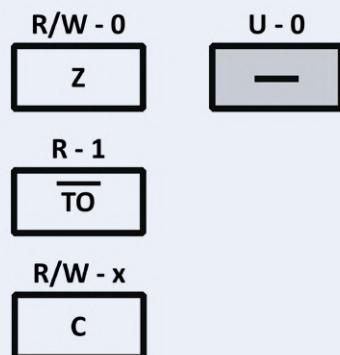


LIENS

- [1] Bibliothèque *IRremoteESP8266* : <https://www.arduino.cc/reference/en/libraries/irremoteesp8266/>
- [2] Le croquis Arduino BlueRC pour ESP32 : <https://elektor.link/BlueRCESP32>
- [3] Application Android BlueRC : <https://elektor.link/BlueRCApp>
- [4] Le système BlueRC - vidéo : <https://youtu.be/Kb-TdF84Oz4>
- [5] *ESP32 Save Data Permanently using Preferences Library*:
<https://randomnerdtutorials.com/esp32-save-data-permanently-preferences/>

la documentation des microcontrôleurs sans peine (2)

registres et schémas de principe



- R – readable
- W – writable
- U – unimplemented
- 1 – one after reset
- 0 – zero after reset
- x – unknown value after reset
- X – bit value inverted

Figure 1 : La description des bits des registres peut sembler quelque peu mystérieuse. (Source : Microchip Technology)

Stuart Cording (Elektor)

Que vous aimiez ou non la documentation des microcontrôleurs, vous aurez à négocier pour l'utiliser. Dans la première partie de cette série d'articles [1], nous avons examiné le contenu d'une fiche technique de microcontrôleur. Ici, nous allons plus loin en examinant comment les utilisateurs sont informés des fonctionnalités d'un appareil, en abordant tous les aspects, depuis les registres jusqu'aux schémas de principe des périphériques.

Fiche technique : premières pages

Maintenant que vous savez ce que contient une fiche technique, il est temps de commencer à examiner certains détails. Les fiches techniques semblent décrire les choses de manière quelque peu surprenante, et chaque fournisseur crée ses descriptions et ses schémas de différentes manières. Nous allons nous focaliser spécifiquement sur le microcontrôleur 8 bits PIC16F18877 de Microchip Technology [2]. Nous vous conseillons donc de télécharger la fiche technique si vous souhaitez suivre notre démarche.

Après avoir téléchargé et ouvert la fiche technique, nous avons sous les yeux les deux premières pages, qui contiennent une description succincte de ce microcontrôleur ainsi qu'une liste des principales caractéristiques et des périphériques inclus. Les fonctionnalités principales (*core features*) ne sont pas des périphériques autonomes ; il s'agit de capacités étroitement liées au noyau de traitement. On y trouve des fonctionnalités comme un circuit d'arrêt qui détecte le moment où la tension d'alimentation tombe au-dessous du niveau requis, mais aussi les temporiseurs « chien de garde »,

chargés de réinitialiser le microcontrôleur, activés par les systèmes de sécurité critiques.

La troisième page approfondit un peu plus le sujet en expliquant le nombre de chaque périphérique obtenu, et la quantité de mémoire RAM, de mémoire flash de programmation et d'EEPROM mise en œuvre. Viennent s'y ajouter les pages quatre à sept, qui contiennent des informations de base du conditionnement (boîtier) et expliquent les fonctionnalités affectées à chaque broche.

Les tableaux d'attribution des broches (Pin Allocation Tables) constituent un élément essentiel d'une fiche technique de microcontrôleur. Aujourd'hui, les microcontrôleurs ont davantage de fonctionnalités que de broches. Il est donc possible d'affecter une capacité (parfois même plusieurs) à chaque broche. Sur le composant abordé ici, la broche RA2 peut servir d'entrée de CA/N, de référence de tension analogique, d'entrée de comparateur, de sortie de CN/A ou de broche d'interruption sur changement. Pour obtenir une certaine flexibilité, vous pouvez également commuter une capacité spécifique entre une ou plusieurs broches. L'affectation des fonctionnalités aux broches peut donc s'avérer très complexe.

Comprendre les descriptions des registres

Les descriptions des registres se composent de deux parties : le nom du registre et les noms des bits (ou des groupes de bits) de ce

registre. La page 38 de la fiche technique décrit la fonctionnalité du registre STATUS, qui fait partie du processeur. Ce registre est composé de bits qui sont mis en œuvre et d'autres qui ne le sont pas.

Les bits mis en œuvre sont marqués d'une combinaison de mentions R et/ou W pour indiquer s'ils peuvent être lus ou écrits. Le trait d'union est généralement suivi d'un chiffre : « 0 » ou « 1 ». Ces chiffres indiquent la valeur de ce bit après réinitialisation de l'appareil. Certains bits peuvent être marqués d'un x, ce qui indique que leur valeur est inconnue. La valeur de certains bits peut dépendre de la raison pour laquelle le microcontrôleur est sorti de la réinitialisation. Si le microcontrôleur a simplement été mis sous tension, il s'est réactivé par le biais d'une réinitialisation lors de la mise sous tension (POR). Si l'alimentation électrique a provoqué une réinitialisation à la suite d'une panne de courant (BOR), cela peut également se traduire par une valeur différente dans un bit spécifique. Dans cet exemple, ces bits « dépendants de la réinitialisation » sont indiqués par la lettre q.

Les bits non mis en œuvre peuvent être un peu dangereux. Selon cette fiche technique, la lecture des trois bits supérieurs doit aboutir à un « 0 ». Cependant, il n'est pas évident de savoir ce qui survient si vous écrivez dans ces bits. La fiche technique peut comporter une recommandation générale. S'il n'y en a pas, vous devrez probablement vous assurer qu'une opération d'écriture dans ces registres

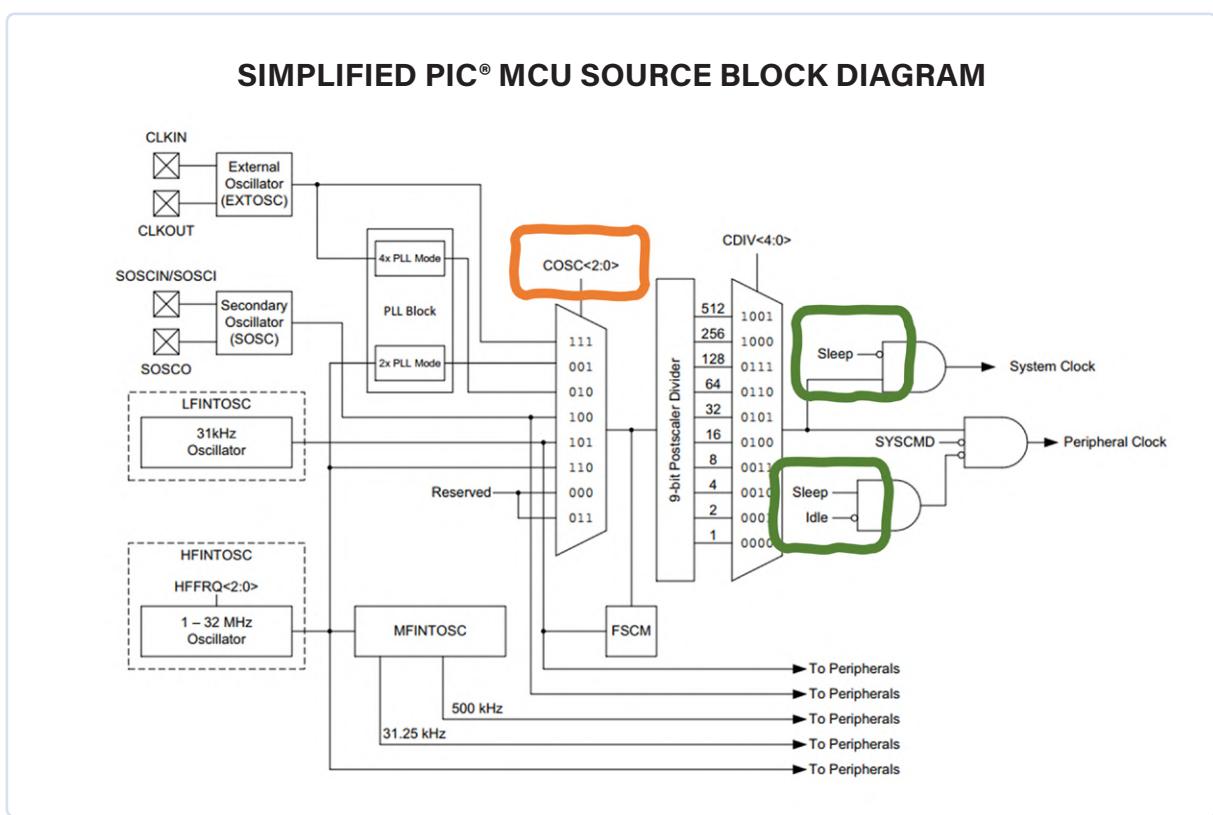


Figure 2. Schéma de principe d'un périphérique d'horloge, qui met en évidence le multiplexeur de sélection d'horloge et les commandes servant à désactiver la source d'horloge. (Source : Microchip Technology)



Bits non implémentés remis à zéro

```
myValue &= 0b00011111;      // clear 3 most-significant bits
                           // (unimplemented bits)
STATUS = myValue;          // write 'myValue' into STATUS register
```

garantit que les bits non mis en œuvre sont remis à zéro. En langage C/C++, cela peut se faire comme suit :

```
myValue &= 0b00011111; // clear 3 most-significant bits
                      // = (unimplemented bits)
STATUS = myValue;    // write 'myValue' into
                      // STATUS register
```

Remarque : il est judicieux d'insérer un commentaire expliquant la raison de cette démarche. Ainsi, personne n'essaiera ultérieurement d'« optimiser » la précaution prise !

La barre au-dessus du nom d'un bit indique que sa valeur est inversée. C'est le cas du bit « Time Out », TO. En cas de dépassement de délai, que l'on pourrait s'attendre à voir signalé par une réponse positive « 1 », la réponse sera en fait « 0 ». Souvent, ces bits inversés n'ont que peu de sens pour le programmeur ; vous devez simplement vous en accommoder et les traiter de manière appropriée dans votre code.

Parfois, un groupe de bits est nécessaire pour configurer quelque chose, par exemple un prédiviseur (ou prescaler) d'une source d'horloge ou la vitesse de transmission d'une interface série. Ces groupes de bits sont alors appelés **BIT_GROUP<X:Y>**, où X désigne le bit le plus significatif et Y le bit le moins significatif du groupe. Attention également : il arrive qu'un ou plusieurs bits appartenant à un même groupe soient répartis dans différentes zones d'un même registre, voire dans plusieurs registres !

Priorité à l'horloge

La partie la plus importante du microcontrôleur est son périphérique d'horloge. Il s'agit ici de définir la source d'horloge pour le

processeur et les périphériques intégrés. Il est donc judicieux de prendre le temps de revoir la structure du système, comme indiqué page 110. Normalement, ce périphérique est configuré au démarrage du code de l'application et reste inchangé ultérieurement. S'il est modifié par la suite, tous les périphériques peuvent être affectés, depuis la fréquence d'échantillonnage du CA/N jusqu'à la vitesse de transmission des interfaces série UART et CAN.

Dans cet exemple, nous avons des sources externes (oscillateurs à quartz ou à résonateur céramique) et des sources internes. Il est possible que ces sources internes ne soient pas suffisamment précises (en particulier avec les changements de température) pour servir de source fiable pour certaines interfaces, l'UART par exemple. La fiche technique le mentionne probablement, faute de quoi, vous devrez vérifier la précision dans la section relative aux caractéristiques électriques.

Dans le diagramme, nous pouvons voir qu'un groupe de trois bits, **COSC<2:0>**, souligné en orange et positionné dans un registre très probablement lié à l'oscillateur, peut servir à sélectionner différentes sources d'horloge disponibles à l'aide d'un multiplexeur. Pour des raisons indéterminées, seul le groupe de bits est nommé dans ces diagrammes et non le registre auquel ils appartiennent. Le meilleur moyen de savoir à quel registre appartient le groupe est de consulter la fiche technique au format PDF – il peut en effet se trouver dans un registre d'un autre périphérique ! Les diagrammes multiplexeurs sont fréquemment utilisés dans les schémas fonctionnels où les signaux peuvent être commutés.

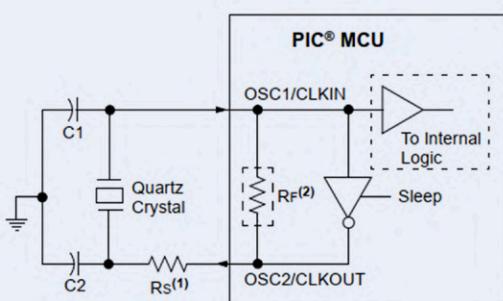
La sortie de ce bloc fournit l'horloge système (*System Clock*), probablement au processeur, et l'horloge périphérique (*Peripheral Clock*) pour les périphériques. La fiche technique peut faire référence à ces deux termes dans le reste du document, en particulier lorsqu'elle aborde les modes Basse consommation (*Low Power*) ou Veille (*Sleep*). D'après les sections marquées en vert, nous pouvons déjà déduire qu'il existe un mode Inactif (*Idle*) qui supprime l'horloge des périphériques uniquement, et un mode Veille (*Sleep*) qui supprime l'horloge des périphériques et du processeur.

Si un quartz ou un résonateur en céramique doit être utilisé, il est probable que des indications soient fournies sur la façon de construire le circuit et de réaliser le circuit imprimé. C'est également le cas ici (page 112), où deux condensateurs sont nécessaires, ainsi que, facultativement, une résistance en série. Quatre notes d'application relatives à la conception d'oscillateurs accompagnent ce schéma fonctionnel, ce qui souligne l'attention que nécessite la mise en œuvre des oscillateurs à quartz.

Au-delà des registres et des schémas de principe

Nous avons abordé les bases de la description des registres et la compréhension du schéma de principe du périphérique d'horloge.

Figure 3. Circuit recommandé pour un oscillateur à base de cristal de quartz. (Source : Microchip Technology)



Nous avons aussi jeté un coup d'œil à l'un des oscillateurs et à l'exemple de schéma de circuit pour son utilisation avec un cristal de quartz. Dans la prochaine partie de cette série, nous allons examiner un autre bloc périphérique clé qu'il faut maîtriser : le bloc de réinitialisation. Nous établirons également tout ce que la fiche technique n'aborde pas. Si vous recherchez des ressources pour accéder plus facilement à l'univers des microcontrôleurs, n'hésitez pas à consulter nos articles d'introduction [3] et nos livres. ↪

VF : Pascal Godart — 230101-04



Produits

- Livre en anglais : T. Hanna, *Microcontroller Basics with PIC*, Elektor 2020 (SKU 19188)
www.elektor.fr/19188
- Livre en anglais : A. Pratt, *Programming the Finite State Machine*, Elektor 2020 (SKU 19327)
www.elektor.fr/19327

Des questions, des commentaires ?

Envoyez un courrier à l'auteur (stuart.cording@elektor.com).

LIENS

- [1] Stuart Cording, « la documentation des microcontrôleurs sans peine (1) », elektormagazine.fr : <https://www.elektormagazine.fr/news/la-documentation-des-microcontroleurs-sans-peine-1ere-partie-la-structure-dune-fiche-technique>
- [2] microcontrôleur 8 bits PIC16F18877 : <https://microchip.com/wwwproducts/en/PIC16F18877>
- [3] Tam Hanna, « programmer les PIC – à petits pas », elektormagazine.fr : <https://www.elektormagazine.fr/articles/programmer-les-pic---petits-pas>

MagPi, le magazine officiel du Raspberry Pi



6 x MagPi :
Édition
imprimée



Accès aux
archives en
ligne du MagPi

12 mois
Plus de
100 projets
Le prix

54,95 €



COMMANDÉZ DÈS MAINTENANT AU
WWW.MAGPI.FR/ABO

MagPi
www.magpi.fr Magazine

automatisation des tests et des mesures

programmation de l'équipement d'essai pour qu'il fasse ce que vous voulez

Stuart Cording (Elektor)

Les essais peuvent être monotones et répétitifs, et la recherche de défauts est particulièrement difficile. Alors pourquoi ne pas utiliser les interfaces de communication des équipements de test et de mesure pour faciliter les choses ?

Nous explorons ici des outils qui peuvent facilement être contrôlés à l'aide de Python, sans avoir à investir dans des licences ou des logiciels coûteux.

Pour la plupart des développeurs, les équipements de test et de mesure sont posés sur l'établi, attendant d'effectuer une tâche. Mais si vous retournez la plupart de ces appareils, vous trouverez souvent une interface de communication à l'arrière. Associée au logiciel approprié, elle permet de contrôler les mesures et de rassembler les résultats en vue d'une analyse ultérieure. Cela peut s'avérer très utile pour rechercher des événements et des défaillances sporadiques ou pour tester une application en fonction de divers paramètres du système. C'est également de cette manière que les tests de fin de ligne de production et le regroupement des composants peuvent être automatisés.

Contrôle à distance des blocs d'alimentation des laboratoires

Un bon point de départ est l'alimentation électrique. La plupart des applications nécessitent un cycle d'alimentation pour effectuer une réinitialisation. Dans le cadre de tests plus avancés, l'application cible sera exploitée aux extrêmes de la spécification de l'alimentation d'entrée autorisée, et il peut être nécessaire d'effectuer des tests de surtension. Cela est courant dans l'automobile, où des impulsions allant jusqu'à 87 V doivent être supportées

pendant 400 ms (ISO 7637-2). Un autre mode de défaillance courant lié aux conditions d'alimentation est une tension qui augmente ou diminue lentement. Dans de telles conditions, les circuits tombent souvent dans une condition de *brown-out* dont ils ne peuvent pas se remettre. Enfin, pour les systèmes de test automatisés, il est utile de disposer d'un équipement qui peut être correctement configuré à la fois pour la tension et le courant maximal, en particulier lorsque la même installation est utilisée pour plusieurs produits différents. Les prix des alimentations de laboratoire ont considérablement baissé ces dernières années. Pour compenser l'afflux d'équipements bon marché, les marques plus connues élargissent généralement leur gamme de caractéristiques, car elles ne veulent ou ne peuvent pas rivaliser uniquement sur le plan des prix. On ne sait pas si c'est le cas d'Aim and Thurlby Thandar Instruments (Aim-TTi) à Cambridge, au Royaume-Uni, mais l'entreprise tient sa promesse d'offrir un « meilleur rapport qualité-prix ». Ils proposent une gamme étendue d'alimentations CC, mais leur série EL-R d'entrée de gamme (**figure 1**) vaut la peine d'être examinée de plus près, surtout si l'on caresse l'idée d'une première installation de test automatisée.



Figure 1. Deux des alimentations CC de la série EL-R sont dotées d'une interface série (RS-232/COM virtuel via USB) pour faciliter l'automatisation des tests. (Source : Aim-TTi)

Ces alimentations utilisent une régulation linéaire à faible bruit pour fournir des sorties simples, doubles et triples. Elles sont dépourvues de ventilateur, s'appuyant sur un refroidissement par convection, et leur puissance varie de 30 W à 130 W. Dotés d'un ou deux écrans LED rouges et de commandes analogiques, certains modèles comprennent également des bornes de détection à distance. Deux modèles intéressants sont le EL302P à sortie unique, avec RS-232, et le EL302P-USB, avec USB, tous deux des unités de 60 W avec des sorties de 0 V à 30 V et 0 A à 2 A.

Les unités sont livrées avec des pilotes logiciels pour l'interface. En outre, l'utilitaire « PSU Sequencer » est fourni sur le site web du fournisseur [1], permettant aux paramètres de tension/courant configurés par l'utilisateur d'être échelonnés manuellement ou automatiquement (**figure 2**). En outre, des séquences préparées à l'avance peuvent être importées à partir d'une feuille de calcul.

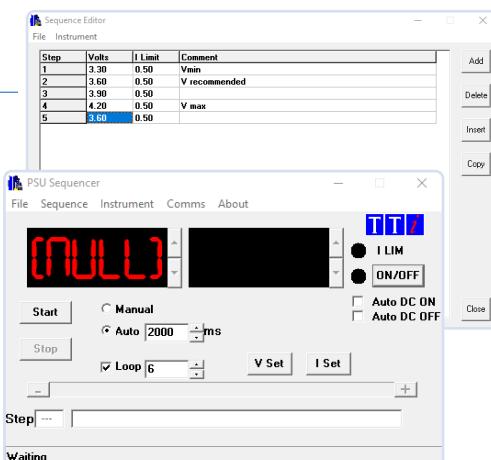


Figure 2. Le logiciel PSU Sequencer d'Aim-TTi permet d'effectuer des cycles simples et répétitifs de réglages de tension et de courant. (Source : Aim-TTi)

Python pour le contrôle de la puissance

Le développement de votre propre logiciel de contrôle, adapté à vos besoins de test, est également simple. L'interface RS-232 matérielle fonctionne entre 600 et 9 600 bauds, et l'interface USB apparaît comme un port série virtuel. Les commandes sont toutes répertoriées dans le manuel d'instructions [2] et, avec un peu d'organisation, il est assez simple de développer une bibliothèque qui transforme toutes les commandes en fonctions ou méthodes clairement nommées. Une option consiste à utiliser un Arduino et un émetteur-récepteur RS-232 pour contrôler l'alimentation électrique. Cette solution présente l'avantage de permettre l'intégration d'autres fonctions de test, telles que la commutation de signaux à l'aide de relais ou la capture de signaux analogiques et numériques.

Alternativement, Python fournit le module pySerial [3]. Avec un script simple (**exemple 1**), l'interface de commande peut être implémentée comme un module Python, et le contrôle automatisé peut être mis en œuvre. En lisant la documentation, il est également possible d'implémenter un pont TCP/IP vers série [4]. Cette fonction est définie dans le mémo expérimental RFC2217 [5] et permet à un PC distant de configurer une interface série et de mettre en œuvre la communication.

Détermination de la portée des signaux

Les oscilloscopes peuvent également être contrôlés à distance. Grâce à leur large éventail de capacités, allant de la capture de signaux analogiques et numériques aux FFT, ils peuvent être utilisés pour divers tests automatisés. Par exemple, certaines défaillances sont difficiles à détecter en raison d'une série complexe d'événements qui doivent se produire, dans une séquence spécifique,

Exemple 1 : script Python simple pour acquérir la chaîne d'identification de l'unité PSU à l'aide de pySerial

```
import serial
ser = serial.Serial('/dev/ttyUSB0', 9600, timeout=0)

print(ser.name)
ser.write(b'*IDN?')
response = ser.readline()
print(response)
ser.close()
```

pour les déclencher. Une fois que vous avez déterminé comment déclencher la défaillance, l'étape suivante consiste à configurer l'oscilloscope pour capturer les signaux pertinents qui aideront à déterminer la cause ultime.

Les cartes telles qu'Arduino et Raspberry Pi sont idéales pour le développement rapide d'un ensemble de tests qui peut déclencher la défaillance de manière répétée, en utilisant des sorties analogiques et des signaux numériques, parfois complétés par un relais ou un MOSFET si nécessaire. Ils peuvent également fournir des signaux de déclenchement synchronisés avec précision pour un oscilloscope afin de s'assurer que la bonne partie des signaux pertinents est sauvegardée pour l'analyse.

De nombreux oscilloscopes proposent des interfaces USB et LAN, mais certains n'offrent qu'un support pour un logiciel propriétaire ou un accès à un serveur web pour la configuration. Cependant, il existe une spécification de classe USB pour le test et la mesure,

connue sous le nom d'USBTMC [6]. À l'instar des dispositifs de stockage, tels que les clés USB, et des dispositifs d'interface homme-machine (IHM), tels que votre souris et votre clavier, cette classe fournit des commandes prédéfinies adaptées à l'interface avec les équipements de test et de mesure.

Des appareils tels que le B&K Precision 2567B [7], un oscilloscope à signaux mixtes (MSO) de 200 MHz et 2-GSa/s, prennent en charge cette interface. Avec quatre voies analogiques, un port numérique à 16 voies, un générateur de formes d'ondes arbitraires intégré de 50 MHz et des déclencheurs avancés, il est simple à configurer à l'aide de son grand écran tactile capacitif de 10,1 pouces (**figure 3**). Mais c'est tout aussi facile via l'USB.

Configuration à distance

Là encore, Python est le langage de programmation préféré grâce au projet USBTMC [8] d'Alex Forencich, qui est hébergé sur GitHub [9].



Figure 3. Grâce à une interface USB compatible USBTMC, les MSO de la série 2560B de B&K Precision sont facilement automatisés à l'aide de Python. (Source : B&K Precision)

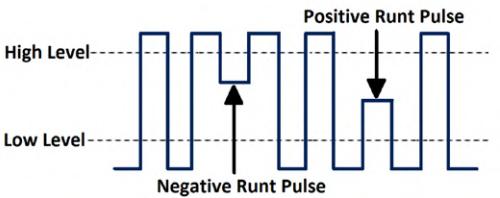


Figure 4. Le runt trigger est un mécanisme astucieux qui permet de détecter les impulsions qui ne montent ou ne descendent pas complètement. (Source : B&K Precision)

Le module fonctionne sous Linux, mais peut nécessiter de modifier les permissions de manière appropriée. Sous Windows, *PyUSB* [10] et *Libusb* [11] doivent être installés au préalable.

Les premières étapes sont relativement simples. Avant de commencer, il faut connaître les identités du vendeur (VID) et du produit (PID), deux valeurs de référence spécifiques à l'USB qui permettent d'identifier un produit USB. Sous Linux, il est facile d'obtenir ces informations à partir de la ligne de commande en utilisant `lsusb` une fois que le périphérique est connecté. La liste qui en résulte fournit les détails nécessaires. Sous Windows, ces valeurs peuvent être déterminées via le gestionnaire de périphériques et les propriétés du périphérique. Dans les deux cas, les valeurs fournies sont des valeurs hexadécimales de 16 bits.

Il ne reste plus qu'à importer le module `usbtmc` dans votre code Python et à utiliser l'interface de programmation disponible (API). Cela revient à imprimer du texte à l'écran et à évaluer les données saisies au clavier. L'USBTMC n'est en fait qu'une enveloppe permettant de communiquer avec des équipements de test et de mesure performants. Les commandes de contrôle sont des chaînes ASCII avec diverses options, comme décrit dans le manuel de programmation de l'oscil-

loscope [12]. Si elles sont disponibles, les chaînes de ressources VISA [13] peuvent également être utilisées.

Le VID et le PID sont propres au produit, et non à l'unité individuelle. Ainsi, si deux ou plusieurs appareils identiques sont connectés, ils peuvent être adressés indépendamment à l'aide d'un troisième paramètre, leur numéro de série. Celui-ci figure généralement sur une étiquette ou peut être obtenu à l'aide de la commande `*IDN?` (**exemple 2**). De nos jours, il n'y a pas grand-chose qui différencie les oscilloscopes au-delà de la bande passante, de la fréquence d'échantillonnage et de la quantité de mémoire. Mais de temps en temps, une petite caractéristique intéressante apparaît. La série B&K dispose d'une fonction déclenchement *runt* (« avorton ») (**figure 4**) qui se déclenche lorsqu'un signal franchit un seuil (par exemple, un niveau élevé) mais pas l'autre (par exemple, un niveau bas). En matière d'analyse des défaillances, tout est bon à prendre.

Équipement de test sans tête

Red Pitaya a innové dans le domaine des équipements de test, en remettant en question la nécessité pour un outil d'avoir son propre écran. Grâce à un puissant FPGA et à une interface Ethernet, votre PC ou votre ordinateur portable constitue l'interface utilisateur.



Figure 5. Le Moku:Lab, de la taille d'une petite assiette, est constitué de 12 instruments en un seul, et peut être configuré à l'aide d'une API Python. (Source : Liquid Instruments)

Liquid Instruments a adopté la même approche avec son Moku:Lab [14], un laboratoire de la taille et de la forme d'une assiette. Il offre une gamme d'entrées et de sorties analogiques et est conçu comme un laboratoire dans une boîte.

Quatre connecteurs BNC sont encastrés dans la face avant (**figure 5**). La paire de droite fournit les sorties analogiques, supportant un taux d'échantillonnage de 1 GSa/s par canal avec une résolution de 16 bits et une bande passante (-3 dB) de >300 MHz. La paire de gauche correspond aux entrées analogiques, avec une bande passante (-3 dB) de 200 MHz en 50 Ω et un taux d'échantillonnage de 500 MSa/s par voie à une résolution de 12 bits. La base de temps interne offre une précision meilleure que 500 ppb. Une entrée de déclenchement est également fournie, ainsi que des connecteurs permettant la synchronisation de plusieurs unités. La connectivité filaire est assurée par un port Ethernet et une interface USB, et un second port d'alimentation USB est disponible pour charger une tablette. Enfin, il y a un emplacement pour carte SD et une entrée d'alimentation en courant continu.

Bien que l'appareil dispose d'interfaces filaires, il est réellement conçu pour être utilisé en conjonction avec un iPad en Wi-Fi (802.11 b/g/n) et l'application correspondante. Douze instruments peuvent être sélectionnés via l'interface utilisateur, allant du familier, comme l'oscilloscope et l'analyseur de spectre, à l'éclectique, comme le contrôleur PID et la boîte de verrouillage laser. L'outil fonctionne également comme un enregistreur de données. La taille de votre carte SD est la limite pour le stockage des données jusqu'à une acquisition de 100 kSa/s, tandis que la capacité de la mémoire interne la définit pour des taux allant jusqu'à 1 MSa/s.

Bien entendu, comme tous les outils modernes, le Moku:Lab propose une API Python et un support MATLAB et LabVIEW.



Exemple 2 : utilisation de USBTMC dans Python pour contrôler un oscilloscope B&K Precision

```
import usbtmc
instr = usbtmc.Instrument(<VID>, <PID>, 'ABC123456')
# or, without serial number
instr = usbtmc.Instrument(<VID>, <PID>)
print(instr.ask("*IDN?"))
# returns 'BK Precision,2567B-MSO,ABC123456,5.0.1.3.9R3'
```

Appuyé par de nombreux exemples (**exemple 3**), le dispositif peut aussi être rapidement intégré dans un harnais de test automatisé. Il serait bien adapté aux tests de masse des dispositifs RF et au tri de composants.

Gagner du temps, améliorer la précision

Les équipements d'essai sont les yeux de l'ingénieur, car ils montrent ce qui se passe dans les systèmes complexes. Mais ils ont leurs limites. Par définition, les événements sporadiques sont difficiles à repérer, et une solution ne peut être proposée que pour une cause. Les interfaces de programmation supportées par les API sur les équipements de test et de mesure sont plus courantes aujourd'hui, allant du plus simple au plus raffiné. Et, de plus en plus, c'est le langage Python, facile à apprendre, qui devient le langage de programmation de prédilection. Si votre problème de mesure devient difficile à déclencher, nécessite des changements consécutifs pour compléter une seule série de tests, ou est simplement ennuyeux et répétitif (conduisant à des erreurs de l'opérateur), l'automatisation n'est pas aussi complexe que vous l'avez peut-être pensé. ↗

VF : Maxime Valens — 230046-04

Des questions, des commentaires ?

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



Exemple 3 : création d'une forme d'onde arbitraire sur le Moku:Lab en Python

```
"""pymoku example: Arbitrary waveform generator
(c) 2019 Liquid Instruments Pty. Ltd.
(shortened version for Elektor)
"""

from pymoku import Moku
from pymoku.instruments import ArbitraryWaveGen
import numpy as np

# generate a signal the the Arb Waveform Gen should generate on the output
t = np.linspace(0, 1, 100) # Evaluate our waveform at 100 points
# Simple square wave (can also use scipy.signal)
sq_wave = np.array([-1.0 if x < 0.5 else 1.0 for x in t])
# More interesting waveform. Note that we have to normalize this waveform
# to the range [-1, 1]
not_sq = np.zeros(len(t))
for h in np.arange(1, 15, 2):
    not_sq += (4 / (np.pi * h)) * np.cos(2 * np.pi * h * t)
not_sq = not_sq / max(not_sq)

# Connect to your Moku by its device name
m = Moku.get_by_name('Moku')
# Prepare the ArbitraryWaveGen instrument
i = m.deploy_or_connect(ArbitraryWaveGen)
try:
    # Load the waveforms to the device.
    i.write_lut(1, not_sq)
    i.write_lut(2, sq_wave)
    # Configure on-device linear interpolation
    i.gen_waveform(1, period=1e-6, amplitude=1, interpolation=True)
    i.gen_waveform(2, period=1e-6, amplitude=2, interpolation=False)
finally:
    m.close()
```

LIENS

- [1] PSU Sequencer, Aim-TTi: <http://bit.ly/40jGiCQ>
- [2] EL-R Series DC Power Supplies, Aim-TTi: <http://bit.ly/3kWZmXk>
- [3] Module pySerial: <http://bit.ly/3YgSJgZ>
- [4] Exemple de pont TCP/IP pySerial: <http://bit.ly/3kTDKem>
- [5] G. Clark, « RFC 2217 Telnet Com Port Control Option », Cisco Systems, Inc, octobre 1997: <http://bit.ly/3jcQGM4>
- [6] « Universal Serial Bus Test and Measurement Class Specification (USBTMC) », USB Implementers Forum, Inc. avril 2003: <http://bit.ly/3YbJY7L>
- [7] Modèle 2567B-MSO, B&K Precision: <http://bit.ly/3kN3rNy>
- [8] A. Forencich, « Python USBTMC », juillet 2014: <http://bit.ly/3HMLeZN>
- [9] A. Forencich, python-usbtmc, Projet GitHub: <http://bit.ly/3wK4i4r>
- [10] Logiciel PyUSB: <http://bit.ly/3WRFW3l>
- [11] Logiciel libusb: <http://bit.ly/3wLEvIY>
- [12] « Programming Manual 2560B Series », B&K Precision, septembre 2022: <http://bit.ly/3WRDYjs>
- [13] « VISA Resource Syntax and Examples », National Instruments Corp., mai 2022: <http://bit.ly/3XK3okh>
- [14] Moku:Lab, Liquid Instruments: <http://bit.ly/3WT6bXb>

Les solutions de test et de mesure sont importantes et largement utilisées par les ingénieurs professionnels, les étudiants et les fabricants. Nous présentons ici des données informatives sur des sujets liés aux tests et aux mesures, notamment les tests de logiciels, la sécurité des systèmes, le marché des MEMS et le marché mondial des capteurs industriels.

Continuez à tester ce logiciel

Vous travaillez dans le secteur des tests de logiciels ? Si c'est le cas, félicitations : vous travaillez dans un secteur en bonne santé qui pèse plus de 40 milliards de dollars. Ce n'est pas mal du tout ! Selon un récent rapport de Global Market Insights, le marché des tests de logiciels devrait croître de 6 % d'ici à 2030, ce qui signifie que vous aurez probablement des opportunités d'emploi décentes (et peut-être la sécurité de l'emploi) dans les années à venir. [1]



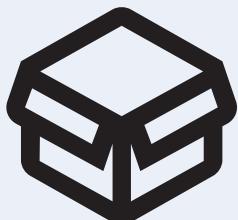
Les sociétés à surveiller

- Accenture
- Atos SE
- Amdocs
- Cognizant
- IBM
- Infosys
- Keysight (Eggplant)

(Source : Global Market Insights)

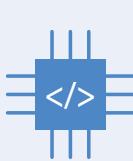
Qu'y a-t-il dans un test ?

Black Box



- Functional test: test app functionality and behavior.
- Pro programming insights are not required.
- Managed by testers.
- Code access is not required.

White Box



- Structural test: test app infrastructure.
- Pro programming insights are required.
- Managed by developers.
- Code access is required.

Besoin de tester un logiciel ? Vous voulez déboguer une application ? La sécurité est-elle importante pour vous ? Choisissez la boîte noire ou la boîte blanche ! Avant de choisir blanc ou noir [2] et de commencer à tester, pensez à consulter les ressources utiles suivantes !

RESSOURCES :

- S. Cording, « Déboguer des microcontrôleurs sans débogueur », 01 février 2023. <https://www.elektormagazine.fr/articles/deboguer-des-microcontroleurs-sans-debogueur>
- S. Cording, « Quelques ficelles pour le débogage des µC », 21 juillet 2020. <https://www.elektormagazine.fr/news/fr-the-full-gamut-of-microcontroller-debugging-techniques>
- M. Horkan, "A Black-Box Approach to Embedded Systems Vulnerability Assessment" (Elektor Business 4/2018). <https://elektormagazine.com/blackbox2018>
- L. Labbe, « Débogage par traçage sans fil », 26 septembre, 2022. <https://www.elektormagazine.fr/articles/procede-sans-fil-de-debogage-de-micrologiciels>
- C. Valens, « Sécuriser nos communications – entretien avec Luka Matic » 7 novembre, 2019. <https://www.elektormagazine.fr/news/fr-secure-communications-an-interview-with-luka-matic>

Le marché des MEMS

Le marché des systèmes micro électromécaniques (MEMS) est en pleine croissance, et plusieurs cabinets d'études indiquent que le marché mondial devrait connaître un TCAC de 8 % à 11 % d'ici 2031, selon la société Verified Market Research [3] [4]. En tant que dispositifs dotés de fonctions électriques et mécaniques, les MEMS ont d'innombrables applications dans un large éventail d'industries, notamment les équipements de mesure industriels, l'automobile, l'électronique grand public et l'aérospatiale. Ces dispositifs de très petite taille sont utilisés pour la détection de la pression, le traitement des signaux, la transmission de données et bien d'autres choses encore. « Le segment de l'électronique grand public a acquis la plus grande part en 2021 et devrait croître à un taux de croissance annuel moyen significatif de 2022 à 2031 pour la taille du marché des systèmes microélectromécaniques (MEMS) », a rapporté Allied Market Research en septembre 2022 [5]. Si vous êtes fasciné par les technologies MEMS, gardez un œil sur la façon dont les puces MEMS sont utilisées dans le domaine médical pour les outils micro-chirurgicaux, les implants et les micro-aiguilles pour l'administration de médicaments.

Marché des MEMS =

+ 11 %
d'ici 2031

Types

- > Actionneur
- > Capteur

MEMS
Market

Acteurs clés

- > Analog Devices
- > Broadcom
- > NXP
- > STMicroelectronics

Applications

- > Secteur automobile
- > L'électronique grand public
- > Domaine médical
- > Technologie de l'information

Localisation/taille du marché

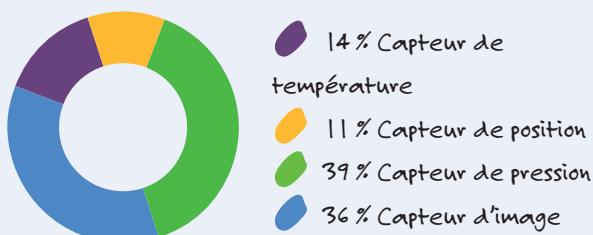
- > #1 Asie-Pacifique
- > #2 Amérique du Nord
- > #3 Europe

Comprendre les capteurs industriels

Le marché mondial des capteurs industriels devrait atteindre plus de 33 milliards de dollars d'ici la fin 2027, selon Fortune Business Insights [6]. Utilisés dans une grande variété d'applications de l'industrie 4.0, ces capteurs sont indispensables à des fins de

mesure et de test. Les capteurs industriels comprennent les capteurs de température, les capteurs de débit, les capteurs de gaz, les capteurs de mouvement de position.

Principaux capteurs par part de marché en 2021

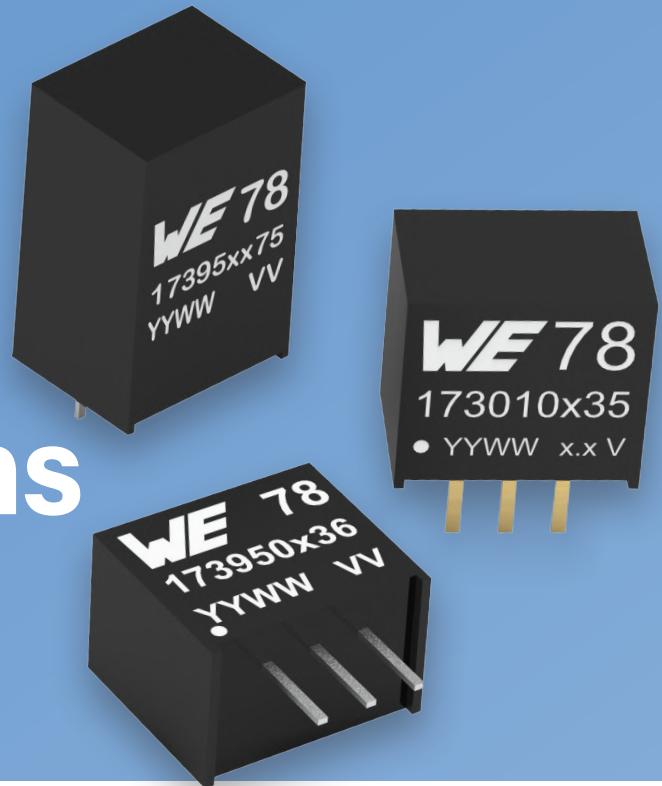


LIENS

- [1] Global Market Insights, "Software Testing Market Report," 2022: <https://www.elektormagazine.com/GMI-testing-software>
- [2] L. Nguyen, "Key Differences and Similarities Between Black Box and White Box Software Testing," Orient Software, 2021: <https://www.orientsoftware.com/blog/black-box-and-white-box-software-testing/>
- [3] Verified Market Research, "Microelectromechanical System (MEMS) Market Size And Forecast," 2023: <https://www.elektormagazine.com/vmr-mems-market>
- [4] Markets and Markets, "Micro-Electro-Mechanical System (MEMS) Market," 2023: <https://www.elektormagazine.com/mandm-mems-market>
- [5] Allied Market Research, "Microelectromechanical System (MEMS) Market," 2022: <https://www.elektormagazine.com/amr-mems-2022>
- [6] Fortune Business Insights, "Industrial Sensors Market Size, Share & COVID-19 Impact Analysis" 2019: <https://www.elektormagazine.com/fortune-sensors>
- [7] Markets and Markets, "Top 10 Sensors Market by Type (2021-2026)," 2021: <https://www.elektormagazine.com/markets-top-10-sensors>

immunité contre les surtensions

protection pour modules de puissance CC/CC non isolés



Timur Uludag (Würth Elektronik)

De par leur étendue, les infrastructures électriques des environnements industriels provoquent souvent des surtensions transitoires. Plusieurs paramètres sont à prendre en compte lorsqu'on cherche à mettre au point un filtre efficace contre ces phénomènes.

De nos jours, un grand nombre d'applications industrielles reposent sur des niveaux logiques d'entrée de 5 V_{CC} ou moins (**figure 1**). Le système de distribution électrique alimentant ces applications est généralement composé d'un bus de 24 V_{CC} et d'un convertisseur CC/CC à découpage pour en abaisser la tension au niveau logique inférieur souhaité.

La **figure 1** schématisé la structure électrique d'une usine. Les différentes parties sont alimentées par un bus CC. Chaque charge électrique séparée est connectée à un sous-circuit de distribution de 24 V. Des modules de puissance non isolés sont implantés pour délivrer la tension de service des sous-systèmes.

Origine des transitoires

Dans le cas d'un système électrique, un transitoire est un bref écart de la tension nominale hors de la plage

de tolérance admissible. Les transitoires ont généralement un effet destructeur.

Sur un bus CC, les transitoires peuvent avoir plusieurs origines. Un pic de tension peut être causé par la foudre (figure 1, **partie A**) ou être produit par le système lui-même (figure 1, **partie B**). En environnement industriel, le bus CC de 24 V est typiquement spécifié pour opérer entre 19,2 V et 30 V. Les transitoires peuvent alors prendre différentes formes. Si par exemple la ligne d'alimentation de 24 V est installée parallèlement à la ligne de commande d'un variateur de vitesse, les impulsions seront couplées de manière capacitive et la tension de 24 V reproduira les schémas d'impulsions du variateur et oscillera en conséquence.

Une protection inadéquate ou inexisteante contre les transitoires entraîne en amont des dommages au niveau du convertisseur CC/CC, et en aval des dysfonctionnements. D'où des temps d'arrêt et des coûts de maintenance. Pour être réalistes, les calculs et les hypothèses de travail doivent s'appuyer sur un transitoire normalisé comme celui de la norme IEC 61000-4-5.

Filtre d'immunité

La **figure 2** représente le concept d'un filtre d'immunité (en vert) à deux étages de filtrage. Le premier limite la hauteur d'un transitoire au moyen de composants tels que des diodes d'écrêtage unidirectionnelles.

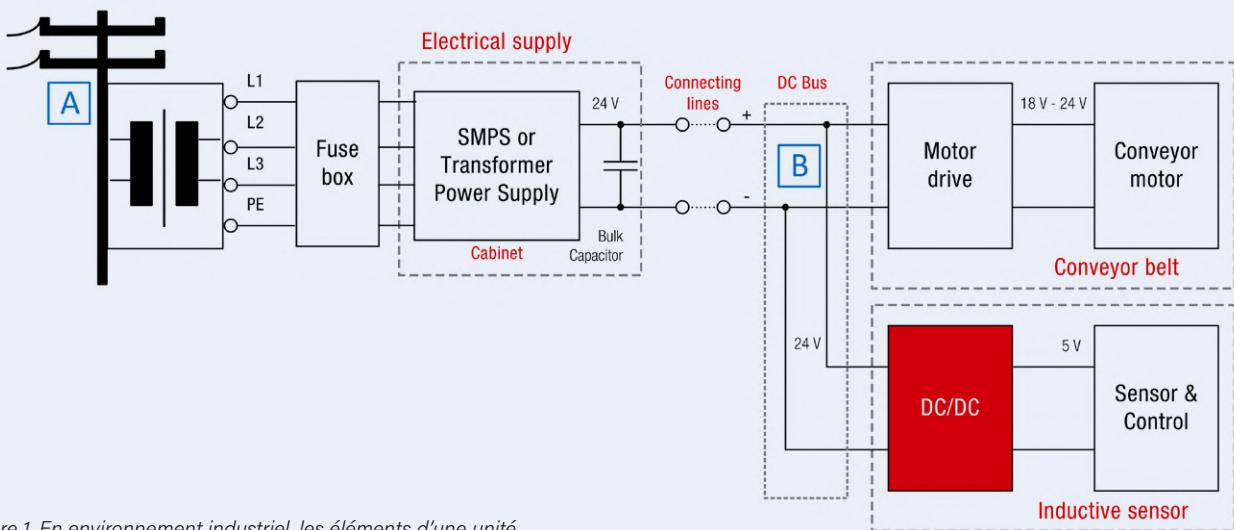


Figure 1. En environnement industriel, les éléments d'une unité de production sont typiquement alimentés par un bus CC.

Le second atténue les tensions excédant la tension de service maximale du convertisseur DC/DC à l'aide d'un filtre LC passif (recommandé).

La valeur critique

La tension d'entrée du régulateur à découpage est le paramètre le plus critique. La fiche technique d'un module de puissance indique en général deux valeurs : sa tension d'entrée maximale absolue, c.-à-d. à ne pas dépasser sous peine de dommages irréversibles, et sa tension de service maximale, c.-à-d. la valeur spécifiée par le fabricant pour l'utilisation du module. Un filtre efficace doit empêcher cette dernière d'être dépassée, même en cas de surtension transitoire. Les calculs qui suivent prennent comme exemple le module de puissance 173010535 Sip-3 dont la tension d'entrée maximale V_{in} vaut 44 V.

Conception d'un filtre d'immunité

Pour cet article, l'entrée du module de puissance est supposée être protégée par une diode d'écrêtage unidirectionnelle. Une telle diode a presque la même caractéristique V-I qu'une diode Zener, autrement dit s'utilise normalement en sens inverse, et devient conductrice lorsque la tension appliquée excède sa tension de claquage. La valeur de la tension limite d'écrêtage (notée ci-après $V_{Clamp\ max}$) est alors déterminée par l'intensité du courant traversant le composant. L'exemple numérique qui suit montre comment estimer les valeurs des composants d'un filtre. Cette modélisation simplifiée permet un cycle d'affinement rapide des valeurs trouvées par essais en conditions réelles.

Nous aurons besoin des paramètres suivants pour protéger efficacement le module de puissance à l'aide d'un filtre reposant sur une diode suppresseuse de transitoires (appelée ci-après diode TVS) :

- V_{DC} : tension d'alimentation du module de puissance
- V_{BR} : tension lorsque 1 mA traverse la diode TVS
- I_{Peak} : courant de pointe max. traversant la diode TVS @ $V_{Clamp\ max}$
- P_{Diss} : puissance dissipée max. autorisée pour la diode TVS
- $V_{Clamp\ max}$: tension lorsque la diode est traversée par le courant max. spécifié.

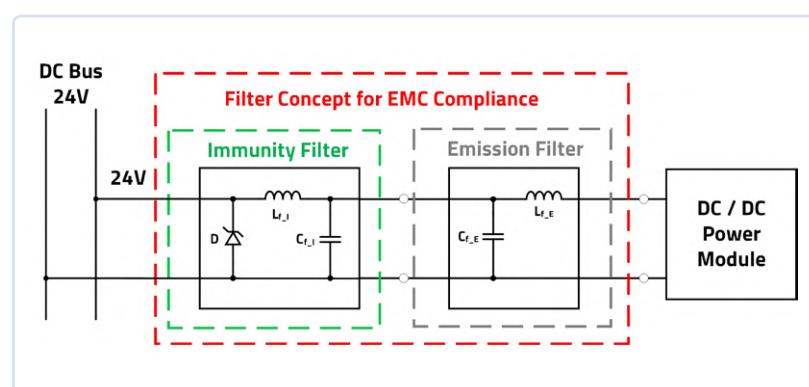
Premier étage du filtre

Détermination de V_{DC}

La valeur déterminant V_{DC} est la tension maximale possible sur le bus CC, pas sa valeur nominale (24 V pour un bus de 24 V). En environnement industriel, un bus de 24 V_{CC} est spécifié pour opérer entre 19,2 et 30 V. La valeur qui nous intéresse est donc $V_{DC} = 30$ V.

La suite des calculs nécessite une diode TVS adaptée à la situation. Le catalogue en ligne de Würth Elektronik [1] fournit deux candidates possibles : les références 824541301 et 824551301.

Figure 2. Concept de filtre (immunité + antiparasites) pour conformité CEM des modules de puissance Mag³C.





Détermination de V_{BR}

V_{BR} représente la tension aux bornes de la diode TVS traversée par un courant de 1 mA. Cette valeur, ici 35,05 V, n'est pas une constante puisque la transition est de type PN et a donc une tolérance – de $\pm 5\%$ d'après la fiche technique. V_{BR} est donc comprise entre 33,325 V et 36,8025 V, intervalle dans lequel la diode est conductrice lorsque le courant vaut 1 mA.

Nous devons ensuite déterminer $V_{Clamp\ max}$, la tension max. d'écrêtage d'un transitoire.

Détermination de $V_{Clamp\ max}$

Cette valeur est elle aussi fournie par la fiche technique. Elle vaut, pour la diode choisie, 48,4 V pour un courant de crête I_{Peak} de 31 A représentant une impulsion de 10/1000 μ s.

Les valeurs utilisées sont celles obtenues dans un laboratoire idéal où règne une température constante de 25 °C. La réalité est bien sûr différente, puisqu'en pratique une diode TVS peut être exposée à une température de 55 °C. Les calculs doivent donc inclure un facteur de température.

Ce sont $V_{Clamp\ max}$ et la puissance crête qui dépendent le plus fortement de la température.

L'**équation 1** montre l'effet de la température sur $V_{Clamp\ max}$:

$$V_{Clamp\ max}(T_j) = V_{Clamp\ max}(25^\circ\text{C}) * \left(1 + \alpha T * (T_j - 25^\circ\text{C})\right)$$

Pour « l'état de veille » durant lequel le seul courant traversant la diode TVS est le courant de fuite de 1 μ A, la température de la jonction (T_j) est à peu près égale à la température ambiante. En prenant pour la TVS un coefficient de température αT de $9,9 \times 10^{-4} 1/\text{C}$, on trouve que $V_{Clamp\ max}$ vaut 49,84 V à 55 °C. Cette valeur servira de point de départ pour le dimensionnement du second étage du filtre d'immunité.

Figure 3. Circuit équivalent pour les calculs du second étage du filtre d'immunité.

Second étage du filtre

Il nous faut maintenant estimer l'atténuation du filtre et trouver les meilleures valeurs possibles pour les composants qui le constituent. Commençons par l'atténuation. Sa valeur minimale est donnée par l'**équation 2**.

$$G=20 \cdot \log\left(\frac{V_{PM\ Max}}{V_{Clamp\ max}}\right) \quad G=20 \cdot \log\left(\frac{44V}{49.84V}\right) = -1.08dB$$

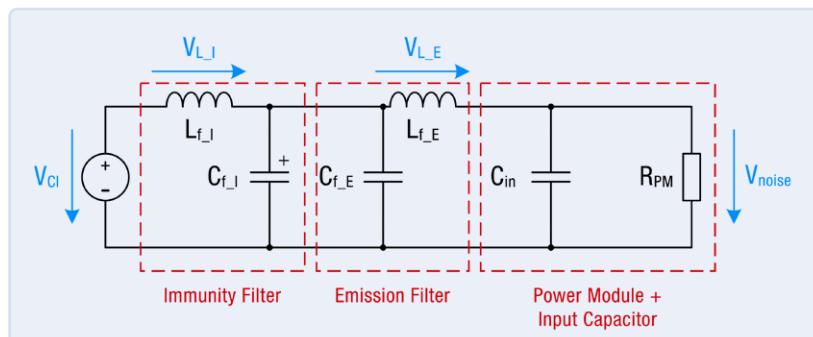
Notez que nous n'utilisons pas la lettre A (atténuation) mais G (gain). Un gain négatif équivaut à une atténuation.

L'**équation** prend en compte la valeur d'écrêtage $V_{Clamp\ max}$ de la diode TVS durant un transitoire, ainsi que $V_{PM\ Max}$, la tension maximale de service du convertisseur choisi. L'objectif est de concevoir le circuit de la **figure 3** en ajoutant un circuit de filtrage LC à la diode TVS. Tenant compte du calcul donnant la capacité d'un condensateur de filtrage adéquat, le concepteur peut sélectionner la valeur de l'inductance. Comme l'inductance du filtre est en série avec l'application et que sa résistance (R_{DC}) provoque des pertes indésirables, l'inductance sélectionnée doit avoir une valeur R_{DC} la plus faible possible au regard du courant de sortie maximal du convertisseur CC/CC. Pour cet exemple, nous avons choisi l'inductance WE PD2 (744776112) de 12 μ H, à courant nominal de 2,72 A et R_{DC} de 336 m Ω . La résistance d'entrée du module de puissance peut être déterminée en combinant les tensions d'entrée et de sortie, le courant de sortie, ainsi que le rendement durant le fonctionnement. Le résultat est donné par l'**équation 3**.

$$R_{PM} = \frac{V_{in}^2}{V_{Out} \cdot I_{out}} = \frac{V_{in}^2}{P_{in}} \quad R_{PM} = \frac{(24V)^2}{5V * 1A} = 101\Omega$$

La **figure 3** montre le circuit équivalent. La diode TVS attaquée par des pics de tension y est représentée par une source de tension simplifiée. Viennent ensuite un filtre LC d'immunité (contre les transitoires), un filtre LC d'atténuation des parasites, le condensateur d'entrée du convertisseur CC/CC, et la résistance d'entrée du régulateur.

Comme il s'agit d'une application en intérieur et qu'il y a donc surtension par couplage indirect, les hypothèses et calculs suivants sont basés sur une impulsion de 8 / 20 μ s. On peut simplifier encore plus le modèle en omettant C_{f_E} et L_{f_E} puisque ce filtre vise à supprimer les perturbations à la fréquence de



commutation du module de puissance (**figure 4**). Cette fréquence vaut typiquement 520 kHz, et on se base ici sur une valeur de 1 kHz pour la surtension.

Pour déterminer l'atténuation G du filtre, il faut comparer la tension de sortie à celle d'entrée du système (**équation 4**).

$$\frac{V_{\text{noise}}}{V_{\text{Cl}}} = \frac{1}{Z_{\text{Lf}} + \left(\frac{1}{Y_{\text{Cf}} + Y_{\text{Cin}} + Y_{\text{RPM}}} \right)}$$

$$G = 20 \cdot \log \left(\left| \frac{V_{\text{noise}}}{V_{\text{Cl}}} \right| \right)$$

$$G_{\text{immunity}_{\text{filter}}} = 20 \cdot \log \left(\left| 1 - \omega^2 L_f (C_f + C_{\text{in}}) + j\omega \frac{L_f}{R_{\text{PM}}} \right| \right)$$

La résistance d'entrée du convertisseur calculée plus haut (R_{PM} , **équation 3**) permet de déterminer la capacité du condensateur nécessaire :

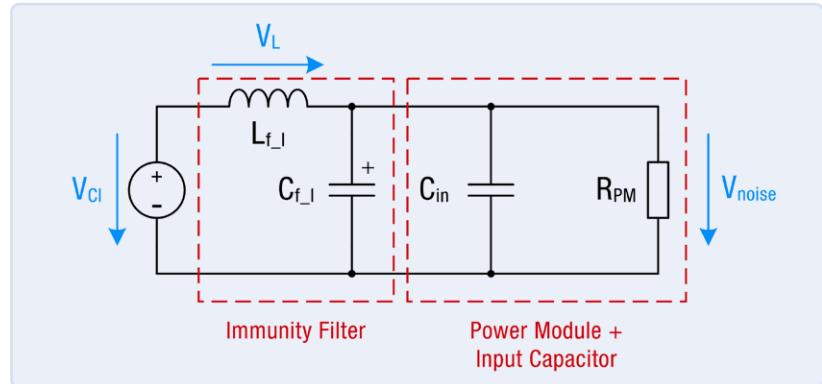
$$C_f = \frac{1 - \left(10^{\frac{G}{10}} - \left(\omega \frac{L_f}{R_{\text{PM}}} \right)^2 \right)^{\frac{1}{2}}}{\omega^2 \cdot L_f} - (C_{\text{in}} + C_{\text{f_emission}})$$

L_f représente l'inductance du filtre d'immunité, R_{PM} la résistance d'entrée CC du convertisseur, C_{in} la capacité du condensateur d'entrée, et $C_{\text{f_emission}}$ la capacité du condensateur d'entrée antiparasite d'un filtre d'entrée en Pi. Les valeurs sont fournies par la fiche technique du module 173010535. L'examen du spectre de fréquence d'une surtension montre que la valeur la plus élevée de la tension de bruit se produit à $f = 1 \text{ kHz}$. Cette valeur est donc utilisée pour le calcul du pire scénario. Avec une inductance de $12 \mu\text{H}$, le calcul pour le condensateur du filtre donne $C_{\text{f_i}} = 218 \mu\text{F}$. Le condensateur choisi serait donc un $220 \mu\text{F}$ (860010775018) puisqu'il s'agit du plus proche à être supérieur à $218 \mu\text{F}$. Un condensateur de capacité plus basse n'offrirait pas assez d'atténuation. Au final, les composants du filtre seraient les suivants (références Würth Elektronik) :

Diode TVS = 824541301, $L_{\text{f_I}} = 744776112$, $C_{\text{f_I}} = 860010775018$

Le **tableau 1** indique l'influence de la température sur $V_{\text{Clamp max}}$, et donc sur la capacité du condensateur de filtrage.

Les capacités ont été obtenues par calcul, mais celle des condensateurs réels sont affectées d'une tolérance



pouvant aller jusqu'à $\pm 20\%$. Si la dépendance en température de $V_{\text{Clamp max}}$ n'est pas prise en compte, le condensateur sélectionné peut avoir une capacité trop faible.

Transitoires et interférences

Plusieurs paramètres sont à prendre en considération lorsqu'on cherche à concevoir un filtre efficace contre les surtensions transitoires. Ces filtres jouent un rôle crucial dans les environnements industriels où l'infrastructure électrique provoque souvent des pics de tension. Ils protègent efficacement les convertisseurs CC/CC tout en amortissant les rayonnements de haute fréquence. ↳

VF : Hervé Moreau — 230180-04

Tableau 1.

Température ambiante	$V_{\text{Clamp max}}$	Condensateur de filtrage
25 °C	48,4 V	178 µF
55 °C	49,84 V	218 µF



Figure 4. Circuit équivalent simplifié pour les calculs du second étage du filtre d'immunité.



À propos de l'auteur

Timur Uludag a obtenu son diplôme d'ingénieur en mécatronique à l'université des sciences appliquées de Regensburg, Allemagne. Il a travaillé plusieurs années dans le domaine des alimentations à découpage et de la conception de circuits analogiques. Depuis 2015, Timur est responsable du marketing technique des modules de puissance MagIC au sein de la division eiSos du groupe Würth Elektronik. Il a notamment en charge le calendrier du lancement commercial des nouveaux modules de puissance.

LIENS

- [1] Fiches techniques des diodes TVS de Würth Elektronik : <https://we-online.com/en/components/products/WE-TVSP>

L'offre de Wiha comprend trois testeurs de tension, qui diffèrent par le prix et leurs possibilités.

appareil de mesure Wiha

testeurs et compteurs d'installation sur lesquels vous pouvez compter



Contribué par Transfer Multisort Elektronik Sp. z o.o.

Les appareils de mesure sont indispensables pour tout électricien souhaitant travailler en toute sécurité. La gamme d'appareils de mesure Wiha présentée ici fait partie de l'impressionnant assortiment de Transfer Multisort Elektronik.

Les appareils de mesure sont un élément d'équipement qui peut être utile dans l'atelier de maintenance d'une grande usine moderne et d'un garage de bricolage du dimanche. Comme son nom l'indique, l'appareil de mesure est utilisé pour tester les valeurs caractéristiques des dispositifs électriques et des installations, bien qu'ils puissent également être utilisés pour mesurer les conditions environnementales. Grâce à eux, il sera beaucoup plus facile de trouver un défaut dans un équipement non travaillant ou de vérifier le bon fonctionnement du circuit juste préparé.

Lorsque vous recherchez la bonne classe d'appareils de mesure [1], il vaut la peine de s'intéresser à l'offre de l'entreprise WIHA [2]. Les outils de ce fabricant allemand sont connus dans le monde entier principalement pour une excellente qualité, une précision et une fiabilité. Cela peut être confirmé par la multitude de récompenses que l'entreprise reçoit chaque année, entre autres pour son innovation, sa conception et son approche du client. Maintenant, l'offre a été complétée par des appareils de mesure.

Testeurs de tension sans contact et indicateurs de tension Wiha

Le testeur de tension sans contact est presque un équipement obligatoire de chaque électri-

cien. Il permet de manière simple et facile de constater si un circuit donné est sous tension. Les mesures faites avec ce type de dispositifs sont extrêmement sûres car l'équipement peut détecter la tension à une certaine distance, il n'est pas nécessaire de toucher une zone potentiellement dangereuse.

L'offre de Wiha comprend trois testeurs de tension, qui diffèrent par le prix et leurs possibilités. L'outil qui est le plus simple est équipé d'une seule diode informant l'utilisateur d'une détection de tension de 90 V AC à 1000 V AC. Les deux autres appareils informeront les autres appareils sur l'échelle LED, et les œufs seront toujours accompagnés d'un signal acoustique. De plus, les testeurs de meilleure classe ont été équipés d'une lampe de poche. Il convient également de mentionner que la plage de tension mesurée est supérieure à celle de la structure de base et varie de 12 V AC à 1000 V AC. La seule différence entre l'indicateur intermédiaire et la classe supérieure est le certificat ATEX. Grâce à cela, nous savons que le modèle le plus élevé peut être utilisé avec succès dans des endroits menacés d'une explosion potentielle.

Testeurs de tension bipolaires (testeurs électriques)

Un testeur de tension à deux points est un autre appareil utile dans l'atelier surtout lors

de la mesure des installations électriques et des armoires de contrôle.

Chacun des trois dispositifs disponibles permet de mesurer la tension alternative dans la plage de 12 V à 1000 V et continue de 12 V à 1500 V conformément à DIN EN 612433 et VDE 0682-401 et un test de continuité de circuit de 0 Ω à 500 kΩ. La plupart des différences sont entre le dispositif de classe moyenne et supérieur. Le testeur indirect est destiné à tester les installations solaires et permet également la mesure de tension sans contact, la mesure de la fréquence (de 1 Hz à 950 Hz) et la résistance (de 0 Ω à 1999 Ω). Le dispositif de classe supérieure est destiné à travailler dans l'industrie de l'électromobilité et vous permet de mesurer la fréquence de la plage de 40 Hz à 400 Hz.

Multimètre numérique

Les multimètres numériques sont l'un des types de dispositifs de mesure les plus courants. La multitude de mesures possibles et la simplicité d'utilisation doivent leur popularité. Dans cette catégorie d'appareils, le fabricant allemand a préparé deux mesures différent dans certains détails. Le multimètre plus simple, WIHA.45218 permet de mesurer la continuité du circuit < 30 Ω, mesurer la résistance jusqu'à 40 mΩ, mesurer la tension dans la plage de 0 V à 600 V (à la fois avec courant fixe et alternatif),



Le testeur permet de vérifier la continuité du circuit de 0 Ω to 500 kΩ.



WIHA.45218 permet de mesurer la tension dans la plage de 0 V to 600 V.



Le mesureur à pince Wiha permet la mesure du courant à un maximum de 400 A AC/DC.



Le testeur permet une mesure facile et rapide de la prise électrique.



Solution idéale pour les services de maintenance.

mesurer le courant de 40 mA à 10 A AC / DC et mesurer la fréquence jusqu'à 5 MHz. L'appareil a également une fonction de maintien et un réglage automatique de la plage de mesure. Le deuxième multimètre – WIHA.45215 est un dispositif légèrement plus compliqué et permet de mesurer la continuité du circuit < 30 Ω, la résistance jusqu'à 200 mΩ, la tension de 0 V à 1000 V AC / DC, l'intensité de courant de 40 mA à 10 A AC / DC et la fréquence jusqu'à 60 MHz. De plus, l'appareil est équipé d'une fonction TrueRMS, HOLD, Min/Max. Il convient également de mentionner qu'une mesure de tension sans contact est possible et que les plages multimètres peuvent être définies manuellement ou automatiquement.

Mesureur à pince Wiha

Les mesureurs à pince sont utiles principalement pour la mesure de l'intensité du courant sans contact. Il suffit de placer tout simplement le câble testé entre les pinces du mesureur pour lire la valeur du courant. Grâce à cette fonctionnalité, ce type d'appareil a gagné en popularité auprès des électriciens. WIHA propose un mesureur à pince professionnel WIHA.45219. Il s'agit d'un dispositif qui permet de mesurer la continuité du circuit < 30 Ω, la tension dans la plage de 0 V à 1000 V à un courant alternatif et de 0 V à 1500 V à un courant continu jusqu'à un maximum de 400 A AC / DC, la fréquence

jusqu'à 5 MHz et la résistance jusqu'à 40 mΩ. Le mesureur est équipé également d'une fonction TrueRMS, HOLD et Min/Max/Avg. Il convient de mentionner que l'appareil est en outre équipé d'une lampe de poche et peut être utilisé pour les installations solaires et dans l'industrie de l'électromobilité.

Testeur de prises réseau

Le testeur de prise réseau 45220 est un appareil intéressant de la marque Wiha. Ce dispositif permet d'effectuer les tests faciles et sûrs de divers types de prises de réseau. Il y a trois LED sur le panneau avant, qui brillent dans la configuration appropriée, selon l'état de la prise. Le testeur peut détecter, entre autres, la connexion inversée des câbles L/N, manque de phase ou de mise à la terre. Ce type d'équipement peut être extrêmement utile dans le travail d'un électricien.

Indicateur de champ rotatif (testeur d'ordre de phase)

Dans les systèmes de moteurs en trois phases, l'ordre dans lequel les phases sont connectées est extrêmement important. Lorsqu'une erreur se produit, le moteur peut être endommagé et provoquer des temps d'arrêt indésirables de la ligne de production. Le testeur d'ordre de phase est un appareil qui permet facilement de vérifier l'exactitude de la connexion du moteur en trois phases. Après avoir connecté le compteur, grâce aux LED construites,



Le résultat de la mesure est signalé par la LED et le signal sonore.

indiquera la direction de la rotation et informera l'utilisateur de l'erreur dans le cas d'une connexion incorrecte. Ce type d'indicateur de champ rotatif doit être un équipement obligatoire pour les services de maintenance.

Testeur de continuité de câbles

Lors de la construction de différents types d'installations électriques, même le meilleur professionnel peut faire une erreur. Pour trouver cette erreur, il peut être utile d'utiliser le testeur de continuité de câbles. Ce dispositif est dédié aux professionnels et permet la mesure dans deux plages (jusqu'à 10 Ω et jusqu'à 500 Ω). Le test est compatible avec la norme de mesure Cat II 400 V. La continuité du circuit est signalée en utilisant la LED placée sur le compteur ainsi qu'un signal acoustique à volume élevé. Selon le fabricant, il sera entendu dans une autre pièce et même à un autre étage. ▶

230192-04

LIENS

- [1] https://www.tme.eu/fr/katalog/appareils-de-mesure_100164/p/wiha_248/
- [2] Produits WIHA : http://www.tme.eu/fr/linecard/p/wiha_248/
- [3] Source : <http://www.tme.eu/fr/pages/library-articles/51790/appareil-de-mesure-wiha>



Figure 1. Le Moku:Lab dispose de deux entrées et deux sorties analogiques. La fente sur le dessus du boîtier sert de support de tablette.



automatisation des tests et partage des résultats

Par Stuart Cording, pour Mouser Electronics

Lorsqu'il s'agit de tester des applications complexes, il est essentiel de disposer de quelques outils incontournables tels que ces appareils de mesure de la gamme disponible chez Mouser Electronics.

Les applications d'aujourd'hui sont de plus en plus complexes à développer, mais aussi à tester. Pour résoudre tous ces petits problèmes sporadiques mais difficiles à corriger, l'automatisation des tests apparaît dès lors comme une nécessité, y compris en phase de recherche et de développement. Or, exécuter des tests la nuit ou le week-end exige un équipement de test et de mesure flexible qui puisse être programmé pour vérifier une multitude de cas. Les résultats des mesures doivent ensuite être enregistrés pour être éventuellement consultés plus tard durant le débogage.

Heureusement, la gamme d'équipements de test et de mesure disponibles sur le marché ne cesse de se développer et les capacités d'automatisation et de partage des résultats de ces appareils ne cessent d'augmenter. Les fabricants de matériel haut de gamme ajoutent toujours plus de fonctionnalités et proposent des programmes de certification garantissant des résultats calibrés. Mais de nouveaux acteurs ont fait leur entrée sur le marché. Ces fabricants s'orientent davantage sur une flexibilité sans précédent avec la possibilité d'intégrer leurs appareils dans des solutions de test HIL (*hardware-in-the-loop*) artisanales. Ils offrent ainsi une alternative intéressante aux équipes de

recherche et de développement désireuses de tester et d'itérer rapidement leurs conceptions, mais qui ne disposent pas d'un budget à cinq ou six chiffres pour acquérir des outils de mesure haut de gamme ou une configuration HIL professionnelle.

Le top de la flexibilité de mesure et le « headless »

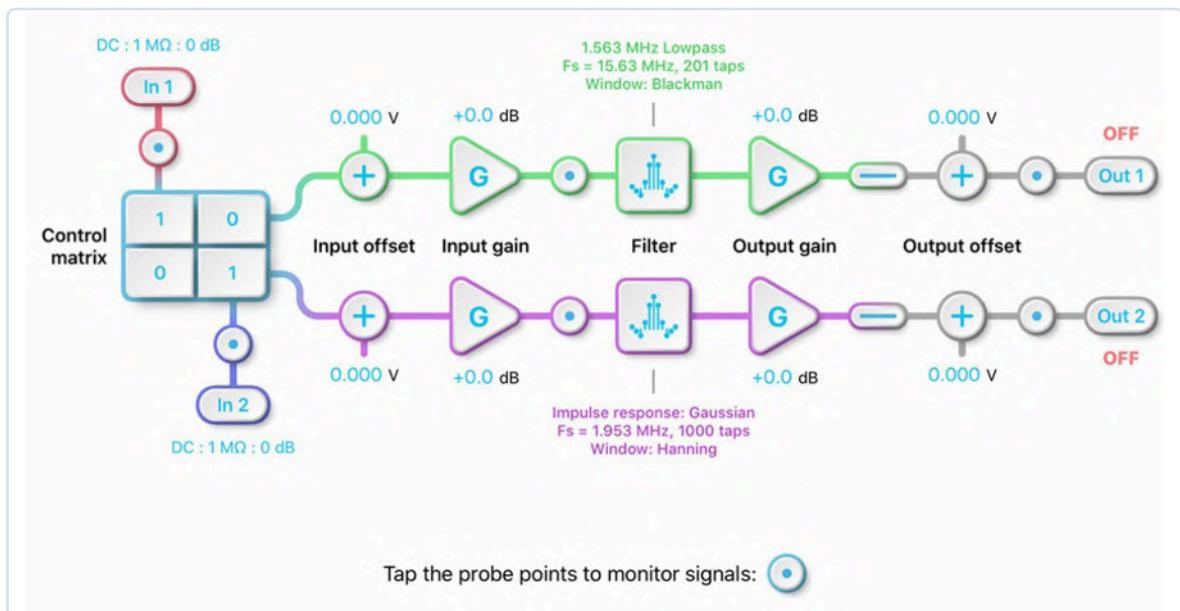
L'offre en matière d'équipements de mesure « headless » (ou sans tête) s'est considérablement étendue ces derniers temps. Il s'agit d'appareils utilisant un logiciel (qui peut tourner sur un ordinateur portable, un PC ou une tablette) pour afficher les résultats de mesure. Ils sont généralement constitués d'un SoC FPGA couplé à des systèmes frontaux analogiques (AFE) à grande vitesse et des convertisseurs numérique-analogique (CNA). Le FPGA permet aux utilisateurs de reconfigurer l'outil à la volée. Il peut ainsi servir tantôt d'oscilloscope, tantôt d'analyseur de signal ou même de contrôleur de boucle PID. Grâce à leur espace de stockage interne, ils sont aussi parfaits pour enregistrer les mesures, que ce soit en rafale sur une milliseconde ou en continu sur plusieurs jours.

Dans le genre, le Moku:Lab de Liquid Measurement [1] est une plateforme

matérielle reconfigurable regroupant 12 instruments différents. Son logiciel est disponible pour Windows et Mac OS ainsi que sous forme d'appli pour l'iPad. Avec son boîtier circulaire de 20 cm (7,9 po) de diamètre et 4,4 cm (1,7 po) de hauteur, Moku:Lab trouvera aisément sa place, même sur une paillasse de labo encombrée. Le dessus du boîtier présente une fente destinée à accueillir votre iPad. Les données, paramètres et captures d'écran sont enregistrés sur la mémoire interne, mais peuvent aussi être directement partagés avec un service cloud comme Dropbox ou envoyés par e-mail.

Sur le panneau avant se trouvent quatre connecteurs BNC. À droite, les sorties analogiques avec un taux d'échantillonnage de 1 GSa / s par canal à une résolution de 16 bits et une bande passante (-3 dB) de > 300 MHz. À gauche, les entrées analogiques avec une bande passante (-3 dB) de 200 MHz sur 50 Ω et un taux d'échantillonnage de 500 MSa / s par canal à une résolution de 12 bits. La base de temps interne affiche une précision supérieure à 500 ppb (**figure 1**).

À l'arrière du boîtier, deux connecteurs BNC offrent une entrée de 10 MHz pour une horloge de référence externe pour l'un et une sortie de 10 MHz pour l'autre. Ces connecteurs permettent de synchroniser entre eux les signaux capturés par différents appareils Moku. On notera également la présence d'un connecteur d'entrée de déclenchement. Outre ces connecteurs BNC, on trouvera également un port Ethernet, une interface USB, un port d'alimentation USB (pour charger une tablette), une



baie pour carte SD et l'entrée d'alimentation CC. L'appareil dispose aussi d'une connectivité Wi-Fi (802.11 b/g/n).

Parmi les 12 instruments regroupés dans cet appareil unique, on trouvera, outre les instruments courants auxquels on peut s'attendre dans pareil dispositif, des outils qui utilisent simultanément les entrées et les sorties. Il s'agit entre autres d'un régulateur PID avec deux chemins entièrement configurables et un générateur de filtres RIF (filtres à réponse impulsionnelle finie) avec 14 819 coefficients (**figure 2**). Notons la présence du Laser Lock Box, un dispositif permettant de stabiliser la fréquence laser

grâce à la technique Pound-Drevel-Hall [2] (notamment utilisée dans des domaines d'application tels que les détecteurs d'ondes gravitationnelles, la mesure du temps et la physique atomique).

Des API sont fournies pour MATLAB, LabVIEW et Python afin d'offrir de multiples possibilités pour l'implémentation de tests automatisés et de la journalisation des données, lesquels sont pris en charge avec des exemples hébergés sur GitHub [3]. Enfin, une application à deux canaux robuste et simple permet de programmer l'enregistrement de données en quelques lignes de code Python seulement (**figure 3**).

```
from pymoku import Moku, StreamException
from pymoku.instruments import Datalogger
import time

m = Moku.get_by_name('Moku')

try:
    i = m.deploy_or_connect(Datalogger)

    # 100 samples per second
    i.set_samplerate(100)

    # Start data logger
    i.start_data_log(duration=10, use_sd=True, ch1=True, ch2=True,
                     filetype='bin')

    # Upload the log file to the local directory
    i.upload_data_log()

    # Clean up
    i.stop_data_log()

except StreamException as e:
    print("Error occurred: %s" % e)
finally:
    m.close()
```

Figure 3. Exemple d'enregistreur de données simple capturant les signaux des deux entrées analogiques, écrit en Python pour Moku:Lab.



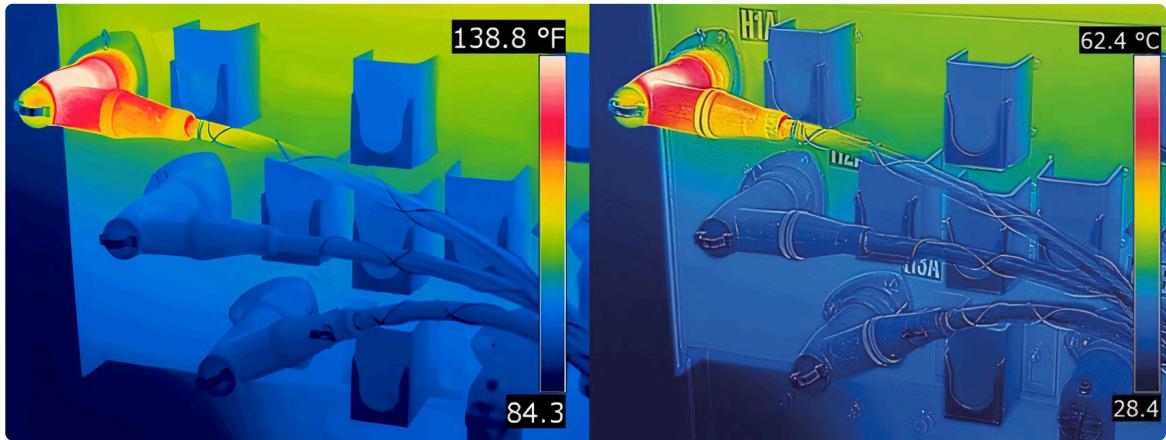
Figure 4. Caméras thermiques telles que la Teledyne FLIR MR265.

Figure 2. L'interface du générateur de filtres RIF du Moku:Lab est simple à configurer et offre une image plus claire de ce qui se passe dans le chemin du signal.

Test à long terme de problèmes thermiques

Supposons que vous voulez examiner l'autoéchauffement ou la dissipation thermique à long terme du boîtier de votre appareil. Des caméras infrarouges peuvent alors vous aider à localiser les points chauds et froids. Teledyne FLIR est un acteur majeur dans le domaine de la mesure visuelle de la température. Ses produits sont compatibles avec un grand nombre de logiciels permettant de simplifier la collecte et l'analyse de données et de faciliter la collaboration. Des appareils comme le MR265 [4] (**figure 4**) sont dotés du système d'amélioration de l'imagerie dynamique multispectrale [5] MSX™ qui associe une caméra thermique 160 × 120 (19 200 pixels) à une caméra visuelle 2 MP. La technologie MSX™ offre ainsi une image plus claire que les caméras reposant uniquement sur des capteurs

Figure 5. À gauche, sans MSX™, les dénominations sur le panneau de connexion n'apparaissent pas. À droite, avec MSX™, les dénominations apparaissent clairement et la qualité globale de l'image est bien meilleure.



d'image thermique, qui ont une résolution plus basse et fournissent par conséquent des images plus floues.

Ces caméras classiques produisent parfois des images si médiocres qu'il devient presque impossible d'en comprendre le contenu sans savoir où l'image a été prise. MSX™ remédie à ce problème en associant l'imagerie thermique à une image à contraste élevé produite par la caméra pour lumière visible (**figure 5**). La combinaison de ces deux images permet de rendre les détails visibles plus clairs et la mesure thermique plus cohérente.

Ces caméras sont évidemment prises en

charge par la suite logicielle FLIR Thermal Studio [6]. Cette suite permet de collecter des images individuelles comme des vidéos en streaming. L'utilisateur peut appliquer des deltas et des formules en vue de produire une analyse d'images, définir des alarmes de mesure et partager facilement les résultats avec ses collègues. Avec les appareils qui le permettent, le logiciel prend en charge le double flux, c'est-à-dire qu'il enregistre à la fois le flux vidéo radiométrique et celui en lumière visible de la cible mesurée.

Le MR265 offre une fréquence d'actualisation d'image de 9 Hz et une plage de mesure des températures de 0 à 100 °C (32-212 °F)

pour des objets à une distance supérieure à 10 cm. Doté d'un capteur de mesure intégré sans broches (ou de sondes à billes et à broches disponibles séparément), l'appareil peut aussi mesurer le taux d'humidité de 7 à 100 %. Les résultats s'affichent sur l'écran TFT couleur intégré de 7,1 cm (2,8 po) avec une résolution de 320 × 240 pixels.

La mesure automatisée avec les équipements de laboratoire traditionnels

Même si les outils de mesure headless hautement configurables présentent un certain attrait dans bon nombre de situations,

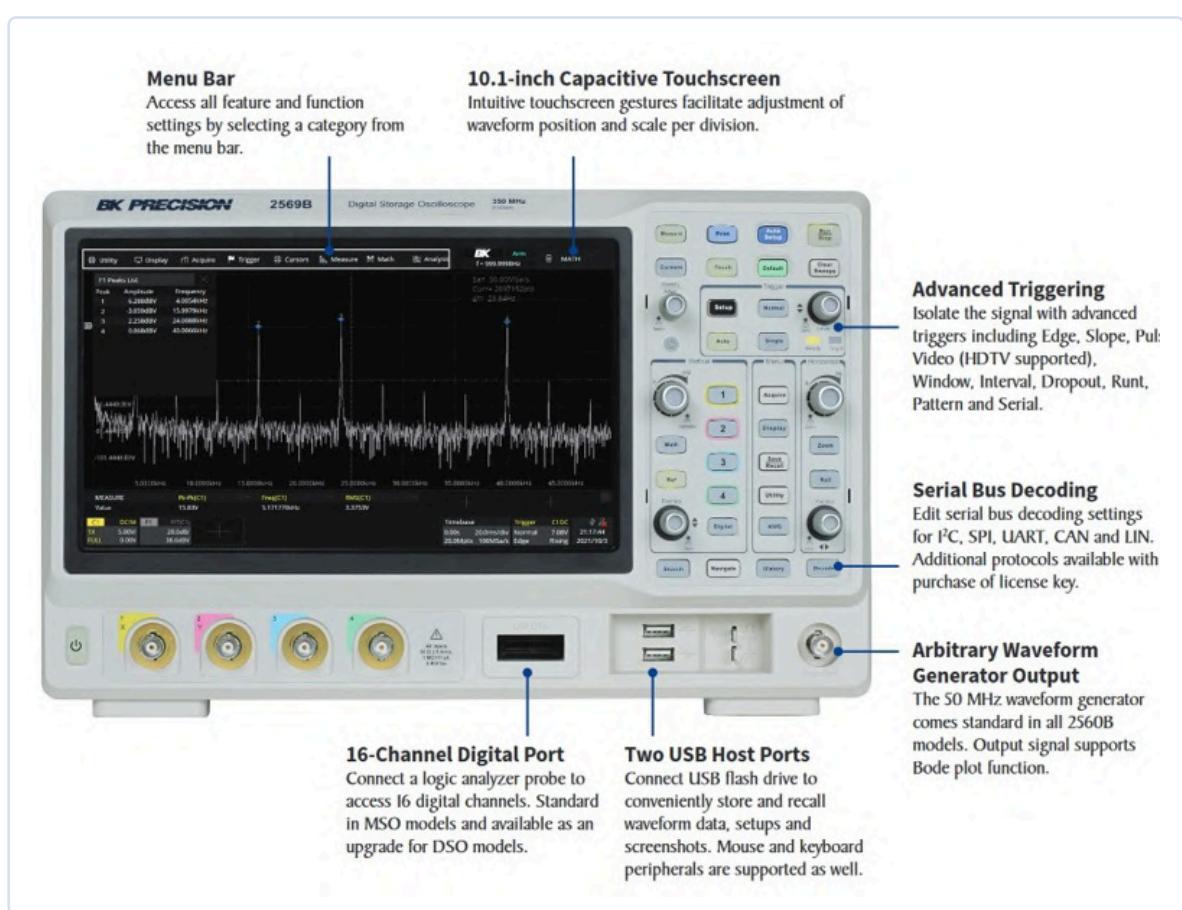


Figure 6.
Le 2569B-MSO de B&K Precision dispose de quatre canaux de 350 MHz et d'un analyseur logique numérique 16 bits.

le simple fait de devoir acheter en plus une tablette ou un ordinateur portable pour en afficher les résultats peut être rédhibitoire. Surtout s'il suffit parfois de se fier tout simplement à ce qu'on connaît. Par exemple, la série d'oscilloscopes à signaux mixtes 2560B [7] de B&K Precision avec leur grand écran tactile capacitif de 25,65 cm (10,1 po) associé à des boutons tactiles et des boutons rotatifs. De plus, avec seulement 111 mm (4,4 po) de profondeur, on ne pourra pas leur reprocher d'être trop encombrants sur le banc.

Le modèle haut de gamme est le 2569B-MSO. Il dispose de quatre canaux d'entrées avec une bande passante de 350 MHz et un port numérique à 16 canaux (**figure 6**). Il intègre également un générateur de formes d'onde de 50 MHz qui prend en charge une fonction de tracé de Bode et propose différents modes de déclenchement avancés (bord, pente, impulsion et vidéo – y compris HDTV) ainsi que le décodage de bus série pour les interfaces courantes utilisées dans les systèmes embarqués. D'autres interfaces série (CAN FD, FlexRay, I2S, MIL-STD-1553B et SENT) peuvent être achetées sous forme de mises à niveau.

Le contrôle à distance et la capture de données sont pris en charge par les interfaces USB et Ethernet. Pour la prise en

charge par USB, un pilote USBTMC (USB Test & Measurement Class) est nécessaire. L'interface de programmation recommandée est la NI-VISA de National Instruments [8], mais d'autres choix existent si vous prenez la peine de vous renseigner. GitHub propose une série de projets prenant en charge les périphériques USBTMC en Python [9]. Grâce au manuel de programmation de la série 2560B [10], il vous sera possible de créer des scripts (**figure 7**) pour automatiser la capture et l'enregistrement des données et pour décoder les données sérielles, effectuer des mesures ou même définir des masques pour les tests de limite réussite/échec de signaux.

Tester, encore tester, toujours tester

Même si la plupart des ingénieurs en développement considèrent le travail sur le banc d'essai comme le meilleur moment de la journée, il faut bien admettre que s'évertuer à identifier les pannes intermittentes dont la cause n'est pas bien définie tient parfois du chemin de croix. Heureusement, une grande partie des équipements de test et de mesure disponibles aujourd'hui sur le marché (notamment les instruments vus plus haut) peuvent être associés à des logiciels en vue de programmer des tests

automatisés ou l'enregistrement des données. Grâce à leur simplicité et leur flexibilité, ces outils ne nécessitent que des scripts simples ou des procédures de test qui peuvent être créées en quelques instants à l'aide d'un logiciel open source pour tester une hypothèse. Une fois la cause de la panne trouvée, ces tests peuvent être enregistrés en vue d'être réutilisés dans le cadre des tests d'intégration continue pour vérifier que la panne ne se reproduit plus. De plus, les résultats peuvent être facilement partagés sous forme de fichiers journaux par e-mail, à l'aide d'un service de partage de fichiers ou sur la plateforme cloud du fabricant. Les équipes de développement peuvent ainsi faire appel à des collègues qualifiés du monde entier pour examiner et évaluer leurs résultats. ▶

230189-04



À propos de l'auteur

Stuart Cording est un journaliste indépendant qui écrit pour Mouser Electronics.

Spécialiste de contenu vidéo, il se concentre sur la recherche technique approfondie. Il s'intéresse particulièrement à une technique, à son intégration dans des applications pratiques et aux perspectives d'évolution... Mouser Electronics est un distributeur agréé de semi-conducteurs et de composants électroniques qui se concentre sur l'introduction de nouveaux produits par ses principaux partenaires fabricants.

```
import usbtmc
instr = usbtmc.Instrument(<VID>, <PID>, <SERIAL NUMBER>
print(instr.ask("*IDN?"))
# returns 'BK Precision,2569B-MSO,XXXXXXXXXXXX,5.0.1.3.9R3'
```

Figure 7. La prise en charge de l'USBTMC peut être trouvée dans des projets open source. Ce code, écrit en Python, accède à l'oscilloscope en fonction de son VID et de son PID pour acquérir son nom et son numéro de série.

LIENS

- [1] The Moku:Lab from Liquid Measurement: <https://bit.ly/3Fl0BAW>
- [2] Pound-Drevel-Hall technique: https://en.wikipedia.org/wiki/Pound%20%20%20%20%20Drever%20%20%20%20Hall_technique
- [3] Liquid Instruments GitHub: <https://github.com/liquidinstruments>
- [4] MR265 : <http://bit.ly/3Zb4OUK>
- [5] Multi-Spectral Dynamic Imaging enhancement: <https://flir.com/globalassets/industrial/instruments/flir-msx-tech-note.pdf>
- [6] FLIR Thermal Studio Suite: <https://flir.eu/products/flir-thermal-studio-suite/>
- [7] B&K Precision 2560 B Mixed Signal Oscilloscope series: <http://bit.ly/40tFNW7>
- [8] National Instruments NI-VISA product: <https://ni.com/en-gb/shop/software/products/ni-visa.html>
- [9] USBTMC devices in Python: <https://github.com/python-ivi/python-usbtmc>
- [10] 2560B Series Programming Manual: <https://bit.ly/40lx6gi>

zone D

Astuces, bonnes pratiques et autres informations pertinentes



Source: Shutterstock / Olha Birivva

sur le vif

l'électronique des blocs

Ilse Joostens (Belgique)

Il y a environ 15 ans, mon employeur m'avait demandé d'élaborer un bouton d'alarme pour des employés ne se sentant pas en sécurité. Ne trouvant aucune solution commerciale satisfaisante, je décidai de mettre au point un système sans-fil. Lorsque j'évoquai mes idées à une connaissance portée sur la technique, celle-ci me répondit avec dédain : « Ça ne marchera jamais, tu devrais acheter un truc prêt à l'emploi ».

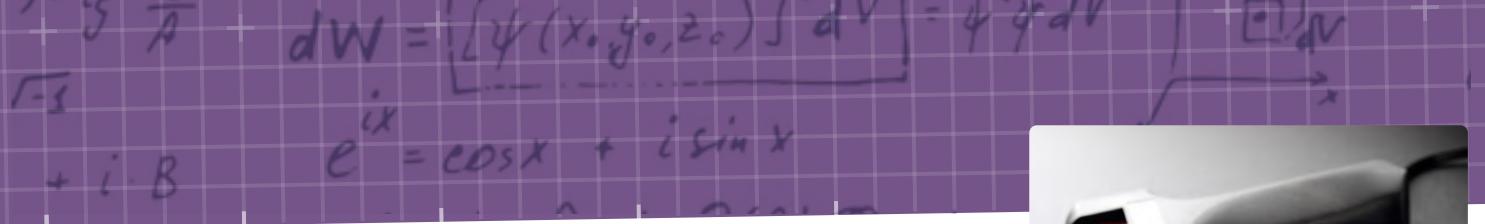
Du sang, de la sueur et des larmes. C'est ce qu'il m'en aura coûté pour faire fi du défaitisme ambiant et réaliser cette alarme – mêlant langage assembleur et codage Manchester. Il n'existe alors aucune bibliothèque de proto-

cole radio à télécharger, et les cartes Arduino n'avaient pas encore conquis le monde. Ce combat d'un autre temps illustre bien la tendance actuelle à vouloir obtenir un résultat rapide avec un minimum d'effort.

Blocs de construction

J'ai eu un jour une discussion animée avec un étudiant qui venait d'obtenir un master en Sciences de l'information. Certaines de ses idées étaient étranges, et il semblait surtout avoir suivi des cours d'arrogance. Selon lui, les micrologiciels, les microcontrôleurs, et plus généralement tout ce qui était de « bas niveau », ne relevaient pas de l'Informatique avec un grand I mais de l'électromécanique. Peut-être n'avait-il d'autre parade que le dédain pour masquer son manque de connaissances sur le sujet. Peut-être. Mais que penser quand on s'entend dire que les systèmes d'exploitation et les langages de haut niveau [1] sont la panacée ? Moi, j'ai pensé à ma cafetière. Au temps qu'elle mettrait à chauffer l'eau si elle devait attendre que son OS, Linux ou autre, soit prêt à bosser.

Il semblerait que cette idée de l'omnipotence du « haut niveau » fasse également son chemin en électronique. Je suis de la vieille école et conçois la plupart de mes circuits avec des composants discrets, une approche



éloignée de la démarche actuelle. La plupart des projets que je vois en ligne et sur *Elektor Labs* [2] sont en effet construits autour d'un nano-ordinateur monocarte et de modules prêts à l'emploi. Quant au logiciel qui pilote l'ensemble, il s'avère bien souvent et à son tour constitué d'un amas de code et de bibliothèques open source glanés ici et là sur le net. Tout le monde semble s'en satisfaire, mais à mes yeux cette façon de faire s'apparente plus à un jeu de construction qu'à une conception de produit réfléchie. J'ai récemment joué avec ChatGPT [3][4], l'agent conversationnel capable de pondre à la demande du code et des descriptions de schémas électriques. Il vous incombe de juger de leur exactitude, mais ici encore la facilité est là. Je n'ai fondamentalement rien contre l'emploi de cartes prêtes à l'emploi et de logiciels open source. Il m'arrive d'y recourir durant la phase de développement d'un projet. Ceci dit, je le fais à des fins d'exploration, pour apprendre quelque chose, pour essayer quelque chose, ou pour ne pas avoir à réaliser trop tôt un circuit imprimé avec de minuscules CMS. Il est par ailleurs indéniable que les cartes de liaison et les modules sont une porte

d'entrée au monde merveilleux de l'électronique, qu'ils contribuent à la formation d'utilisateurs aussi bien débutants qu'avancés – et de tous âges. Divers kits, dont certains se manipulent comme des blocs de construction, ont à cet égard un rôle éducatif précieux – ce sont en quelque sorte les versions *high tech* des coffrets d'initiation classiques.

Système D

Un ami guitariste me dit souvent que la vie d'un musicien est faite de 3 % d'inspiration et de 97 % de transpiration. Je pense qu'il en va de même pour l'électronique. Vous pouvez avoir une excellente idée, mais si votre produit se révèle n'être qu'un fatras de petits circuits imprimés reliés par un fouillis de fils, furent-ils multicolores, votre réputation de professionnel pourrait en prendre un coup. Et même un second si le fonctionnement de votre produit repose sur un échafaudage logiciel monté à la va-vite, au petit bonheur la chance, sans tester toutes les conditions possibles. Il pourrait rejoindre la cohorte d'appareils dont l'interface est mal conçue, ont un comportement inattendu sous certaines conditions, ou dont les boutons font des choses étranges.

N'empruntez pas cette voie. Vous perdriez l'occasion d'élargir vos connaissances, et il y a fort à parier que vous ne comprendriez pas vraiment ce que fait votre bibliothèque open source, ni comment elle opère. Le risque serait également que votre produit dépende de façon critique de la disponibilité des modules qu'il utilise, et que des mises à jour logicielles en brouillent la bonne marche de façon inattendue. La tendance à construire des produits à partir de modules vient aussi de l'abondance des cartes bon marché venues d'Asie. Me sachant affectée par la pénurie de composants, d'aucuns m'ont d'ailleurs suggéré d'acheter une petite carte auprès d'oncle Ali(baba), mais je ne suis pas encore assez désespérée pour ça.

```

    this = $(this)
    target = $($this.attr('data-target'))
    target.replace(/.*(=?#[^\s]+$/),
    if(target.hasClass('carousel')) {
      options = $.extend({}, $target
      slideIndex = $this.attr('data-slide-index')
      options.interval =
        plugin.call(target, options)
      if(slideIndex) {
        target.data('bs.carousel', slideIndex)
      }
    }
  }
}

```

Source : Shutterstock / jivacore



Source : Shutterstock / Sinisa Botas

Il y a parfois de bonnes raisons d'utiliser une carte prête à l'emploi, mais les choses peuvent se gâter lorsqu'un fabricant recourt à ce genre de conception modulaire « en toile d'araignée ». L'an passé, on m'a demandé de jeter un œil à un petit appareil médical dont le connecteur USB était défectueux. Après être (enfin) parvenue à ouvrir le boîtier, j'ai découvert (à ma grande surprise) qu'il contenait un clone bon marché d'Arduino Nano, lui-même flanqué d'une autre petite carte bon marché à capteur de pression I²C. Les fils qui les reliaient étaient fixés avec de la colle chaude (que David Jones, d'EEVblog, appelle de façon imagée de la « morve chaude » [5]). L'appli n'avait rien de formidable non plus, pourtant ce truc avait coûté plusieurs centaines d'euros. Pure recherche de profit ? Manque de compétences ? Les deux ? Je laisse vos blocs de réflexion agir. ►

VF : Hervé Moreau — 230034-04

Note de la rédaction : Ilse est électronicienne de profession. Nous apprécions son franc-parler et respectons son point de vue, mais celui exprimé ici ne doit nullement vous décourager de partager vos projets sur notre plateforme Elektor Labs, qu'ils soient faits ou non de « blocs de construction ». Ébauche de schéma, prototype ou simple idée, vos projets sont les bienvenus à toutes les étapes !

LIENS

- [1] Wikipedia, Langage de programmation : https://fr.wikipedia.org/wiki/Langage_de_programmation
- [2] Elektor Labs : <https://elektormagazine.fr/labs>
- [3] OpenAI: ChatGPT research release: <https://chat.openai.com>
- [4] Modèles Codex d'OpenAI : <https://help.openai.com/en/articles/6195637-getting-started-with-codex>
- [5] EEVblog: Sony Boombox REPAIR and Teardown : <https://youtu.be/8ToVCAhLg8A?t=487>

enregistreur de données énergétiques

mesurer et enregistrer la consommation d'énergie



Georg Luber (Allemagne)

Il existe potentiellement de nombreuses applications pour lesquelles il peut être utile d'enregistrer en continu la production ou la consommation d'énergie électrique. Il peut s'agir par exemple de la consommation d'énergie de certains circuits domestiques, de l'énergie produite par une centrale solaire de balcon, ou encore d'une installation photovoltaïque plus importante. Il existe évidemment des solutions d'enregistrement prêtes à l'emploi, mais vous pouvez également construire vous-même un appareil approprié.

L'enregistreur présenté ici utilise un dispositif de mesure doté d'une sortie conforme à l'interface So pour enregistrer la consommation d'énergie électrique. Des appareils de mesure adaptés existent

Caractéristiques

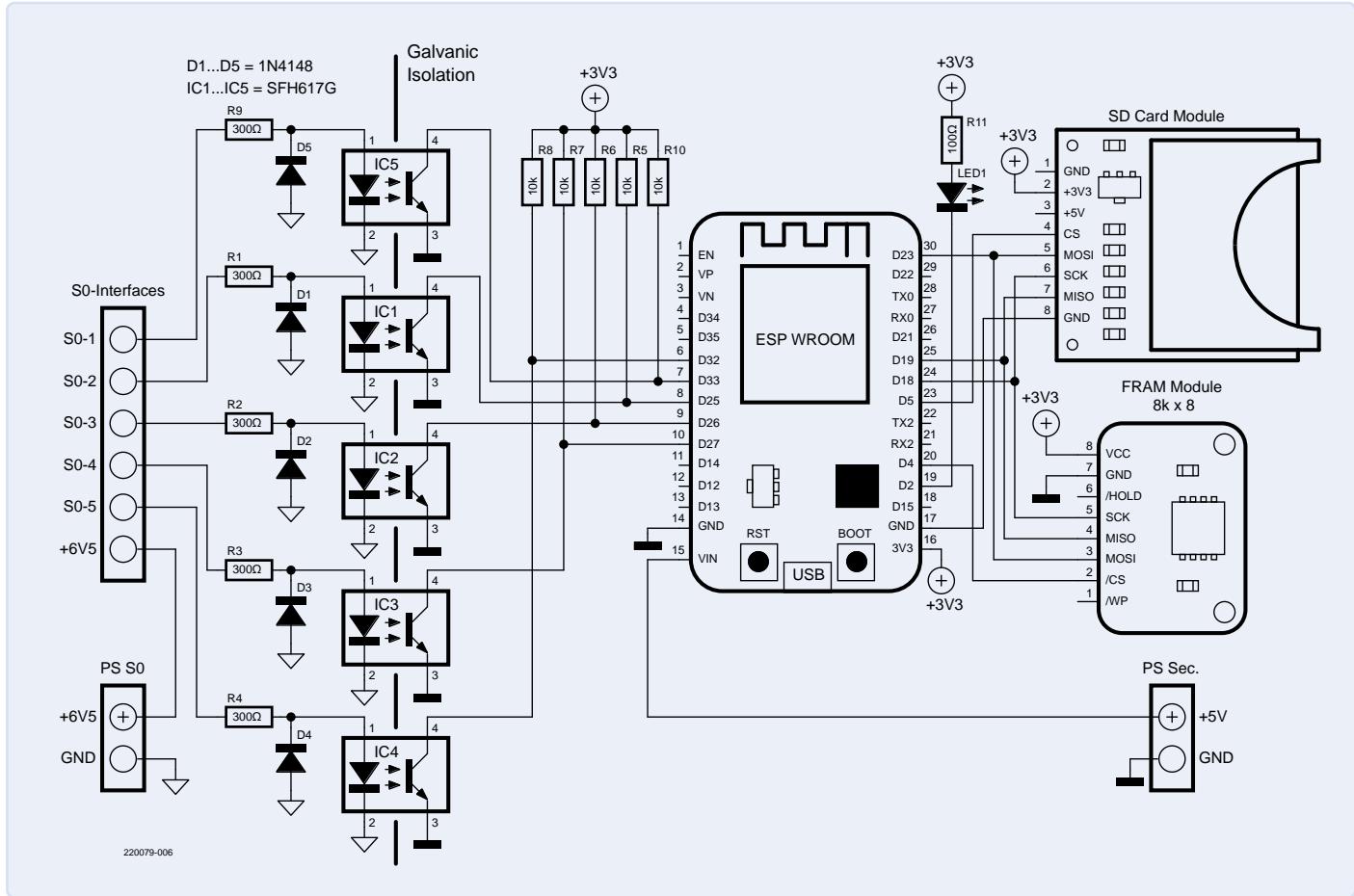
- Précision et sécurité avec un appareil de mesure doté d'une interface So et d'une isolation électrique.
- Acquisition configurable pour intégrer jusqu'à cinq appareils de mesure.
- Mise en mémoire tampon des données dans le module FRAM pour protéger la carte SD.
- Enregistrement sur la carte SD au format CSV.
- Résolution temporelle : 5 minutes.
- Un fichier distinct est enregistré chaque jour.
- Affichage des données individuelles et agrégées (mesures du jour et du jour précédent en kWh).
- Affichage de l'intensité du champ Wi-Fi et du nombre de réinitialisations du µC (compteur de démarrage).
- Téléchargement des données en mode FTP.
- Mise à jour en mode OTA : possibilité de mise à jour par le Wi-Fi.
- Environnement de développement : VSCode avec PlatformIO.

pour moins de 20 € (par exemple, sur Amazon, eBay ou directement auprès de fournisseurs situés en Extrême-Orient). Vous pouvez ainsi mesurer la consommation d'énergie des circuits électriques raccordés au réseau ou le rendement des systèmes photovoltaïques, puis « consigner » ou enregistrer les valeurs à l'aide de l'électronique décrite ici.

Après tout, pourquoi les microcontrôleurs ont-ils été inventés ? À peu de chose près pour ce type d'application. Avant l'ère de l'informatique, l'enregistrement de la consommation ou de la production d'énergie au fil du temps pour un usage domestique aurait été difficile et pratiquement impossible. Grâce aux faibles coûts des microcontrôleurs et à leurs exceptionnelles capacités, non seulement l'enregistrement numérique ne pose plus de problème aujourd'hui, mais même des démarches sophistiquées, par exemple une connexion Wi-Fi, deviennent possibles. Les possibilités de l'enregistreur de données énergétiques sont décrites dans l'encadré **Fonctionnalités**. La **figure 1** donne un aperçu de la solution finalisée. Elle montre comment combiner un dispositif de mesure et un enregistreur, notamment les blocs d'alimentation, installés dans un boîtier de distribution en plastique.



Figure 1. L'enregistreur de données énergétiques finalisé avec l'appareil de mesure et le bloc d'alimentation, installé dans un petit boîtier en matière plastique.



Circuit

Grâce au microcontrôleur, le circuit de l'enregistreur de données énergétiques (**figure 2**) est assez simple. Sur le côté gauche, se trouvent un certain nombre d'optocoupleurs servant à l'isolation électrique, par le biais desquels les données de l'interface So de l'appareil de mesure sont envoyées aux entrées correspondantes du microcontrôleur. Celles-ci sont contrôlées par des interruptions pour qu'aucune impulsion ne soit perdue. J'ai utilisé un module ESP32 bon marché car il possède une puissance de calcul suffisante et une interface Wi-Fi. Sur le côté droit, se trouvent deux modules de mémoire — un emplacement pour une carte SD servant au stockage de masse et un module FRAM supplémentaire, qui met temporairement en mémoire tampon les données correspondant à une période de 5 minutes pour réduire le nombre de cycles d'écriture sur la carte SD.

Le circuit étant très simple, je l'ai construit sur une platine d'essai. Comme vous pouvez le voir sur les deux blocs bleu clair, deux alimentations séparées sont prévues pour des raisons de sécurité. Pour les interfaces So de gauche, une tension de 6,5 V est suffisante. Le reste du circuit reçoit sa tension d'une alimentation en 5 V (à droite). Une capacité de charge de 0,5 A chacun est suffisante. Pour des raisons d'isolation électrique, les deux lignes de masse GND des alimentations ne doivent en aucun cas être connectées. En outre, au-dessous des optocoupleurs, et entre leurs entrées et leurs sorties, tout le cuivre doit être

retiré sur une longueur d'au moins 4 mm. La **figure 3** montre la présentation obtenue.

L'isolation électrique au moyen de deux alimentations séparées permet également de connecter l'ESP32 à l'interface USB d'un PC (par exemple, pour permettre l'envoi de futures mises à jour au contrôleur installé). Au lieu d'utiliser deux alimentations, une solution mono-alimentation associée à une charge de courant plus élevée, avec un convertisseur continu/continu isolant et un régulateur de tension, aurait également été possible, mais sans que cela rende les choses plus faciles ou moins chères.

Figure 2. Le circuit de l'enregistreur de données énergétiques est assez simple.

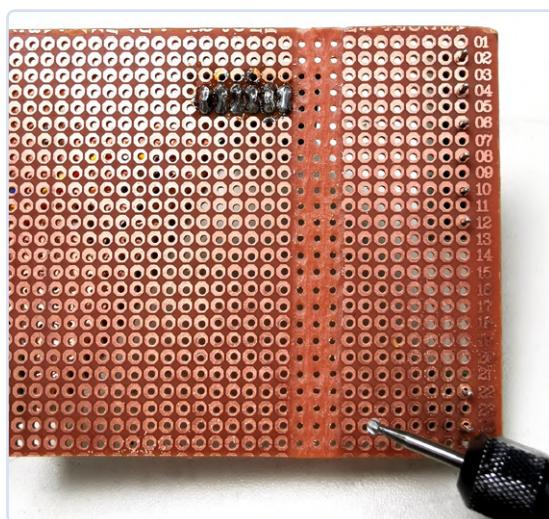


Figure 3. Le cuivre de la face inférieure de la platine d'essai, entre les entrées et les sorties des optocoupleurs, a été enlevé avec un petit cutter.

	A	B	C	D	E
1	/2021/07/31.csv - 00:00				
2	426572	415390	132309	129324	218375
3	05:55				
4	>				
5	426572	415390	132309	129324	218375
6	426574	415391	132309	129324	218375
7	426575	415393	132309	129324	218375
8	426575	415393	132309	129324	218375
9	426576	415394	132309	129325	218375
10	426577	415396	132310	129326	218377
11	426579	415398	132310	129326	218378
12	426583	415401	132310	129326	218380
13	426587	415405	132310	129326	218382

Figure 4. La structure des données devient visible après importation d'un fichier CSV dans une feuille de calcul.

Pour éviter cette situation, un tampon stable est prévu. Le module FRAM externe utilisé ici n'a qu'une capacité de stockage de 8 Ko, mais il est largement suffisant pour collecter un grand nombre de valeurs. Le principal avantage d'une mémoire FRAM est qu'elle autorise au moins 10^{10} écritures, selon le fabricant - typiquement, ce type de mémoire peut même accepter plusieurs millions de milliards de cycles d'écriture. Toutes les cinq minutes, les données collectées dans la mémoire FRAM sont transférées vers la carte SD. Ce mouvement correspond à un peu plus de 100 000 opérations d'écriture par an, et comme une carte SD comporte de nombreuses cellules de mémoire, elle peut être utilisée pendant plusieurs années sans problème.

De plus, le nombre de redémarrages de l'enregistreur est stocké dans la mémoire FRAM, ce qui vous permet de vérifier à tout moment la fréquence de déclenchement des réinitialisations. De plus, chaque jour, un nouveau fichier est créé sur le support SD, et les données du jour y sont enregistrées. Pour économiser de l'espace mémoire, l'enregistrement n'est lancé que si l'une des valeurs du compteur a changé depuis minuit. Si l'enregistreur de données énergétiques sert à consigner la consommation électrique, cette fonction est en fait superflue. En revanche, s'il s'agit de surveiller la production d'énergie d'une installation solaire, il vaut mieux que l'enregistrement ne commence qu'après le lever du soleil (c'est-à-dire au moment où commence la production électrique du jour).

Stockage et transmission des données

Comme mentionné précédemment, l'utilisation de deux modules de mémoire n'est pas un luxe. Les interfaces So peuvent parfois délivrer plusieurs impulsions par seconde. Si les valeurs mesurées étaient recueillies dans la mémoire interne du microcontrôleur, les données pourraient être perdues lors d'une réinitialisation. D'autre part, les enregistrer immédiatement dans une carte SD réduirait massivement la durée de vie de la carte. En effet, avec seulement une valeur par seconde, il se produirait 31,5 millions de cycles d'écriture par an. Or, les cellules de mémoire d'une carte SD atteignent leur fin de vie au-delà de seulement 1000 à 3000 cycles d'écriture. Même une carte à haute capacité présenterait donc très certainement des défauts dans la première année.

Name	Größe	Geändert
2021	9 Elemente	07.01.22 21:06
2022	11 Elemente	02.11.22 22:25
2023	1 Element	08.01.23 10:06
01	17 Elemente	17.01.23 17:21
01.CSV	7,8 kB	08.01.23 10:07
02.CSV	7,8 kB	08.01.23 10:07
03.CSV	7,7 kB	08.01.23 10:07
04.CSV	7,8 kB	08.01.23 10:07

Structure des données, et autres

La **figure 4** montre une section d'une feuille de calcul dans laquelle les données ont été importées au format CSV. La cellule A1 contient la date et l'heure de création du fichier. La ligne 2 contient ensuite les dernières valeurs de la journée précédente servant de nouveau point de départ. Le premier changement d'une valeur mesurée a eu lieu à 05h55 (cellule A3). À partir de la ligne 5, les nouvelles données du jour suivent. Les colonnes contiennent les données des différents appareils de mesure. Chaque ligne comporte un écart de temps de cinq minutes par rapport à la ligne suivante. Les données sont enregistrées pour la journée jusqu'à minuit. Un nouveau fichier est créé chaque jour. La quantité d'énergie peut alors être calculée à partir de la différence entre les mesures relevées à l'aide de la valeur associée à l'impulsion (par exemple, 0,5 Wh/impulsion). Le format CSV est économique et permet l'importation dans n'importe quel tableur comme Excel, OpenOffice/LibreOffice Calc

Figure 5. La structure du dossier, avec une répartition par année, mois et jour.

ou Numbers sur Mac. Il est ensuite possible de traiter les données comme vous le souhaitez.

Sur la carte SD, les données sont stockées selon une structure de dossiers temporels. Au niveau supérieur, il y a des dossiers pour chaque année, chacun d'entre eux contenant des sous-dossiers pour les mois, qui contiennent à leur tour les fichiers individuels de chaque jour. La **figure 5** représente cette structure. Dans cet exemple, les mesures issues de cinq appareils sont enregistrées au format entier (voir la **figure 4**). Grâce au format CSV, la quantité de mémoire nécessaire est relativement modeste. Avec seulement quelques kilo-octets par appareil de mesure et par jour. Le transfert des données de l'enregistreur vers un PC se fait au moyen du protocole FTP. Les clients FTP habituels, comme FileZilla, sont adaptés à cette opération. Comme l'enregistreur est intégré au réseau domestique grâce au Wi-Fi, cette méthode est probablement la plus simple.

J'ai créé le code source de l'ESP32 en utilisant l'IDE PlatformIO pour VSCode [1]. Le code est disponible en téléchargement gratuit sur la page web de cet article [2].

Les données suivantes doivent être personnalisées dans le fichier *main.cpp* : à partir de la ligne 30, le SSID et le mot de passe du réseau Wi-Fi doivent être saisis. Le nom d'utilisateur et le mot de passe pour le FTP sont spécifiés à la ligne 813. Par défaut, « esp32 » sert de nom d'utilisateur et de mot de passe. Bien entendu, le client FTP a également besoin de l'adresse IP de l'enregistreur de données énergétiques. Cette adresse est fournie par l'interface USB après réinitialisation. L'idéal serait de configurer votre routeur de manière à ce que l'enregistreur de données ait toujours la même adresse IP.

Interface Web intégrée

Les données actuelles ainsi que la valeur totale de la journée et de la journée précédente sont affichées à l'aide du serveur web intégré. Le serveur héberge deux pages. Une page affiche la puissance actuelle par appareil de mesure ainsi que le total de tous les appareils connectés. En outre, l'énergie totale du jour et de la veille est affichée en kWh. L'autre page indique l'intensité du champ du réseau Wi-Fi et le nombre de redémarrages de l'enregistreur.

Les **figures 6 et 7** sont des captures d'écran d'un smartphone. Comme vous le remarquez immédiatement, je n'utilise pas l'enregistreur pour mesurer la consommation, mais pour surveiller mon installation photovoltaïque, qui compte cinq onduleurs.

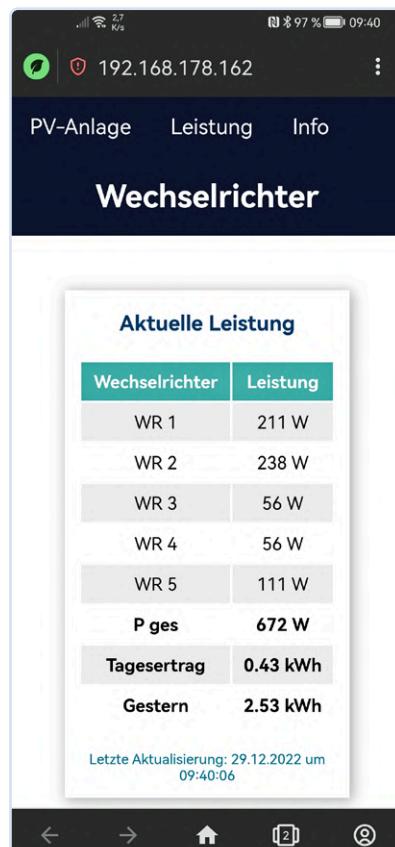


Figure 6. L'interface web intégrée affiche les résultats des mesures.



Figure 7. Si vous ajoutez la mention « / about » à l'adresse IP, ou si vous cliquez sur Info, vous obtiendrez ces informations.

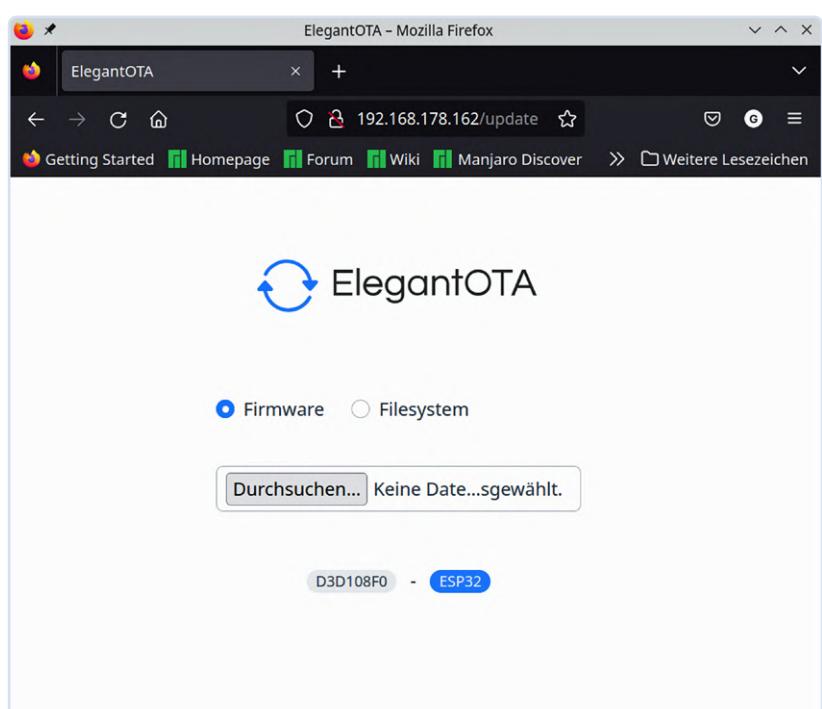


Figure 8. Les mises à jour du micrologiciel et de la structure des données sont possibles grâce au Wi-Fi.



Figure 9. Quatre dispositifs de mesure de type DDS5188 en fonctionnement.

C'est pourquoi j'ai adapté les étiquettes des différentes valeurs à l'utilisation que j'en fais. Cependant, vous pouvez facilement modifier cela dans le logiciel si nécessaire. Lorsque vous vous connectez à l'adresse IP dans le navigateur, vous obtenez l'affichage représenté sur la **figure 6**. En ajoutant « /about » à l'adresse ou en cliquant sur le bouton Info, l'affichage illustré par la **figure 7** apparaît.

Une remarque supplémentaire concernant la connexion Wi-Fi : si vous installez l'enregistreur dans un boîtier métallique, vous aurez très certainement des problèmes de réception.

Mise à jour en mode Wi-Fi

Une fois l'enregistreur de données énergétiques intégré dans un boîtier et installé, vous ne souhaiterez probablement pas démonter à nouveau l'électronique en cas de changement de logiciel. Vous pouvez donc mettre à jour l'enregistreur via le Wi-Fi (mode OTA, ou Over The Air). Il est ainsi possible, si nécessaire, de charger de nouveau le logiciel complet ainsi que le système de fichiers - les pages Web avec JavaScript et le fichier CSS.

Pour ce faire, il vous suffit de saisir l'adresse IP en ajoutant le suffixe « /update ». La **figure 8** illustre ma configuration. J'ai réalisé la mise à jour en mode OTA grâce à la bibliothèque d'Ayush Sharma [3].

Un circuit simple

Le circuit est si simple qu'une liste de pièces n'est pas nécessaire. Outre la carte de développement ESP32, que l'on trouve partout, les composants importants sont un emplacement pour support SD, que l'on peut trouver sur des cartes de liaison bon marché, et le module FRAM, également disponible sur des cartes de liaison. Ces cartes sont disponibles avec différentes implantations de broches. Vous ne devez donc pas

prêter attention aux numéros de broches, mais plutôt à leurs désignations lors de la connexion. Concernant le module de carte SD, MOSI peut être connecté à DI ou SI, et MISO à DO ou SO. Concernant les appareils de mesure, n'importe quel modèle bon marché pour montage sur rail DIN avec une interface SO peut convenir. Le modèle DDS5188 type I que j'ai utilisé (**figure 9**) est facilement disponible et très bon marché.

S'il va de soi que vous devez faire preuve de la prudence nécessaire lorsque vous travaillez sur le secteur, une remarque s'impose : un certain nombre de lois nationales s'appliquent aux interventions sur le tableau contenant les fusibles. En Allemagne, vous devez être enregistré auprès de l'exploitant du réseau en tant qu'électricien qualifié. De même, les interventions sur les sous-répartiteurs externes connectés au tableau contenant les fusibles ne sont autorisées que pour les électriciens qualifiés formés et possédant les connaissances appropriées. ◀

VF : Pascal Godart — 220079-04

À propos de l'auteur

Électricien de formation, Georg Luber a étudié l'électrotechnique et travaillé de nombreuses années dans les installations et la sécurité électriques. Il est intervenu au sein de différents comités de normalisation (DKE/VDE, CENELEC, IEC et ISO) aussi bien au niveau national qu'international. Il s'est également intéressé à l'automatisation des bâtiments, en particulier avec le protocole et les systèmes KNX. Georg Luber est intervenu dans le développement de logiciels et de projets électroniques dans ces domaines.

Des questions, des commentaires ?

Contactez Elektor à l'adresse redaction@elektor.fr.



Produits

➤ **ESP32-DevKitC-32D (SKU 18701)**
www.elektor.fr/18701

➤ **Compteur à pince PeakTech 4350 (SKU 18161)**
www.elektor.fr/18161

LIENS

- [1] PlatformIO pour VSCode : <https://platformio.org/platformio-ide>
- [2] Page web de l'article : <https://www.elektrormagazine.fr/220079-04>
- [3] Bibliothèque OTA d'Ayush Sharma : <https://github.com/ayushsharma82/ElegantOTA>
- [4] Projets ESP32 : <https://randomnerdtutorials.com/projects-esp32>

assemblage du kit Rover M.A.R.S. de 4tronix

Clemens Valens (Elektor)

Inspiré des rovers martiens Curiosity et Persévérance de la NASA, le rover M.A.R.S. de 4tronix est un véhicule autonome conçu pour se déplacer sur des terrains accidentés, mais sur Terre. En effet, dans ce cas, M.A.R.S. ne fait pas référence à la planète mais est un acronyme pour *Mobile Autonomous Robotic System*.

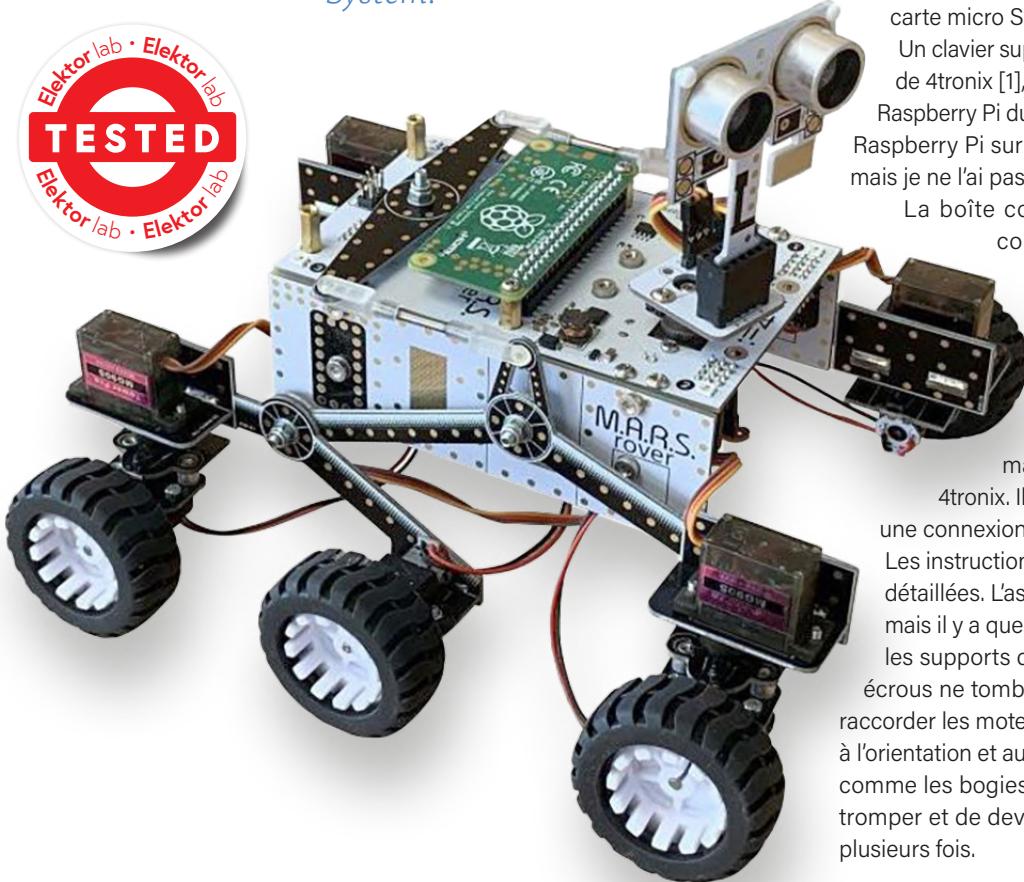
Le rover M.A.R.S. est livré sous forme d'un kit de pièces détachées, mais il ne nécessite aucune soudure. Le kit est complet, y compris les outils, j'ai cependant préféré utiliser mes propres tournevis à la place, car ils conviennent mieux.

Il existe deux versions du kit : Raspberry Pi Zero et BBC micro:bit. J'ai construit et essayé la version Raspberry Pi. Notez que le Raspberry Pi Zero ou le BBC micro:bit ne sont pas inclus, tout comme les quatre piles AA. Pour le Raspberry Pi, vous aurez également besoin d'une carte micro SD.

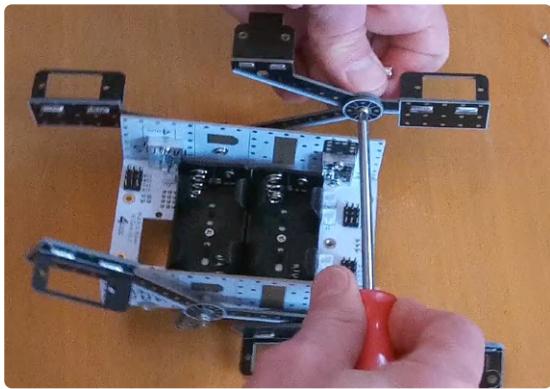
Un clavier supplémentaire est disponible sur le site de 4tronix [1], mais je ne l'ai pas essayé. La version Raspberry Pi du rover permet de monter une caméra Raspberry Pi sur le mât du transducteur à ultrasons, mais je ne l'ai pas essayée non plus.

La boîte contenant le kit est étonnamment compacte (18 cm × 16 cm × 8 cm), mais elle est assez lourde (600 g) et remplie à ras bord de sachets en plastique contenant les pièces détachées. Aucun manuel n'est inclus. Des instructions détaillées pour l'assemblage et la programmation se trouvent sur le site Web de 4tronix. Il vous faudra donc un ordinateur avec une connexion Internet pour assembler le rover.

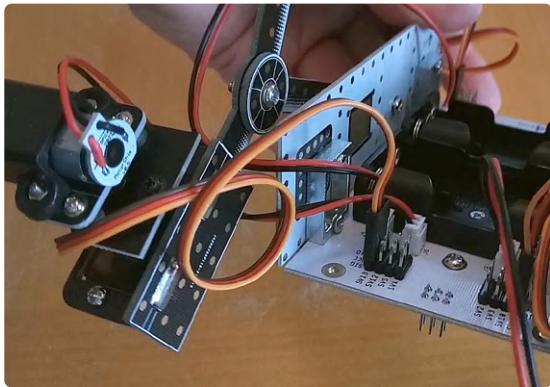
Les instructions d'assemblage sont assez claires et détaillées. L'assemblage du rover n'est pas difficile, mais il y a quelques parties délicates comme visser les supports de montage du moteur sans que les écrous ne tombent, fixer certains contre-écrous ou raccorder les moteurs. Faites cependant bien attention à l'orientation et au positionnement de certaines pièces comme les bogies et les servos, car il est facile de se tromper et de devoir recommencer. Cela m'est arrivé plusieurs fois.



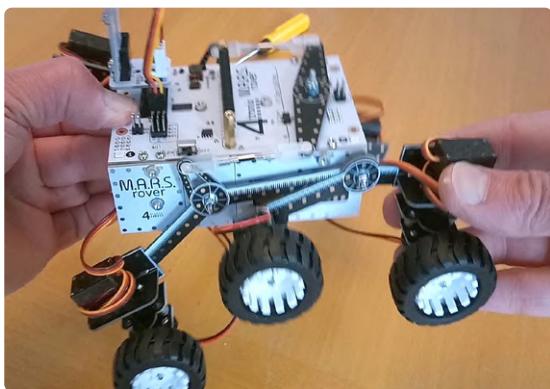
► Figure 1. Utilisez vos propres tournevis.



► Figure 2. Tous les connecteurs des servos sont orientés de la même manière. Ils sont repérés clairement, et les fils des servos ont un code de couleur. La masse (GND) en brun, le signal (SIG) en jaune.



► Figure 3. Le bras de suspension doit tourner avec un minimum de frottement.



Quelques conseils d'assemblage

L'assemblage du rover m'a pris deux heures environ à cause de quelques erreurs que j'ai faites.

Voici quelques conseils :

- Utilisez de vrais tournevis (**figure 1**).
- Il y a trois vis de 18 mm de long qui traversent les circuits imprimés. Veuillez bien les serrer avant de fixer la pièce de l'autre côté à l'aide d'un contre-écrou, car elles pourraient sinon se desserrer lorsque vous devrez ajuster l'écrou pour que les pièces qui y sont fixées puissent tourner librement.
- Tous les connecteurs des servos sont orientés de la même manière. Ils sont repérés clairement, et les fils des servos ont un code de couleur (**figure 2**). La masse (GND) en brun, le signal (SIG) en jaune. Il est pratique de noter par écrit le numéro de chaque servo (SVx) pour l'utilisation du programme *servoTest*.
- Le mât du transducteur à ultrasons peut frotter sur les fils des servos si ces fils ne sont pas suffisamment rabattus vers le bas.
- Après avoir raccordé les servos, vous pouvez enrouler leurs fils autour des corps des servos afin de mettre un peu d'ordre. En revanche, laissez libres les fils des moteurs des roues afin d'éviter qu'ils ne bloquent les mouvements des bras de suspension (**figure 3**).
- Il n'est pas nécessaire de couper les extrémités des bras de servos qui dépassent, cela permet uniquement d'obtenir une apparence plus propre.

Cerveau et alimentation

Après avoir assemblé le rover, j'ai monté un Raspberry Pi Zero 2 W sur le dessus de celui-ci. J'ai utilisé une carte microSD comportant le système d'exploitation Raspberry Pi OS Buster installé. Je l'avais utilisé avec un autre projet, ce qui fait que le Raspberry Pi pouvait déjà se connecter à mon réseau wifi et que l'accès SSH était de même activé. Pour économiser les piles, j'ai alimenté le Raspberry Pi via USB pour le téléchargement et l'installation du logiciel permettant de faire fonctionner le rover. De nouveau, les instructions sur la manière de procéder sont claires et détaillées, mais elles ne sont disponibles qu'en ligne [2]. J'ai aussi essayé un Raspberry Pi Zero à la place d'un Zero 2 W et je me suis aperçu qu'il fonctionnait aussi bien. L'avantage d'utiliser un Zero 2 W est, bien entendu, qu'il démarre bien plus rapidement.

Le Raspberry Pi n'alimente pas le reste du rover, il vous faut des piles pour cela. Si donc, en essayant les exemples, vous voyez un message d'erreur concernant le bus I²C, et que vous êtes certain de l'avoir activé sur le Raspberry Pi, vérifiez le commutateur *On/Off* et vos piles. Une LED bleue indique si le rover est alimenté.



J'ai mesuré une consommation de l'ordre de 400 à 500 mA lorsque le rover se déplace avec le script *motorTest*. Elle est de l'ordre de 200 mA au repos.

Programmer le rover

Lors du développement de vos propres programmes pour le rover [3], il est probablement préférable de le faire autant que possible avec le Raspberry Pi alimenté par USB afin d'économiser les piles, et de n'alimenter le rover que pour faire les tests.

J'ai essayé tous les exemples fournis par 4tronix [2] et j'ai trouvé que tout fonctionnait sans aucun problème. Je n'ai pas fait du tout de programmation moi-même. Notez que pour une raison quelconque, l'utilisation des LED RVB du rover nécessite des priviléges d'administrateur, tout script les utilisant doit donc être exécuté avec *sudo*. Faire rouler le rover par-dessus les obstacles est assez impressionnant (**figure 4**). Il n'avance pas très vite, mais les bras de suspension lui permettent de franchir les obstacles comme s'ils n'existaient pas.

Presque rien ne peut l'arrêter

Le kit de rover M.A.R.S. de 4tronix et un kit de haute qualité qui, une fois assemblé, a une allure assez sympa grâce à ses impressions décoratives et ses éléments en circuits imprimés. Il mesure 20 cm × 18 cm × 16 cm, il est donc un peu plus gros que la boîte dans laquelle il est livré. L'assemblage du robot est assez facile, car tout s'adapte parfaitement (ce qui est assez rare) et les instructions sont claires et détaillées. Son utilisation est facile aussi, et les exemples et les programmes de test s'installent et fonctionnent sans aucun problème. Une fois assemblé et le logiciel installé, le rover est prêt à se déplacer et pratiquement rien ne peut l'arrêter.

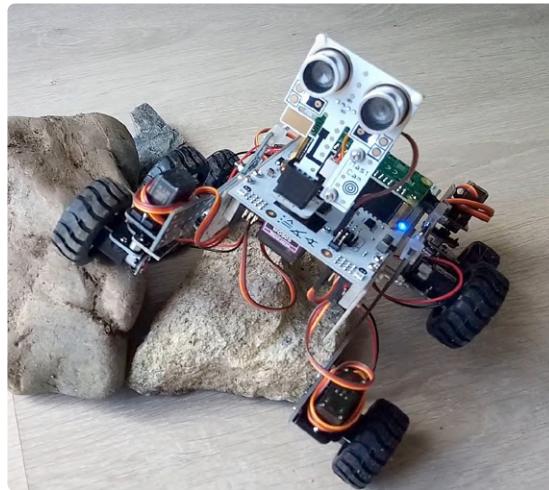


Figure 4. Une fois assemblé, pratiquement rien ne peut arrêter le rover M.A.R.S.

Certaines connaissances en programmation sont bien entendu nécessaires, mais c'est bien pour ça que vous en êtes procuré, n'est-ce pas ?

Étendre le Kit

Le circuit imprimé principal possède quatre trous de fixation supplémentaires qui peuvent être utilisés pour fixer des extensions, et il y a de nombreuses connexions disponibles pour ajouter davantage de servos. (À propos, il y avait un servo supplémentaire dans le sachet de pièces détachées de mon kit).

Je n'ai pas trouvé de schéma, je ne sais donc pas si les connecteurs des servos peuvent être utilisés pour autre chose. Il y a aussi deux connecteurs d'extension I²C disponibles auxquels vous pourrez raccorder des capteurs, par exemple. ▶

220107-04



Produits

- ▶ **4tronix M.A.R.S. rover (SKU 19996)**
www.elektor.fr/19996

LIENS

- [1] Assemblage du rover M.A.R.S. : <https://4tronix.co.uk/blog/?p=2112>
- [2] Programmer le rover M.A.R.S. avec Raspberry Pi Zero (inc v2) : <https://4tronix.co.uk/blog/?p=2409>
- [3] MARS-Rover [GitHub] : <https://github.com/4tronix/MARS-Rover>
- [4] Assemblage du kit Rover M.A.R.S. de 4tronix — revue [vidéo] : <https://youtu.be/Np8ZQQd85oc>

disque de stationnement avec affichage e-papier

une version informatisée innovante

Antonello Della Pia (Italie)

Un disque de stationnement est un accessoire présent dans toutes les voitures. Ce projet en présente une version numérique utilisant un afficheur e-papier, avec des caractéristiques particulières, telles qu'un bouton unique pour le réglage de l'heure d'arrivée, un message en quatre langues au choix et l'affichage à la demande de l'heure et de la date courantes, de la température ambiante et du niveau de la batterie.

L'intégration de la technologie électronique dans les véhicules a atteint des niveaux extrêmement élevés. Un nombre toujours croissant de capteurs performants, d'interfaces numériques, de microprocesseurs et de logiciels associés assurent désormais une gestion optimale de pratiquement tous les processus physiques. Dans l'habitacle de ce que l'on peut désormais considérer comme des « ordinateurs sur roues », il reste quand même un instrument indispensable, souvent en carton et à commande manuelle : le disque horaire, obligatoire pour indiquer l'heure de début du stationnement dans les zones réglementées. Au fil des décennies, cet accessoire est resté pratiquement inchangé – en carton, en plastique ou autre matériau plus noble – et ce n'est que récemment que des modèles numériques sont apparus sur le marché. Celui proposé dans l'article utilise un afficheur moderne e-papier et présente quelques caractéristiques particulières, comme le réglage de l'heure d'arrivée à l'aide d'un seul bouton, un message en quatre

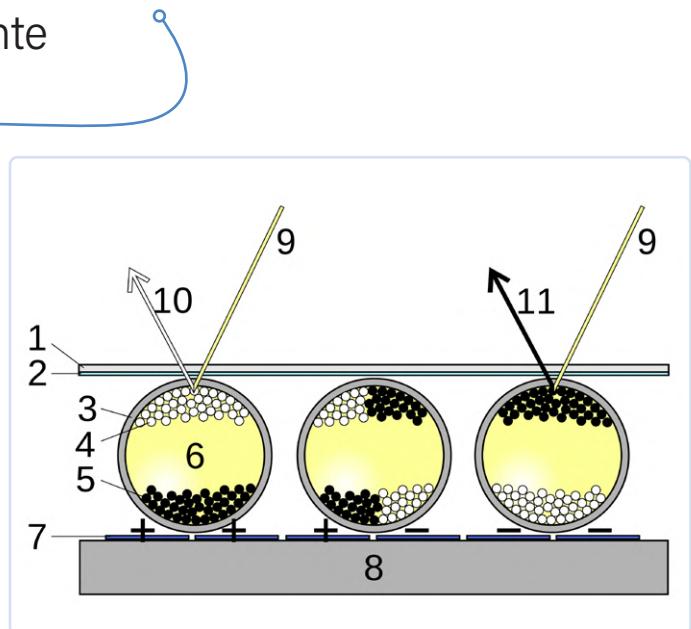


Figure 1. Principe de base de fonctionnement de l'afficheur e-papier. (1 couche supérieure, 2 couche transparente d'électrodes, 3 micro-capsules transparentes, 4 pigments blancs chargés (+), 5 pigments noirs chargés (-), 6 fluide transparent, 7 couche d'électrode de pixel, 8 couche inférieure, 9 lumière, 10 blanc, 11 noir). (Source : https://en.wikipedia.org/wiki/E_Ink)

langues au choix, l'affichage à la demande de l'heure et de la date courantes, de la température ambiante et du niveau de la batterie.

L'afficheur e-papier

Invention relativement récente (1996), la technologie e-ink (encre électrophorétique), généralement appelée e-papier [1], doit son succès avant tout à son utilisation dans les liseuses de livres électroniques, ces dispositifs portables qui offrent une alternative électrique aux livres traditionnels, grâce à une expérience de lecture semblable à celle du papier et à une visibilité parfaite, même par forte luminosité ambiante. Toutefois, la caractéristique unique qui a conduit à la diffusion de cette technologie dans d'autres domaines est qu'il n'y a pas de consommation d'énergie pour maintenir l'affichage des informations, même pendant une longue période, mais seulement pour la mise à jour de l'afficheur. Les applications typiques, de plus en plus populaires dans les points de vente, sont

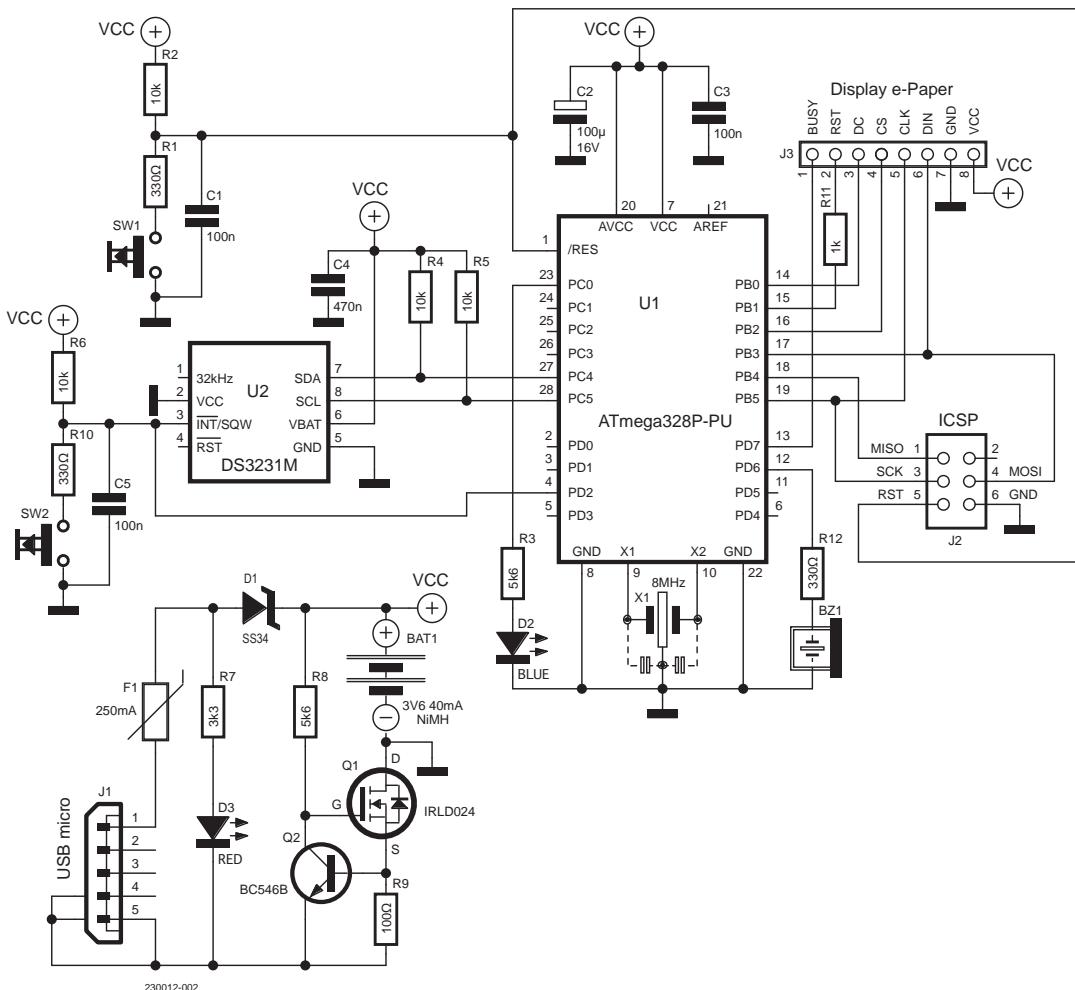


Figure 2. Schéma du circuit.

les étiquettes électroniques d'articles et de prix, souvent difficiles à distinguer des étiquettes en papier, faciles à mettre à jour quand c'est nécessaire, même à distance grâce aux technologies sans fil. La **figure 1** va nous aider à mieux comprendre le fonctionnement de l'encre électronique.

Dans la version la plus simple, l'affichage en noir et blanc, des pigments chargés positivement (blanc) et négativement (noir) sont en suspension dans un liquide contenu dans des microsphères représentant des pixels. Grâce à la polarisation créée par un champ électrique approprié, les pigments, attirés par la charge de signe opposé (electrophorèse), se positionnent pour créer des pixels noirs ou blancs, composant l'image désirée. À ce stade, même si le champ électrique est supprimé, les pigments restent en position jusqu'à ce qu'une nouvelle charge soit appliquée. La visibilité, sous un angle particulièrement large, est obtenue par réflexion de la lumière ambiante, ou, en son absence, d'une source lumineuse. Notons toutefois que la lumière directe du soleil peut perturber la mise à jour correcte de l'afficheur. Sur la base de ce principe de fonctionnement, de nombreux types d'afficheurs ont été réalisés, même de grande taille et en couleur, qui restent cependant très chers. Parallèlement, l'offre abordable de modèles plus petits, en noir et blanc, en niveaux de gris ou à gamme de couleurs limitée, proposée par des détaillants spécialisés, a augmenté. En revanche, l'intérêt des bricoleurs pour ces composants n'a pas suivi cette augmentation,

d'après ce que j'ai pu voir sur Internet. À mon avis, cela est dû à différents facteurs critiques, devenus apparents au cours du développement de ce projet, tels que la pléthore de modèles, de versions, de tailles, de pilotes et de gammes de couleurs qui ont envahi le marché, le manque de bibliothèques bien documentées et faciles à mettre en œuvre sur les différentes plates-formes de développement, et des informations et un soutien souvent fragmentaire et insuffisant de la part des fabricants eux-mêmes. Malgré cela, en choisissant parmi les produits les mieux supportés et en appliquant les méthodes des faiseurs, il a été possible d'obtenir un résultat que je crois intéressant. Poursuivons donc avec l'analyse du schéma du circuit (**figure 2**).

Schéma

Un microcontrôleur (MCU) ATmega328P [2] a été choisi pour gérer l'affichage, le même que celui utilisé dans la carte Arduino UNO, étant donné la disponibilité d'une bibliothèque polyvalente spécifique à cette plateforme. Le MCU fonctionne à une fréquence d'horloge de 8 MHz, grâce à un résonateur céramique externe, et se trouve normalement en mode sommeil (SLEEP_MODE_PWR_DOWN), pour ne se réveiller que si l'affichage doit être mis à jour. La puce DS3231M [3], l'horloge en temps réel (RTC), quant à elle, reste active en permanence, capable de maintenir, tant qu'elle est alimentée, la date et l'heure avec une précision de ± 5 ppm ($\pm 0,432$ seconde/jour). Elle communique avec le microcontrôleur via une interface

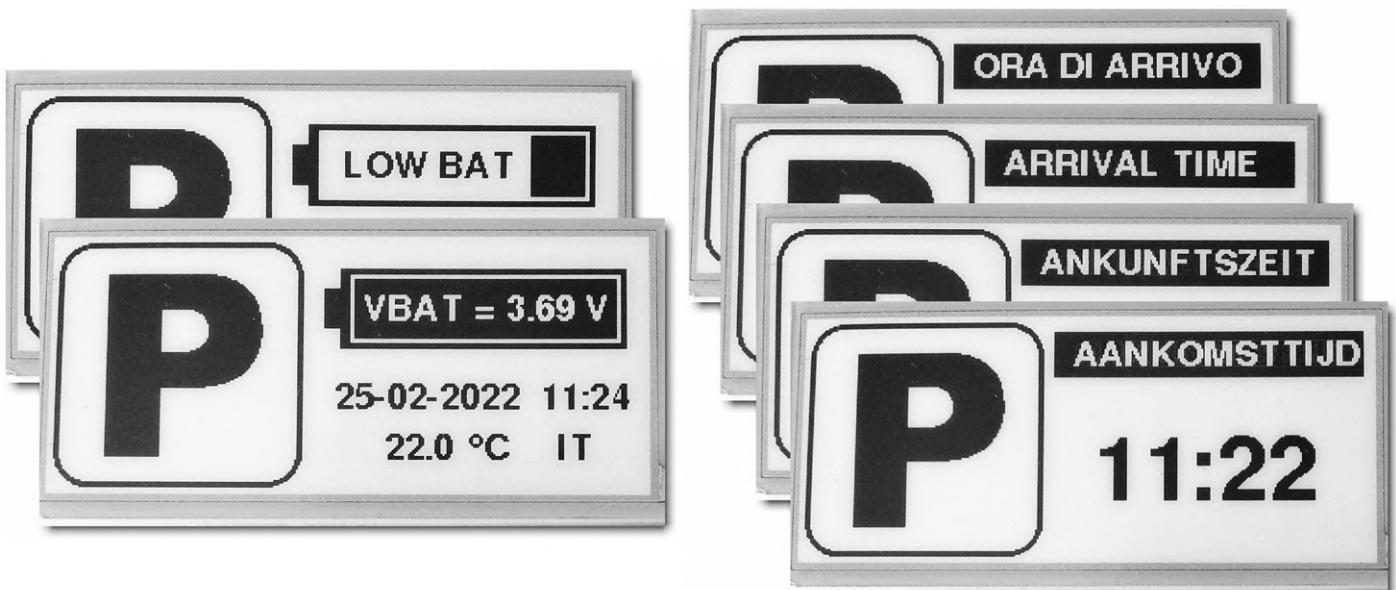


Figure 3. Exemples d'affichages info et parking.

I²C et une bibliothèque dédiée. Parmi ses autres caractéristiques notables, citons un capteur de température, une double alarme et une consommation de courant extrêmement faible, de l'ordre de quelques microampères (μ A) dans la configuration utilisée, à source d'alimentation unique (VBAT). Le bouton SW2 (bleu dans le prototype) « réveille » le MCU, qui peut ainsi recevoir l'heure courante et la transmettre à l'afficheur, ainsi que le message « TIME OF ARRIVAL » (HEURE D'ARRIVÉE) et le logo « P », avant de retourner en mode sommeil. Un court bip du bipper confirme l'action du bouton tandis que la LED bleue D2 reste allumée pendant la durée de l'opération. Le bouton SW1 (rouge dans le prototype) produit une réinitialisation matérielle de l'ATmega328P, suivie immédiatement par l'affichage du logo « P », le symbole et la tension de la batterie, la température ambiante, l'heure et la date courantes et la langue sélectionnée pour le message (figure 3).

Toutes les 24 heures (à l'heure souhaitée, grâce à la fonction d'alarme de la RTC), selon la recommandation du fabricant, un rafraîchissement complet de l'affichage est effectué, qui permet aussi de vérifier l'état de la batterie et le bon fonctionnement de l'ensemble du système. Par conséquent, nous pouvons appeler SW2 le « bouton parking » et SW1 le « bouton info ». Les résistances R1 et R10, avec les condensateurs C1 et C5, contribuent à supprimer tout bruit dû au rebondissement des contacts des boutons-poussoirs, tandis que R2, R4, R5, R6 sont des résistances de rappel. C2, C3 et C4 sont les condensateurs de découplage habituels de l'alimentation des circuits intégrés. R11 limite le courant tiré par la broche de réinitialisation de l'afficheur utilisé. JP2 est le connecteur de l'afficheur, tandis que JP1 permet de connecter un programmeur USBasp pour télécharger le micrologiciel. En raison de la faible consommation de courant, une petite batterie rechargeable Ni-MH de 3,6 V avec une capacité de 40 mAh suffit pour l'alimentation. Le circuit formé par Q1, Q2, R8, R9 est un régulateur de courant constant à faible chute de tension, qui limite le courant de charge à environ 6 mA, une valeur que ce type de batterie peut supporter sans inconvénient même si les 14...16 heures de charge prévues sont dépassées.

Une prise micro-USB permet de connecter un chargeur standard de 5 V, via un fusible de protection réarmable, tandis que la diode D1 empêche l'inversion du courant de la batterie. La LED D3 indique que l'appareil est en charge. Nous reviendrons plus en détail sur l'alimentation électrique un peu plus tard.

Réalisation pratique

Le prototype du disque de stationnement a été réalisé comme d'habitude sur une carte de prototypage (figure 4). Bien sûr, ce n'est qu'une des nombreuses possibilités ; en utilisant des CMS et un circuit imprimé spécial, on pourrait probablement obtenir

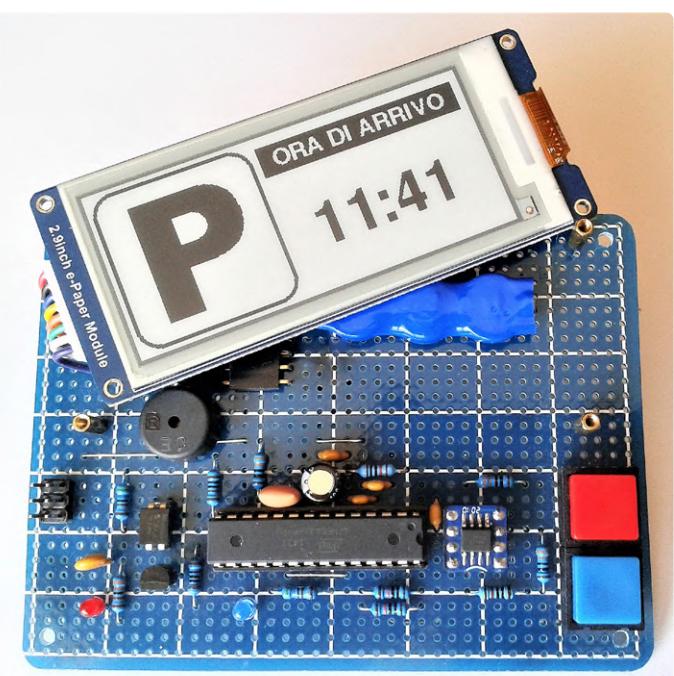


Figure 4. Le prototype, côté composants.



Figure 5. Le module e-papier utilisé dans le projet avec son connecteur « personnalisé ». (Source : Waveshare)

une taille plus petite, légèrement plus grande que l'afficheur. La batterie et le bipuce sont situés sous le module e-papier, tandis que le connecteur à huit broches de ce dernier est réalisé à partir de barrettes mâle et femelle (**figure 6**). Les circuits intégrés sont montés sur embase et un adaptateur SO-8 facile à trouver a été utilisé pour la puce DS3231M. Le choix du boîtier est libre, pourvu qu'il possède une fenêtre transparente !

En ce qui concerne le module e-papier, comme déjà mentionné, il est parfois difficile de s'y retrouver dans l'abondante offre de composants apparemment similaires. Celui utilisé dans le prototype (**figure 5**) est un Waveshare de 2,9 pouces de diagonale, noir et blanc, de résolution 296 x 128 [4]. Il possède un convertisseur de niveau logique intégré et supporte le rafraîchissement partiel, une fonctionnalité indispensable dans ce projet. D'autres modèles peuvent ne pas arriver aux résultats que j'ai obtenus, ou nécessiter des modifications du micrologiciel ou du schéma de câblage. Il faut aussi tenir compte de la plage de température de fonctionnement autorisée, qui, dans ce cas, s'étend de 0 à 50 °C.

La **figure 6** montre le côté soudure du prototype, sur lequel se trouvent également la diode D1 (type SMD) et le connecteur de recharge micro-USB.

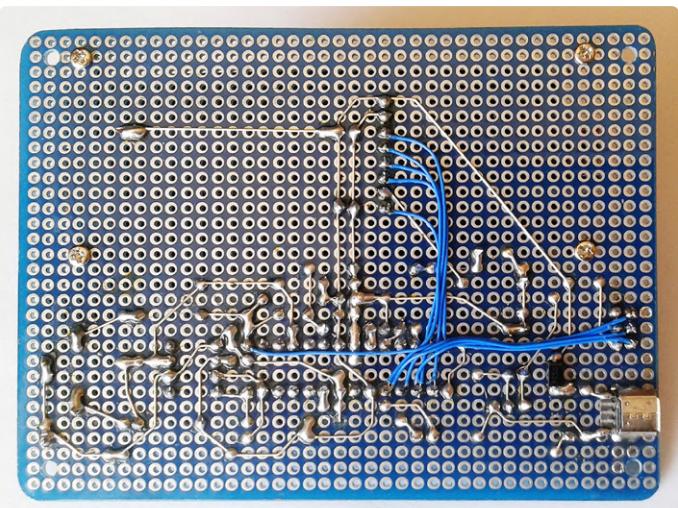


Figure 6. Le prototype côté soudure.

Pour en revenir à l'alimentation électrique, le choix d'une batterie Ni-MH rechargeable, une technologie qui date déjà un peu, est principalement motivé par la recherche d'une sécurité et d'une fiabilité accrues dans les conditions pas toujours optimales d'utilisation dans un véhicule. Par rapport, par exemple, aux batteries Li-Ion, le risque d'explosion, même lointain, est nul et même les performances à basse température sont meilleures. La charge normale est certes plus lente, mais plus facile à gérer. Quoi qu'il en soit, compte tenu des courants minimes en jeu et de la tension de fonctionnement du circuit, qui peut varier sans problème entre 3,3 V et 5,0 V, une solution utilisant trois piles alcalines AAA en série peut également être envisagée, éliminant ainsi la nécessité du circuit de charge et assurant une durée de vie qui peut dépasser celle de la voiture ! Quantifier avec précision la « consommation » des appareils à piles, de plus en plus populaires, qui alternent périodes d'activité et pauses en mode d'économie d'énergie (appelé aussi mode veille, hibernation ou basse énergie) n'est pas simple ; malheureusement, il ne suffit pas d'insérer un multimètre en série avec l'alimentation et de lire le courant. Le calcul de base à effectuer pour obtenir la valeur moyenne du courant consommé, et donc une estimation de l'espérance de vie de la batterie, peut être représenté par la pseudo-formule suivante :

$$I_{AVG} = \frac{[(I_{ON} \cdot T_{ON}) + (I_{SBY} \cdot T_{SBY})]}{(T_{ON} + T_{SBY})}$$

où I_{AVG} représente le courant moyen, I_{ON} le courant en mode actif, I_{SBY} le courant de veille, T_{ON} le temps d'activité et T_{SBY} le temps de veille.

J'ai observé les fluctuations du courant actif à l'oscilloscope (**figure 7**) en détectant la chute de tension aux extrémités d'une résistance de précision de faible valeur en série avec l'alimentation, obtenant, avec une approximation acceptable, la valeur de 10 mA pendant un temps de 7,5 secondes, qui représente la durée d'un cycle d'affichage. Le courant de repos, mesuré à l'aide d'un

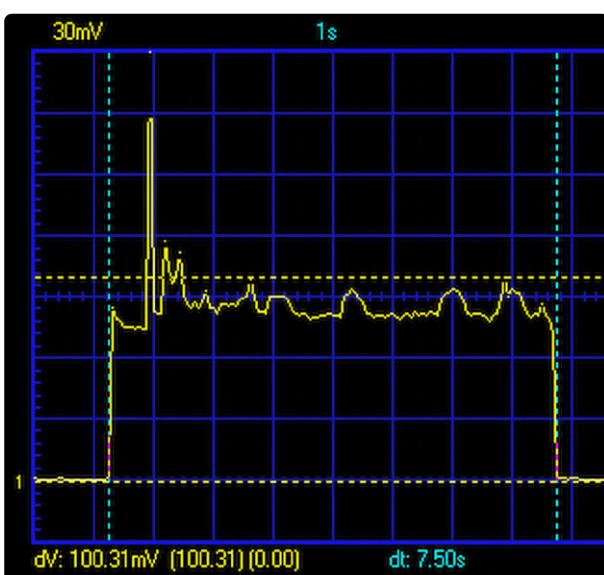


Figure 7. Oscilloscopogramme du courant consommé par le circuit.

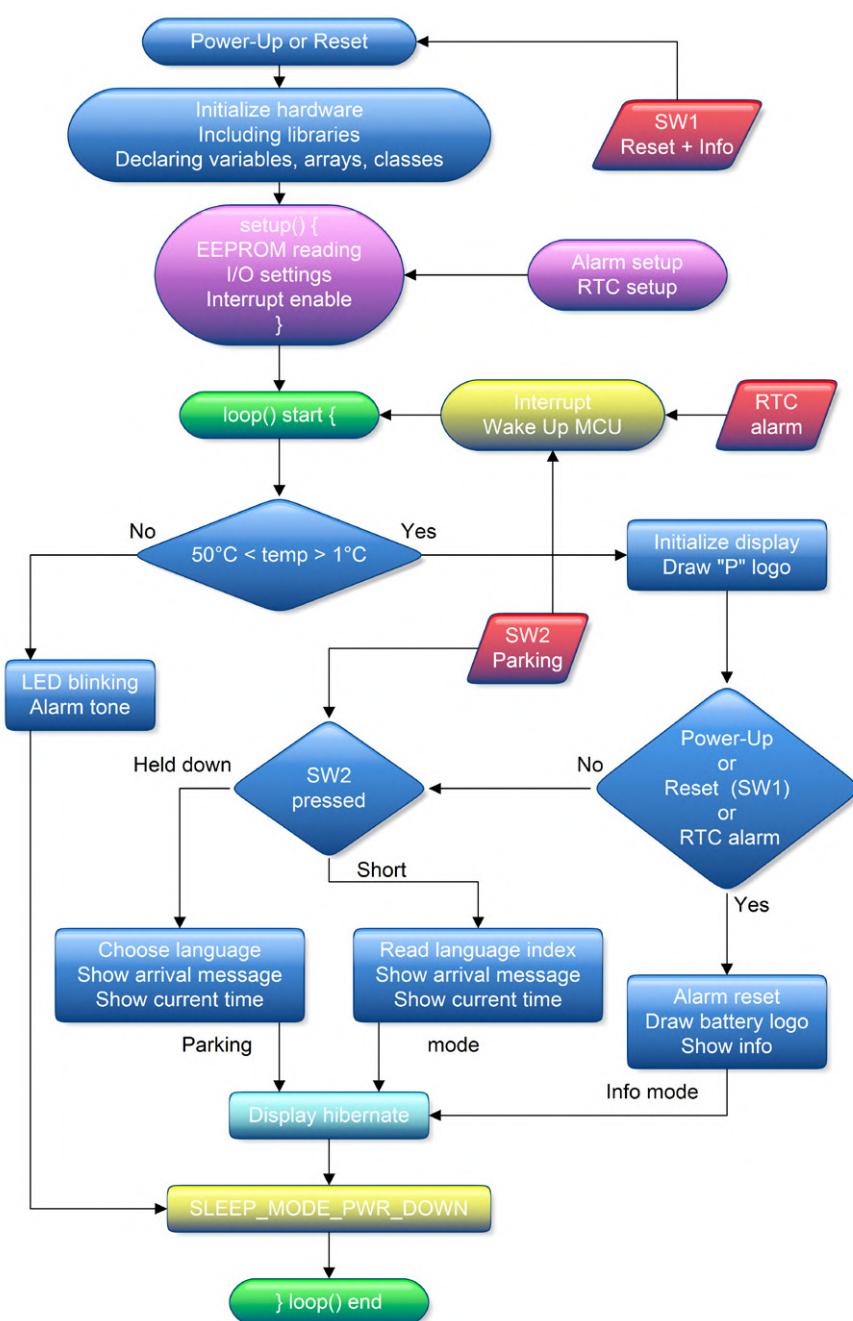


Figure 8. Organigramme du micrologiciel.

de température et à d'autres facteurs, une batterie de 40 mAh, à un courant moyen de 0,008 mA, pourrait théoriquement alimenter le circuit pendant environ $40/0,008 = 5\,000$ h (c'est-à-dire plus de 200 jours ou au moins six mois) ! De manière plus réaliste, même la moitié de cette durée serait un bon résultat. Des applications en ligne sont disponibles pour ce type de calcul. En voici une [5] que j'ai trouvée efficace et pratique.

Le micrologiciel et sa fonctionnalité

Dans ce projet, le code source [9] est écrit avec l'EDI Arduino 1.8.19 et nécessite, pour une compilation correcte, l'installation du noyau Arduino MiniCore v2.1.3 et de quelques bibliothèques spécifiques. Le noyau utilisé permet une gestion plus souple et efficace du microcontrôleur ATmega328P et surtout optimise la taille mémoire du code compilé, qui occupe 31,264 des 32,768 octets de mémoire programme (Flash) et 1,501 des 2,048 octets de mémoire dynamique (SRAM), presque à la limite des capacités de ce MCU.

Il faut noter que ce projet n'est pas réalisable, même en tâtonnant, avec une carte Arduino Uno, car une partie de la mémoire de cette dernière est utilisée par le chargeur pour permettre une programmation directe, alors que pour le microcontrôleur nu, nous utilisons un programmeur USBasp externe. À propos de mémoire, le mot clé **PROGMEM** apparaît plusieurs fois dans le listing, faisant référence aux tableaux d'octets des chaînes

multimètre numérique, n'est que de 3 µA. En supposant une utilisation du disque de stationnement quatre fois par jour, plus le cycle de rafraîchissement, nous obtenons un temps de fonctionnement total de $7,5 \times 5 = 37,5$ secondes et un temps de repos de $(24 \times 3\,600) - 37,5 = 86\,362,5$ secondes sur 24 heures. Nous obtenons donc :

$$I_{AVG} \approx \frac{[(10\text{ mA} \cdot 3,75\text{ s}) + (0,003\text{ mA} \cdot 86362,5\text{ s})]}{(37,5\text{ s} + 86362,5\text{ s})}$$

≈ 0,008 mA

En négligeant pour simplifier, la réduction de capacité due à l'autodécharge, à une baisse de la tension nominale, aux variations

de caractères bitmap et texte, qui sont des données en lecture seule. En déclarant ces tableaux comme **PROGMEM**, les fonctions peuvent accéder à ces données en les lisant directement depuis la mémoire Flash, sans les copier au préalable dans la SRAM, beaucoup plus petite, qui reste alors disponible pour l'exécution « dynamique » du programme. La bibliothèque DS3231M 1.0.6 est utilisée pour la communication avec l'horloge en temps réel (RTC), tandis que la bibliothèque GxEPD2 1.3.6, supportée par la bibliothèque graphique GFX_Root 2.0.0, a été choisie pour la gestion de base de l'afficheur e-papier.

Cette dernière doit être remplacée par celle fournie avec le projet, qui comporte des polices modifiées. La bibliothèque GxEPD2 est un monument qui ne dispose malheureusement pas d'une documentation structurée, qu'il faut aller chercher dans le code



des exemples disponibles, qui sont très nombreux, mais à première vue déconcertants en raison de leur complexité apparente, et dont on découvre par la suite qu'elle est due à la tentative d'être compatibles avec autant de modèles d'afficheur que possible. J'ai donc tenté de simplifier, en ne conservant que les fonctions et les définitions nécessaires au type d'afficheur utilisé dans le projet. Celles-ci se trouvent dans le fichier *Waveshare_29_BW_avr.h*, tandis que le fichier *ParkBitmap128x128.h* contient le tableau d'octets, obtenu au moyen d'un convertisseur spécial [6], représentant l'image bitmap du logo du parking (P majuscule, noir et blanc, inscrit dans un carré aux coins arrondis de taille 128 x128 pixels). Ces fichiers, disponibles au téléchargement, se trouvent dans le dossier du croquis, avec le fichier principal du code source *Disco_Orario_e-Paper.ino*, dans lequel on trouve également des liens vers les sites du noyau et de la bibliothèque, des commentaires approfondis sur le code, et d'autres indications que j'ai trouvées utiles. Je conseille aux lecteurs intéressés par les détails du listage de l'examiner en l'ouvrant avec l'EDI Arduino (ou leur éditeur préféré). Au lieu de cela, je voudrais illustrer ici le fonctionnement du programme d'une manière plus descriptive, à l'aide de l'organigramme de la **figure 8**.

En tenant compte du fait que le circuit est en permanence connecté à la batterie, lors de la première mise sous tension (Power-Up), les opérations d'initialisation et la fonction `setup()` sont exécutées, avec l'activation d'une interruption, puis la `loop()` est exécutée.

Au début de la boucle, la température ambiante est vérifiée. Si elle n'est pas dans la plage prévue, un signal d'alarme sonore et lumineux est émis, puis le microcontrôleur est placé en mode d'économie d'énergie maximale (*SLEEP_MODE_PWR_DOWN*). Sinon, la `loop()` continue en initialisant l'affichage avec le logo « P », l'état de la batterie, la date et l'heure courantes, la température et la langue sélectionnée pour le message d'arrivée, montrant ce que nous pouvons appeler le mode info, après quoi l'affichage passe en hibernation (économie d'énergie), le MCU est mis en sommeil et la `loop()` interrompue. À partir de cet état, le microcontrôleur peut être « réveillé » (*Wake Up*) par une réinitialisation matérielle (avec le bouton SW1) ou par une interruption, un événement généré par la fonction d'alarme quotidienne du RTC ou par le bouton SW2. Si la reprise est provoquée par SW1 ou l'alarme, la `loop()` redémarre depuis le début et se termine toujours en mode info. Par contre, si le microcontrôleur est réactivé via SW2, la `loop()` redémarre, initialise l'affichage avec le logo et, si le bouton a été pressé et immédiatement relâché, affiche le message d'arrivée et l'heure courante (mode *parking*). En revanche, en maintenant SW2 enfoncé, les messages dans les quatre langues défilent successivement. En relâchant le bouton lorsque la langue désirée apparaît, ce choix est mémorisé dans l'EEPROM jusqu'au prochain changement. L'heure courante apparaît alors, et le cycle se termine comme précédemment en mode *parking*. La **figure 3** montre un exemple des captures d'écran. Nous avons vu comment la date et l'heure courantes sont gérées et fournies par le circuit intégré RTC DS3231M, qui doit encore

être programmé après la première mise sous tension et en cas de défaillance de la tension d'alimentation. Afin de simplifier le micrologiciel et le circuit, en évitant l'ajout de boutons supplémentaires, la programmation de la date et de l'heure est effectuée en même temps que le chargement du croquis, au moyen d'une ligne de code spéciale insérée dans la fonction `setup()` :

```
DS3231M.adjust(DateTime(2022, 03, 02, 19, 10, 00));
```



Liste des composants

Résistances (0,25 W, 1%)

R1, R10, R12 = 330 Ω
R2, R4, R5, R6 = 10 k
R3, R8 = 5,6 k
R7 = 3,3 k
R9 = 100 Ω
R11 = 1 kΩ

Condensateurs

C1, C3, C5 = 100 nF, 50 V condensateur en polyester ou céramique multicouche
C2 = 100 µF, 10 V condensateur électrolytique
C4 = 470 nF, 50 V condensateur en polyester ou céramique multicouche

Semi-conducteurs

Q1 = IRLD024 MOSFET à grille de commande par niveau logique
Q2 = BC546B
D1 = Diode Schottky à faible chute
D2 = LED bleue 3 mm, forte luminosité
D3 = LED rouge 3 mm, forte luminosité
U1 = ATmega328P-PU
U2 = DS3231M RTC, boîtier SO-8

Divers

Y1 = Résonateur céramique 8 MHz
BUZZER = Bipeur piézo-électrique
SW1, SW2 = N.O. Bouton-poussoir
F1 = Fusible réarmable (polyfusible) 250 mA
X1 = Prise Micro-USB type B
JP1 = Connecteur PCB ICSP 6 broches, M
JP2 = Connecteur PCB 8 broches M/F
AFFICHAGE = Afficheur e-papier 2,9 pouces N/B, (voir texte)
BAT1 = Batterie Ni-MH 3,6V, 40 mAh (3 x V40H)
Embase DIP 28 broches
Embase DIP 8 broches
Adaptateur CMS — DIP8 pour boîtier SO-8
Carte de prototypage



Le format à utiliser est du type « YYYY,MM,DD, hh,mm,ss ». Une fois que vous avez entré les données correctes, vous pouvez charger le croquis, vérifier la date et l'heure, changer la ligne en commentaire (en ajoutant une double barre oblique en tête) et recharger le croquis, ceci afin d'éviter que l'horloge ne revienne à ses paramètres initiaux à chaque réinitialisation. Pour obtenir une précision suffisante, il suffit de mesurer le temps qu'il faut pour télécharger le code, disons 30 secondes, puis de lancer le premier téléchargement 30 secondes à l'avance. Après quelques essais, une synchronisation à la seconde près peut être obtenue gratuitement ! L'heure de rafraîchissement quotidien est également effectuée à partir du code, en entrant simplement l'heure souhaitée.

Enfin, une note sur la méthode utilisée pour mesurer la tension de la batterie. On la trouve souvent sur Internet, avec quelques variantes, et Microchip elle-même la documente dans sa propre note [7]. L'astuce consiste à définir, via les registres appropriés, la tension de référence interne (1,1 V) comme valeur d'entrée de l'ADC et la tension à mesurer (VCC) comme référence. Tout changement de V_{CC} modifiera la lecture de l'ADC, ce qui permettra de calculer la valeur de la tension avec suffisamment de précision.

Conclusion

Bien qu'offrant des caractéristiques d'un intérêt indéniable, qui la rendent particulièrement adaptée au projet présenté, la technologie e-papier présente également certaines limites, dont la plus évidente est la faible fréquence de rafraîchissement. Le modèle utilisé ici effectue le cycle de rafraîchissement complet en 2 secondes et le rafraîchissement partiel en 0,3 seconde. Ces valeurs ne peuvent donc pas rivaliser avec d'autres types d'affichage dans la visualisation d'images, de graphiques et de textes qui changent rapidement. Enfin, comme je le dis souvent, au-delà de l'utilité réelle de l'objet proposé, j'espère que vous avez trouvé quelques idées intéressantes à méditer et à réutiliser avec l'esprit d'un faiseur, et que cet article a éveillé votre curiosité et une envie d'expérimenter avec les afficheurs e-papier ! 

VF : Helmut Müller — 230012-04

Des questions, des commentaires ?

Envoyez un courriel à l'auteur (a.dellapia@eletronicaemake.it) ou contactez (redaction@elektor.fr).



À propos de l'auteur

Depuis son enfance, Antonello Della Pia est attiré par l'électricité et les appareils électroniques. Il est titulaire d'un diplôme d'études supérieures de « Technicien en génie électrique ». Antonello a toujours cultivé et développé sa passion pour l'électronique analogique et numérique. Actuellement, il s'occupe avec les microcontrôleurs et la programmation, en essayant d'améliorer ses compétences en informatique. Antonello aime développer et proposer des projets aussi originaux que possible et, comme il l'espère, aussi intéressants.



Produits

➤ **Waveshare Module d'affichage E-Ink/E-Paper de 2,9 pouces (SKU 18776)**
www.elektor.fr/18776

➤ **Ynvisible Segment kit d'afficheur e-papier (SKU 20143)**
www.elektor.fr/20143



LIENS

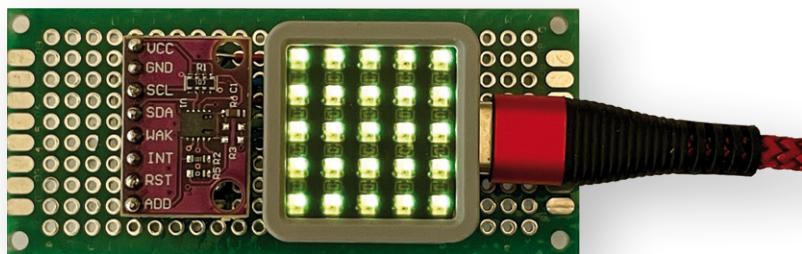
- [1] Papier électronique (Wikipédia) : https://fr.wikipedia.org/wiki/Papier_%C3%A9lectronique
- [2] Fiche technique de l'ATmega328P : <https://www.elektormagazine.com/atmega328p-datasheet>
- [3] Fiche technique du DS3231M : <https://datasheets.maximintegrated.com/en/ds/DS3231M.pdf>
- [4] Waveshare - 296x128, module d'affichage e-Ink de 2,9 pouces : <https://www.elektormagazine.com/waveshareeink>
- [5] Calculateur de durée de la batterie en ligne : <https://www.omniccalculator.com/other/battery-life>
- [6] BitmapToByteArrayConverter : <https://www.briandorey.com/post/bitmap-byte-converter-for-e-ink-display>
- [7] Mesurer la tension VCC/Batterie sans utiliser les broches d'E/S, Microchip : <https://www.elektormagazine.com/vcc/battery>
- [8] Principe de fonctionnement de base de l'afficheur e-papier : <https://www.elektormagazine.com/bwcapsules>
- [9] Code source : <http://www.elektormagazine.fr/230012-04>

eCO₂ Telegram bot

moniteur de qualité de l'air avec notification par Telegram

Peter Neufeld (Allemagne)

Déterminer la qualité de l'air intérieur ne nécessite pas toujours des appareils de mesure très précis et coûteux. Un simple indicateur plus un suivi des valeurs limites peuvent s'avérer très utiles si vous souhaitez simplement savoir quand aérer une pièce très fréquentée. Un tel appareil devient encore plus utile lorsque son simple afficheur intégré est complété par une transmission d'alertes vers un compte Telegram sur votre smartphone.



Une technologie de capteur hautement intégrée, un microcontrôleur moderne et un programme BASIC permettent, et sans grande complexité technique, d'utiliser ce détecteur de CO₂ (équivalent CO₂) et ses différents modes d'affichage pour indiquer une mauvaise qualité de l'air de quatre manières :

- LED NeoPixel locales comme indicateur visuel de la qualité de l'air
- Interface web pour les appareils locaux avec un navigateur web
- Requête manuelle via l'application de messagerie Telegram
- Message d'alerte Telegram envoyé à un utilisateur spécifique de Telegram

Matériel

Le circuit simple de la **figure 1** est basé sur un capteur CCS811 eCO₂ et un module SoC ESP32. La qualité de l'air est indiquée par une ou plusieurs LED NeoPixel. Un seul interrupteur à bouton-poussoir est connecté à une broche d'entrée. Le capteur et le SoC communiquent via un bus I²C à deux fils. L'écran NeoPixel n'utilise qu'une seule ligne de données au niveau du GPIO27, qu'il s'agisse d'une seule LED ou d'une matrice.

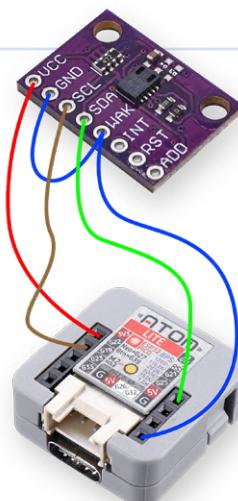
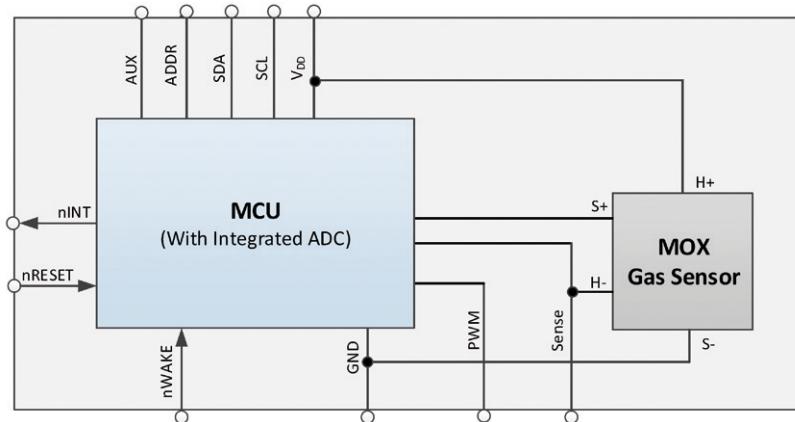


Figure 1. L'eCO₂ Telegram bot avec une matrice Atom M5Stack et un CCS811, y compris les circuits internes (source d'image : fiche technique AMS).

Bien que n'importe quel module ESP32 disposant de broches I²C convienne à cette tâche, je recommande l'utilisation d'un ATOM Matrix ou d'un ATOM Lite, de M5Stack. En effet, ces modules pratiques combinent un ESP32-PICO-D4 avec une antenne, et soit une matrice NeoPixel, soit une LED NeoPixel, ainsi qu'un bouton poussoir et d'autres composants, le tout dans un petit boîtier de protection compact et élégant. Ainsi, le capteur et le module ESP32 sont presque tout le matériel nécessaire pour construire l'appareil sur une carte perforée ordinaire. Pour ce faire, branchez le CCS811 et l'ATOM dans les connecteurs femelle et mâle appropriés, respectivement, et réalisez les différents branchements nécessaires comme le montre la **figure 2**. La face inférieure de la carte perforée traversante est isolée avec du ruban adhésif.

Le circuit est alimenté via le module ESP32 par une alimentation USB (5 V avec un courant maximal de 500 mA), soit via USB C, le connecteur HY2.0 Grove, ou les prises inférieures.

Capteur de CO₂ équivalent CCS811

Le capteur CCS811 ne peut pas mesurer directement le taux de CO₂ ! Il calcule le taux équivalent en CO₂ (CO₂e) en mesurant les VOC (composés organiques volatils totaux), la principale source de ces composés organiques volatils étant l'air expiré par les humains.

Ce capteur semi-conducteur à oxyde métallique (*MOS*) est peu coûteux. Il recherche une valeur relative au « bon air » en déterminant les meilleures conditions d'air sur une longue durée, puis en supposant que le capteur se trouve dans un air frais et non pollué contenant 400 ppm de CO₂ à un moment donné. Mais, le capteur ne stocke pas cette valeur.

En outre, la sensibilité du capteur peut varier dans le temps et dans des conditions environnementales différentes, telles que la température et l'humidité. Ainsi, pour fournir des relevés CO₂e fiables, le capteur nécessite :

- un temps de déverminage unique de plus de 48 heures et
- une période minimale de rodage d'environ 20 minutes après chaque démarrage à froid.

La fiche technique [1] contient des informations plus détaillées sur le capteur.

Logiciel

Le programme a été développé avec Annex32, un interpréteur BASIC pour ESP32 [2]. Après avoir installé l'interpréteur dans la mémoire flash du module ESP32 via un programme d'installation et une interface USB série, l'interpréteur et son environnement de développement fonctionnent directement sur l'ESP32. On a seulement besoin d'un navigateur Chrome ou Firefox pour charger, éditer, tester et exécuter (automatiquement) le script BASIC. La version minimale requise d'Annex32 est V1.435, car elle prend en charge le CCS811 et le messager Telegram. L'aide en ligne [3] d'Annex32 contient une introduction très utile à cet interpréteur BASIC. Les principales tâches du code BASIC sont les suivantes :

- Initialisation du CCS811 et relevé de la valeur du CO₂e une fois par seconde.

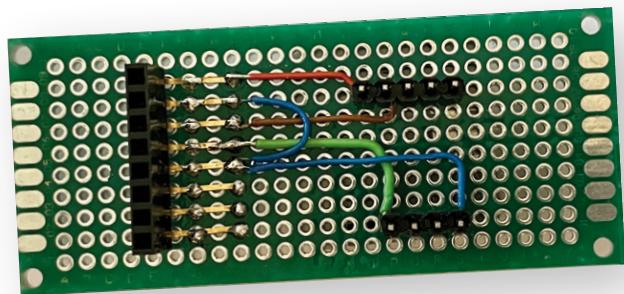


Figure 2. Soudure facile des deux modules sur la carte perforée.

- Classification de l'état de l'air ambiant par couleur VERT, JAUNE ou ROUGE selon le taux du CO₂e.
- Affichage de cet état par la matrice NeoPixel intégrée ou par la couleur de la LED NeoPixel.
- Une interface web affiche la valeur du CO₂e et la catégorie dans un navigateur sur le réseau local (W)LAN.
- Il est possible de consulter manuellement l'état et la valeur du CO₂e via Telegram, car notre programme comprend un bot Telegram et récupère les commandes entrantes de l'utilisateur à partir du serveur Telegram.
- Un message d'alerte Telegram est automatiquement envoyé au dernier chat_id Telegram lorsque la qualité de l'air est dans la zone rouge (**figure 3**).
- Il est possible d'enregistrer le niveau de référence de la qualité de l'air en appuyant sur le bouton frontal du module Atom ou en envoyant une commande Telegram.

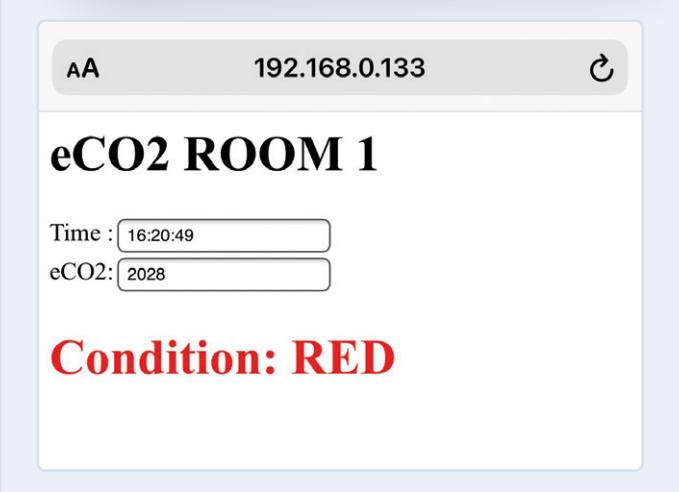
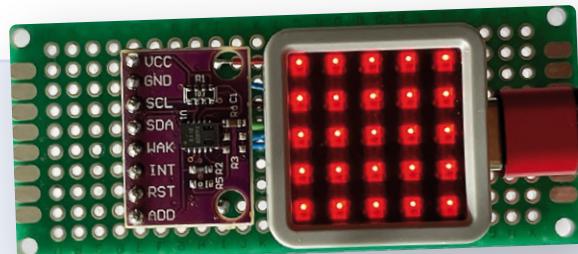


Figure 3. Message d'avertissement : l'air est vicié !



Extrait du programme : données sur une page web

```
'#####
MAKE_WEBPAGE:
'-----
'create the autorefreshing html page with dynamic
'display values and matching colors
A$ = ""
A$ = A$ + "<H1>eCO2 " + LOCATION$
A$ = A$ + "</H1>"
A$ = A$ + "Time :" + textbox$(T$) + "<br>"
A$ = A$ + "eCO2: " + textbox$(STR_eCO2$) + "<br>"
A$ = A$ + "|<span style="color:| + COND_COL$ + "|">
A$ = A$ + "<H1>Condition: "+ CONDITION$+ "</H1>"
A$ = A$ + "</span>"
cls
autorefresh 1000
html A$
return
'#####
```

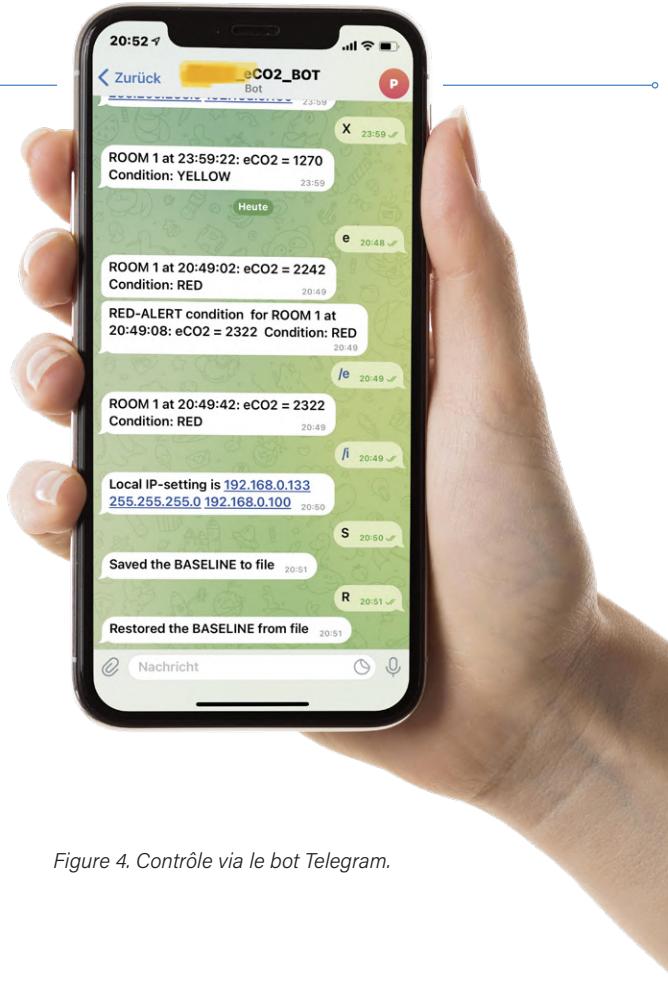


Figure 4. Contrôle via le bot Telegram.

La routine du bot Telegram consulte régulièrement les serveurs Telegram à la recherche de commandes entrantes. Il répond ensuite à ces commandes, comme le montre la **figure 4** :

- **/e** renvoie la valeur du CO₂e et la catégorie [VERT | JAUNE | ROUGE].
- **/s** enregistre la valeur de référence dans */baseline.txt*.
- **/r** rétablit la valeur de base à partir de */baseline.txt*.
- **/i** renvoie les paramètres IP locaux du module.
- [Tout autre caractère] effectue la même chose que **/e**.

Votre propre jeton Telegram

Afin d'utiliser les fonctionnalités de Telegram dans le programme BASIC, vous devez d'abord créer votre propre bot Telegram en suivant les instructions de BotFather [4] dans votre application Telegram. Vous obtiendrez ainsi votre jeton Telegram personnel et un nom de bot. Attention : vous devez inclure ces deux informations dans le programme BASIC pour définir les variables appropriées.

L'utilisation d'Annex32 BASIC, dont vous pouvez toujours trouver la version actuelle dans le forum Annex RDS [5], a au moins l'avantage d'être facile à lire – même pour les développeurs débutants – et d'être adaptable à vos propres besoins. Cependant, le code [6] est abondamment commenté pour une meilleure compréhension des fonctions. Ce script doit être recopié dans l'éditeur web d'Annex32 et sauvegardé dans le module ESP32 en tant que fichier d'exécution automatique (*/default.bas*). ↵

210566-04

Des questions, des commentaires ?

Envoyez un courriel à l'auteur (peter.neufeld@gmx.de) ou contactez Elektor (redaction@elektor.fr).

Produits

- **M5Stack AtomU ESP32 Development Kit (SKU 20184)**
<https://elektor.fr/20184>
- **ESP32-PICO-Kit V4**
Sans connecteurs : <https://elektor.fr/18423>
Avec connecteurs : <https://elektor.fr/20323>
- **Dogan and Ahmet Ibrahim: *The Official ESP32 Book (PDF)***
<https://elektor.fr/18330>

LIENS

- [1] Fiche technique du CCS811 : <https://bit.ly/2qQKqKu>
- [2] AnnexToolKit 1.1 y compris la version 1.48.22 :
<https://bit.ly/3Gtf8RS>
- [3] Annex32 WIFI RDS Help Version 1.48.2:
<https://bit.ly/3Gwhdg9>
- [4] BotFather sur Telegram : <https://t.me/BotFather>
- [5] Annex RDS forum: <https://bit.ly/3VBTRK2>
- [6] Ce projet sur Elektor Labs :
<https://www.elektormagazine.fr/labs/eco2-telegram>



les coulisses de l'audio haut de gamme maison

Ton Giesberts d'Elektor interviewé sur l'art de la conception analogique

De Jan Buiting (Elektor)

Rencontrez l'homme à l'origine de l'amplificateur de puissance Elektor Fortissimo-100 et de nombreux autres projets audio de premier ordre : Ton Giesberts, collaborateur d'Elektor Labs.

Jan : parlez-nous brièvement de votre formation technique, de votre expérience et de la manière dont vous avez rejoint Elektor.

Ton : après mes études, j'ai été contacté presque immédiatement par une agence d'intérim qui m'a proposé plusieurs emplois. Cela ne m'intéressait pas, mais un emploi chez un éditeur d'un mensuel technique, situé à Beek, m'a plu. En tant qu'abonné, je savais qu'il ne pouvait s'agir que d'*Elektuur*. J'avais déjà construit plusieurs circuits à partir du magazine et, après un entretien d'embauche, j'ai d'abord travaillé au laboratoire par l'intermédiaire de l'agence d'intérim avant d'être engagé définitivement. *Elektuur* recherchait simplement quelqu'un qui voulait faire de l'électronique analogique, ce qui me convenait parfaitement.

Jan : en dehors de votre profession, quels sont vos passe-temps ?

Ton : l'électronique était et est toujours mon passe-temps. J'aime lire et regarder des documentaires ou un bon film.

Jan : comment avez-vous vu évoluer le métier d'« électronicien » et sa pratique ?

Ton : au fil du temps, l'utilisation de composants traversants a diminué. Je suppose que l'émergence et l'utilisation de pièces CMS ont rendu l'assemblage maison de moins en

moins attrayant. L'électronique a évolué vers le développement de logiciels à un rythme rapide. Les cartes à microcontrôleur sont populaires depuis longtemps, mais même leur conception est moins attrayante que l'achat de modules prêts à l'emploi. Le logiciel détermine alors l'application, bien qu'évidemment aidé par quelques composants externes (analogiques) tels que des capteurs, etc.

Jan : comment votre travail est-il organisé aujourd'hui, et avec quels instruments et outils de laboratoire travaillez-vous habituellement ? Utilisez-vous un programme de simulation pour vos conceptions ?

Ton : maintenant que je travaille pour un autre département d'Elektor [Books, Kits and Pillar Products - Ed.], je dois consacrer du temps à d'autres tâches. Souvent, il ne s'agit pas de développer de nouveaux circuits ou de traiter les propositions qui nous sont directement soumis. Il s'agit de vérifier et d'obtenir des composants, d'assister des collègues et bien d'autres choses encore. J'utilise le plus souvent un multimètre et un oscilloscope. Bien entendu, quelques alimentations de laboratoire et un générateur de fonctions sont indispensables, sans oublier un analyseur audio. En plus d'un simple fer à souder, j'utilise un fer à souder à air chaud et un four à refusion pour souder les CMS. Pour la photographie des prototypes, j'ai une petite installation dans le grenier pour prendre des photos correctement exposées. Depuis l'époque de MS-DOS, j'ai toujours utilisé Micro-Cap pour la simulation.

Jan : nous avons appris que vous possédez un analyseur de précision audio. Que pensez-vous du fait de travailler avec un instrument dont beaucoup de passionnés de l'audio ne peuvent que rêver ?

Ton : sans un bon analyseur, il est pratiquement impossible de concevoir des circuits audio haut de gamme. Bien sûr, et heureuse-

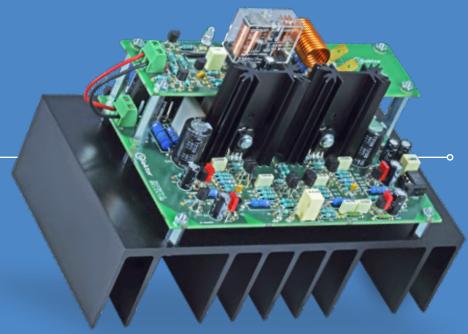
ment, les données mesurées ne font pas tout, mais elles sont excellentes pour révéler les imperfections. Un oscilloscope et un multimètre ne suffisent pas. Je suis très heureux d'avoir pu utiliser l'analyseur Audio Precision (AP) pendant de nombreuses années. Ce qui importe, cependant, c'est la manière dont vous mesurez et ce que vous mesurez. En d'autres termes, il faut savoir ce que l'on mesure ! Sans une bonne expérience et les bonnes conclusions, l'utilisation de l'analyseur AP ne sert pas à grand-chose. Cela ne s'apprend pas en un jour.

Jan : quelles sont vos principales sources d'inspiration ?

Ton : pour les nouvelles conceptions, les anciennes sont de très bonnes sources d'inspiration. On en trouve beaucoup sur l'internet. Le développement des transistors de puissance bipolaires, par exemple, n'est pas resté inactif. Qu'y a-t-il donc de mal à reprendre un vieux modèle, à le modifier ici et là et à utiliser des composants modernes pour lui donner une nouvelle vie ?

Jan : votre travail semble principalement axé sur l'électronique analogique, éventuellement combinée à des éléments numériques. Comment évaluez-vous l'importance relative de l'analogique et du numérique dans vos conceptions ?

Ton : je continue d'aimer les composants logiques standard, également en tant que pilotes ou en combinaison avec des circuits analogiques. Mais j'ai appris à apprécier le fait que les microcontrôleurs permettent de réduire le coût et la taille du matériel, tandis que les logiciels rendent le projet plus polyvalent. La programmation ne m'a jamais attiré ; la technologie analogique semble plus tangible, mais ce n'est peut-être qu'une illusion. Si j'utilise un microcontrôleur, c'est un de la famille AVR, et la programmation avec BASCOM est encore faisable, même



pour moi. Parfois, c'est même un changement amusant et bienvenu, en quelque sorte. De nos jours, cela peut être encore plus facile avec des outils tels que CircuitPython et un Raspberry Pi Pico.

Jan : vous avez contribué à la longue liste d'amplificateurs audio haut de gamme qui ont fait la renommée d'Elektor. Quels sont ceux qui ont eu le plus de succès, et quel est celui que vous préférez ?

Ton : l'un des premiers sur lesquels j'ai travaillé est le LFA-150 [1] (même si ce n'est pas moi qui l'ai conçu) et son dérivé, le LFA-50 Optim-A [2], puis The Discret, un amplificateur de moyenne puissance [3], puis, en avançant dans le temps, l'Audio Drive, un amplificateur d'une puissance modeste. L'amplificateur HEXFET60 de puissance moyenne [4] et son successeur, Nonante, l'amplificateur de puissance IGBT [5] (utilisant le même circuit imprimé) avec des IGBT spéciaux développés pour l'audio, ainsi que des amplificateurs de puissance pour les subwoofers actifs et divers systèmes de haut-parleurs actifs. Plus tard encore, l'amplificateur Crescendo Millennium Edition [6], et l'amplificateur ClariTy 2x300W Class-T [7]. Plus récemment, un amplificateur de puissance audio de classe D de 200 W [8]. L'amplificateur de puissance audio Q-Watt [9] et, bien sûr, le Fortissimo-100 [10] et de nombreux autres amplificateurs, petits et grands. L'amplificateur The Discret a été testé et évalué favorablement par un magazine audio allemand à l'époque, et c'est peut-être la raison pour laquelle je le préfère, mais je n'ai pas vraiment envie de choisir un favori.

Jan : votre amplificateur Fortissimo-100 a démarré doucement sur le site Internet d'Elektor Labs, a connu une excellente campagne Elektor JumpStarter et s'est vendu encore mieux après sa publication

officielle dans Elektor Magazine. Parlez-moi du processus de conception et de publication. Quels ont été les obstacles ?

Ton : on m'a demandé de concevoir un nouvel amplificateur de puissance. Je pensais principalement à un amplificateur de puissance moyenne et j'ai suivi ce concept jusqu'au bout, à l'exception de l'étage de sortie. Le problème était (et est toujours) que les grands transistors doubles tels que les MAT02 et MAT03 ne sont plus produits, et que les petites versions en CMS ne sont pas assez performantes. Je me suis donc tourné vers les bons vieux BC546B et BC556B, qui permettent de réaliser un amplificateur très rapide fonctionnant à une tension d'alimentation plus élevée. Bien sûr, les gens préfèrent maintenant la compensation Miller, mais je voulais changer le concept original aussi peu que possible. Je n'ai utilisé que des transistors de puissance Thermal-Track pour l'étage de sortie, ce qui m'a permis de tenter un bootstrap symétrique, d'abord dans une preuve de concept. Cela a fonctionné étonnamment bien, amenant un plus grand swing de sortie à portée de main. Pour faciliter à l'utilisateur de matcher au mieux les transistors BCx56 dans le kit, ils ont été utilisés dans autant d'endroits que possible, y compris les sources de courant des amplificateurs différentiels. Pour maîtriser la puissance, des résistances de collecteur sont nécessaires, ce qui oblige à utiliser une tension d'alimentation stabilisée, entre autres.

Jan : nous avons lu sur Elektor Labs qu'une alimentation linéaire est en cours de réalisation, notamment pour le Fortissimo-100. Quelle en est la raison ? L'alimentation à découpage SMPS800RE convient parfaitement, n'est-ce pas ?

Ton : pour certains, l'utilisation d'une alimentation à découpage est un blasphème et un

Non ! absolu. Mais il existe aujourd'hui de bonnes versions, spécialement conçues pour l'audio. L'une d'entre elles est le SMPS800RE. Il rend également l'ensemble de l'alimentation très compacte. Bien sûr, les niveaux de bruit d'une alimentation à découpage sont plus élevés et il y a une certaine ondulation RF. Mais ces bruits sont hors de portée de l'oreille et sont dûment supprimés par l'amplificateur. Pour répondre aux besoins de ceux qui s'intéressent à l'amplificateur, j'ai envisagé de fabriquer un régulateur analogique. Mais il devait rester modeste et facile à reproduire à la maison. Malheureusement, le travail de conception d'aujourd'hui est un véritable défi en raison de la faible disponibilité des composants. Il faut alors faire des compromis et c'est pourquoi ce projet utilise des composants anciens tels que le TIP35C et le TIP36C – toujours fabriqués par plusieurs fabricants et disponibles dans les rayons de plusieurs distributeurs.

Jan : comment et où vous voyez-vous travailler dans le domaine de l'électronique dans un an ?

Ton : je m'attends à faire plus ou moins la même chose qu'aujourd'hui, mais le monde est en plein bouleversement et prédire ce que l'avenir nous réserve relève de la pure spéculaction. ►

VF : Maxime Valens — 220603-04



Produits

➤ **Kit d'amplificateur de puissance Elektor Fortissimo-100 (SKU 20273)**
<https://elektor.fr/20273>

LIENS

- [1] « LFA-150 "VIRGIN" » Elektor, 11/1988 : <https://www.elektormagazine.fr/magazine/elektor-198811/53338>
- [2] « LFA-50 Optim-A » Elektor, 10/1991 : <https://www.elektormagazine.fr/magazine/elektor-199110/35041>
- [3] « The Discret » Elektor, 10/1990 : <https://www.elektormagazine.fr/magazine/elektor-199010/34783>
- [4] « amplificateur de puissance symétrique - HEXFET60 » Elektor, 12/1993 : <https://www.elektormagazine.fr/magazine/elektor-199312/35596>
- [5] « ampli 90 W à IGBTs; Nonante » Elektor, 9/1995 : <https://www.elektormagazine.fr/magazine/elektor-199509/36075>
- [6] « Crescendo Millennium Edition » Elektor, 4/2001 : <https://www.elektormagazine.fr/magazine/elektor-200104/9117>
- [7] « ClariTy 2x300W » Elektor, 6/2004 : <https://www.elektormagazine.fr/magazine/elektor-200406/9859>
- [8] « le déwatteur, ampli en classe D » Elektor, 12/2016 : <https://www.elektormagazine.fr/magazine/elektor-201612/40011>
- [9] « ampli de puissance compact » Elektor, 9/2013 : <https://www.elektormagazine.fr/magazine/elektor-201309/23301>
- [10] « Amplificateur haut de gamme Fortissimo-100 » Elektor, 11/2022 : <https://www.elektormagazine.fr/magazine/elektor-283/61180>



visite à domicile

nouvelle cible

Ilse Joostens (Belgique) et Eric Bogers (Elektor)

Les lecteurs d'Elektor connaissent bien Ilse Joostens, d'abord pour sa chronique *Sur le vif*, ensuite en tant que spécialiste des tubes Nixie et VFD. Comme nous tous, Ilse a subi les contraintes et conséquences de la pandémie de Covid-19. C'est l'une d'elles qu'elle nous raconte ici.

Les origines du dispositif présenté ici remontent à plusieurs années, comme nous le raconte Ilse :

« C'était il y a environ 20 ans. Je vivais à Termonde, et la célibataire que j'étais avait besoin de distractions sociales. Le hasard a voulu qu'il y ait près de chez moi un club de tir, je m'y suis donc inscrite.

On y pratiquait des séances de fun shooting, un jeu consistant à tirer sur des balles de ping-pong flottant sur de l'air comprimé. Le truc pénible, c'est qu'une

Figure 1. Une balle de ping-pong flottant sur un jet d'eau.

fois les balles touchées, il nous fallait décharger nos armes (par sécurité) et attendre que quelqu'un place de nouvelles balles. De là est née l'idée d'un tube d'alimentation qui guiderait les balles jusqu'au flux d'air et qui serait commandé par un bouton. Mais comment faire pour qu'une seule balle arrive à la fois ? Personne n'en savait rien. J'ai dessiné une idée de mécanisme sur une feuille de papier, mais en fin de compte n'en ai rien tiré de concret. »

Ilse était alors conceptrice indépendante, spécialisée dans les composants vintage comme les tubes Nixie et les afficheurs fluorescents (les VFD). Le choix de se spécialiser avait été motivé par des raisons pratiques : beaucoup de choses avaient déjà été conçues, et la concurrence de la Chine* était déjà forte. Une niche s'imposait.

« Sur un plan personnel, la pandémie et ses confinements ont été pour moi synonymes d'inactivité et d'ennui. Mon club de tir avait fermé, alors pour m'entraîner à la maison et passer le temps j'ai acheté des pistolets à air comprimé et à billes. »

* Durant la crise de la Covid-19, les Chinois ont acheté aux Russes les stocks de tubes Nixie restants. L'indisponibilité de l'horloge Nixie à six chiffres d'Elektor vient en partie de là.

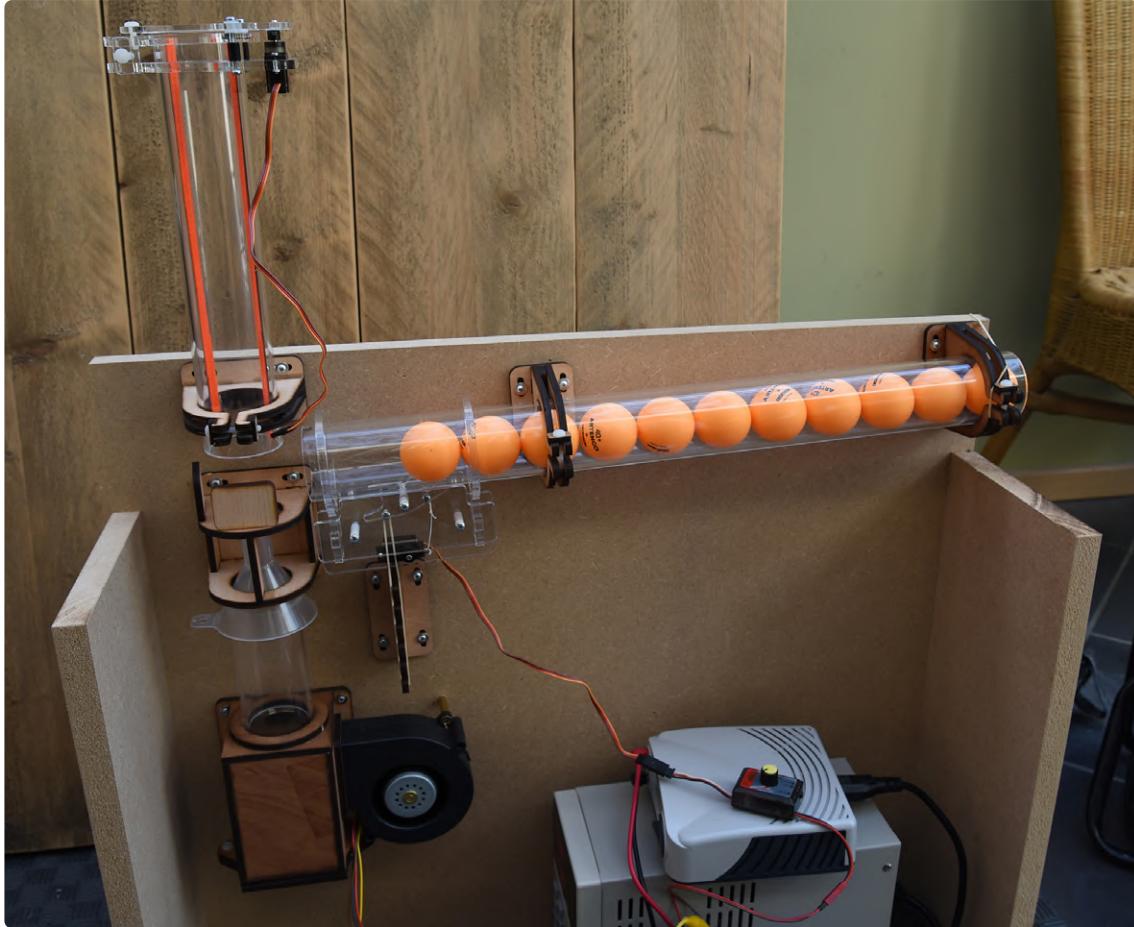


Figure 2. Mécanismes de soufflerie et d'alimentation des balles.

Côté professionnel, la pénurie de tubes Nixie et de plexiglas (elle aussi due à la pandémie) força Ilse à trouver une nouvelle orientation à ses projets. C'est alors que lui revint en mémoire le dispositif d'alimentation des balles de ping-pong. Son idée était de réaliser avec de l'air comprimé l'équivalent des jeux de fête foraine où le tireur doit toucher des balles de ping-pong flottant sur un jet d'eau (**figure 1**).

« Je n'en suis pour l'instant qu'au stade du prototype. Pour le flux d'air, j'ai utilisé un ventilateur 9BMC24P2G001 de Sanyo Denki. Il n'est hélas pas donné, mais il produit un flux d'air vigoureux et sa vitesse se commande par MLI. Or c'est justement en modulant l'intensité du flux d'air qu'il est possible d'obtenir des mouvements verticaux irréguliers de la balle de ping-pong, et donc d'en faire une cible intéressante. Le flux d'air passe par une boîte en bois surmontée d'un cylindre et d'un entonnoir orienté vers le haut, ce qui augmente fortement le débit d'air (**figure 2**). L'ensemble protège la soufflerie des éventuels morceaux de projectile, et la boîte s'ouvre par le bas, ce qui permet de la nettoyer.

Les balles de ping-pong sont stockées dans un tube en plexiglas incliné (d'un diamètre de 50 mm pour une épaisseur de 6 mm, **figure 2**). Elles sont entraînées vers le flux d'air par simple gravité. Le prototype actuel

peut contenir jusqu'à 11 balles, mais rien n'empêche d'utiliser un tube plus long ou de le compléter par une trémie d'alimentation. La rotation d'une came en plastique fait sortir les balles une par une. Une balle est libérée à chaque mouvement de bascule de la came, les autres restant bloquées mécaniquement. La came est maintenue dans sa position de base par un ressort de tension, et déplacée par un cordon relié à un servomoteur miniature (**figure 3**).

Figure 3. Le mécanisme garantissant qu'une seule balle à la fois gagne le flux d'air.



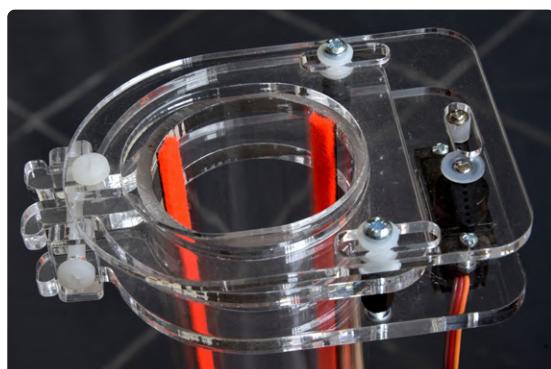
Ce mécanisme pourrait aussi servir à construire une machine à capsules Nespresso. Il y aurait plusieurs tubes côté à côté, un par arôme, et les capsules sélectionnées tomberaient sur un plateau. Je pense à cette idée depuis un certain temps, mais n'ai encore rien fait de précis. »

La balle qui roule hors du tube gagne le flux d'air sortant de l'entonnoir et se retrouve catapultée dans le grand tube vertical en plexiglas [1]. Sous certaines conditions aérodynamiques, il arrive que cette traversée dans le tube s'accompagne de ricochets contre la paroi, ricochets qui produisent alors un « cliquetis ». Deux bandes de feutre collées dans le tube évitent ce phénomène.

« Si le tube vertical était ouvert en haut, les balles seraient simplement projetées et finiraient par terre, sans être prises dans le flux d'air. Peut-être y a-t-il là le point de départ d'un distributeur de balles pour pongistes ? Quoi qu'il en soit la sortie du tube est bloquée par un obturateur à glissement latéral que commande un autre servo miniature (**figure 4**). Son ouverture libère la balle, qui s'élève alors sur une courte hauteur, oscille verticalement pendant un bref instant, puis se stabilise sur le flux d'air [2]. Le ventilateur peut dès lors être modulé par MLI afin de rendre le comportement de la balle plus ou moins imprévisible. Une balle flotte entre 15 et 20 cm au-dessus de l'obturateur. J'ai tenté de supprimer le tube vertical, mais son absence déstabilisait la montée de la balle, et celle-ci quittait presque systématiquement le flux d'air avant d'atteindre l'obturateur. »



Figure 4. L'obturateur (ici ouvert) bloquant les balles en haut du tube.



Il se sert de pistolets, mais on pourrait utiliser autre chose – p. ex. un arc et des flèches. Quelle que soit l'arme, mieux vaut par contre utiliser des balles de ping-pong très bon marché (et de couleur bien visible) puisque leur durée de vie risque d'être assez courte...

« À peine en avais-je terminé avec la partie mécanique que les pénuries de produits apparurent. Le ventilateur est actuellement indisponible, et il faut lui trouver un substitut. La partie électronique est moins sujette à problème puisqu'il n'y a que deux servos miniatures à commander (au sens électronique) et un signal MLI à produire pour le ventilateur. Le projet est pour l'instant en suspens, en partie à cause de la guerre en Ukraine. Je trouverais inopportun de sortir un jeu de tir au moment où les médias relatent tant de souffrances liées aux armes.

De toute façon je ne sais pas trop quelle direction donner à ce projet, ni s'il a un potentiel quelconque. Il faudrait à tout le moins ajouter une télécommande permettant de déclencher l'envoi d'une nouvelle balle, et éventuellement indiquer le nombre de balles restantes – nombre qui serait p. ex. déterminé avec des capteurs IR à réflexion placés sur le tube d'alimentation. On pourrait aussi ajouter un mode « flottement » et un mode « lancement » [2], ainsi qu'un bouton permettant d'activer ou désactiver la soufflerie.

J'aimerais donc savoir si ce projet vous intéresse, vous lecteur ou lectrice, et si vous avez des suggestions à faire ou souhaitez des fonctions supplémentaires (ou différentes). Comme toutes les composantes mécaniques peuvent être facilement découpées au laser et collées, il devrait être possible de les proposer sous forme de kit. Seule la partie électronique est un problème – et le restera au moins jusqu'à la mi-2023. »

210591-04 — VF : Hervé Moreau

Des questions, des commentaires ?

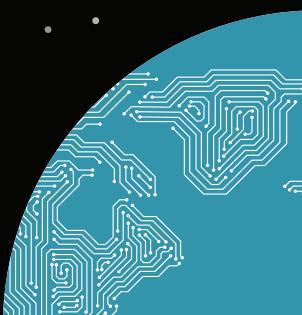
Contactez Elektor (redaction@elektor.fr).

LIENS

[1] Vidéo 1 – Alimentation des balles : <https://youtu.be/gTmytGPtHfQ>

[2] Vidéo 2 – Balles sur le flux d'air : https://youtu.be/iH6_5-zcdw

Ei-???

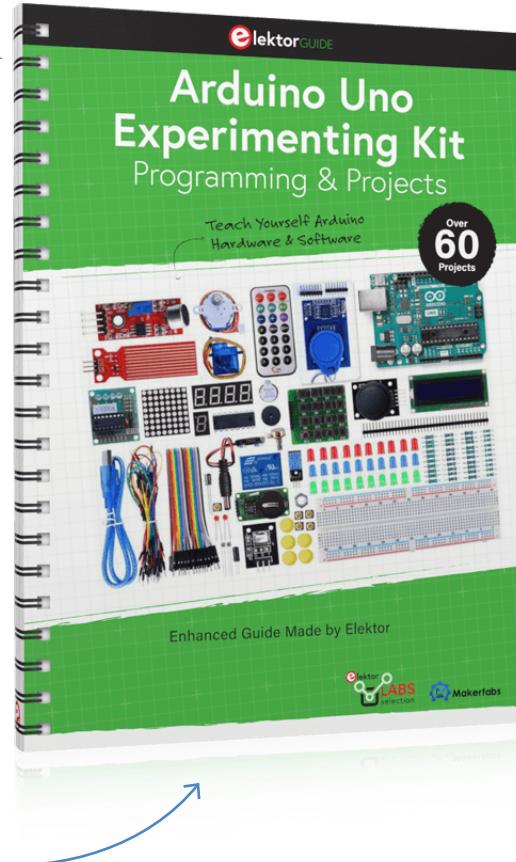


lecture d'étiquettes RFID et serrure de porte RFID

exemples de projets tirés du guide d'expérimentation pour Arduino d'Elektor

RFID est l'abréviation de *Radio-Frequency Identification*, une technologie sans fil qui couvre des dispositifs à des fins de sécurité, d'accès et de suivi des marchandises. En fait, un système RFID comprend un lecteur RFID et une ou plusieurs étiquettes. Les deux dispositifs matériels sont contenus dans un kit polyvalent de composants destiné à l'expérimentation avec Arduino et accompagné d'un guide spécialement rédigé à cet effet. Dans cet article, Dogan Ibrahim, auteur d'Elektor, explore la technologie RFID et s'appuie sur le fameux Arduino UNO, pour illustrer le développement d'un projet embarqué à partir de zéro.

Note de la rédaction. Cet article est un extrait du livre de 238 pages *Arduino Uno Experimenting Kit - Programming & Projects*. Ce guide fait partie de l'offre groupée d'Elektor *Arduino Experimenting Bundle*. L'extrait a été formaté et légèrement modifié pour correspondre aux normes éditoriales et à la mise en page du magazine Elektor. Étant donné qu'il s'agit d'un extrait d'une publication plus importante, certains termes de cet article peuvent faire référence à des discussions se trouvant ailleurs dans le livre. L'auteur et l'éditeur ont fait de leur mieux pour éviter une telle situation et répondront volontiers aux questions. Les coordonnées des personnes à contacter sont indiquées au paragraphe « Des questions, des commentaires ? ».



La technologie RFID utilise des champs électromagnétiques pour transférer des données sur de courtes distances. Les systèmes RFID sont principalement utilisés dans les applications de sécurité. Par exemple, ils peuvent être utilisés pour ouvrir une porte exclusivement par une personne possédant l'étiquette RFID correspondante.

Le lecteur RFID fourni avec l'offre groupée d'Elektor mentionnée ci-dessus (comprenant un livre et un kit de composants) est le « module RC522 » (**figure 1**). Il présente les caractéristiques suivantes :

- Fréquence de fonctionnement : 13,56 MHz
- Tension de fonctionnement : +3.3 V
- Fonctionnement avec le bus SPI et le bus I²C



Figure 1. Lecteur et étiquette RFID.



Le lecteur RFID est livré avec des connecteurs que vous devez souder au lecteur avant de l'utiliser.

Projet 1 : Trouver l'identifiant de l'étiquette

Dans ce projet, vous apprendrez à afficher l'identifiant de l'étiquette dans le moniteur série d'Arduino.

Schéma de principe : la **figure 2** est une illustration du schéma de principe du projet.

Circuit : les connexions entre les ports de la carte de développement UNO et le module de lecture RFID sont comme suit :

Broche du lecteur RFID	Port de la carte de développement
SDA	10
SCK	13
MOSI	11
MISO	12
IRQ	non utilisé
GND	GND
RST	9
3.3 V	3.3 V

La **figure 3** montre le schéma du projet. **Attention à ne pas connecter la broche d'alimentation à +5 V.**

Listage du programme : avant d'utiliser le lecteur RFID, vous devez ajouter la bibliothèque RFID à votre EDI. La bibliothèque est appelée *MFRC522*, et les étapes pour l'ajouter sont les suivantes :

➤ Téléchargez le fichier *rfid-master.zip* et placez-le dans un dossier. Vous pouvez le trouver dans les téléchargements associés au livre sur l'e-choppe Elektor [1].

- Lancer l'EDI.
- Cliquez sur *Sketch* → *Include Library* → *Add .ZIP Library*.
- Cherchez le fichier *.zip* sauvegardé et cliquez sur *Open*
- Vous pouvez maintenant commencer à utiliser la bibliothèque.

La bibliothèque du lecteur RFID offre de nombreuses fonctions qui sont accessibles en décompressant le fichier de la bibliothèque. Voici les fonctions les plus importantes de cette bibliothèque :

mfc522.PCD_Init()	Initialisation du lecteur RFID
mfc522.PICC_IsNewCardPresent()	Recherche d'un module de lecture RFID
mfc522.PICC_ReadCardSerial()	Sélection du lecteur RFID à utiliser
mfc522.uid.uidByte[]	Renvoie l'identifiant de l'étiquette dans un tableau
mfc522.PICC_HaltA()	Arrêt de lecture (Halt PICC)

Vous pouvez utiliser le programme *DumpInfo* de l'EDI Arduino pour déterminer l'identifiant (Tag ID) de votre carte. Voici les étapes à suivre :

- Lancer l'EDI
- Cliquez sur *File* → *Examples* → *MFRC522* → *DumpInfo*.
- Compilez et téléversez le programme sur la carte de développement.
- Lancez le moniteur série.
- Placez l'étiquette blanche sur le lecteur et maintenez-la en place jusqu'à ce que l'affichage des données s'arrête dans le moniteur série.

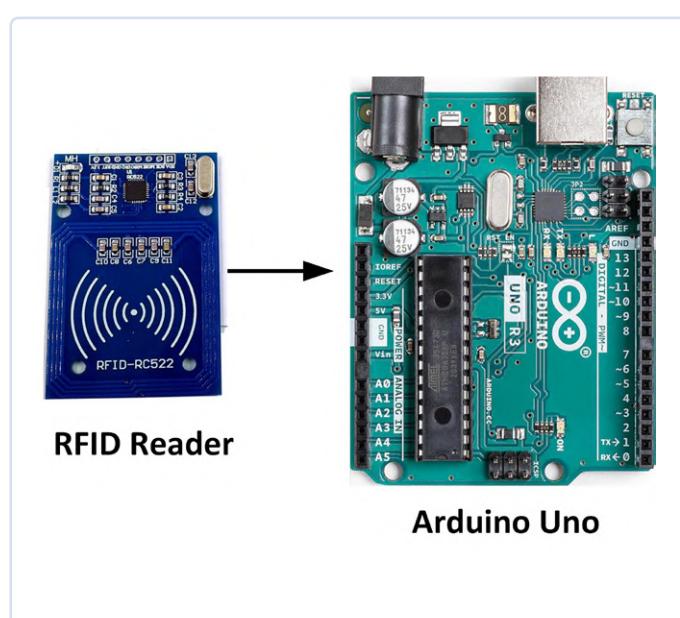


Figure 2. Schéma fonctionnel du projet de lecture d'étiquettes.

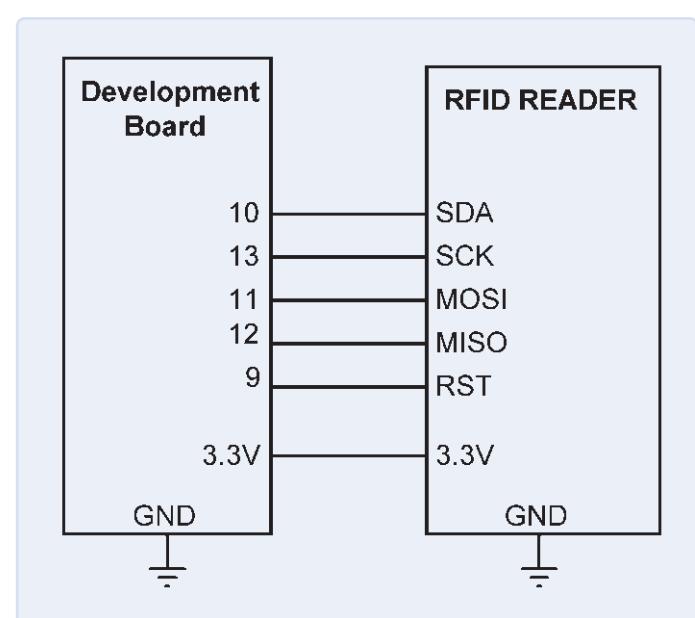


Figure 3. Circuit du projet de lecture d'étiquettes.

```

Firmware Version: 0x92 = v2.0
Scan PICC to see UID, SAK, type, and data blocks...
Card UID: 23 F0 58 A7
Card SAK: 08
PICC type: MIFARE 1KB
Sector Block 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 AccessBits
15 63 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF FF [ 0 0 1 ]
62 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
61 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
60 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
14 59 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF FF [ 0 0 1 ]
58 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
57 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
56 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
13 55 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF FF [ 0 0 1 ]
54 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
53 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
52 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
12 51 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF FF [ 0 0 1 ]
50 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
49 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
48 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]

```

Autoscroll Show timestamp Both NL & CR 9600 baud Clear output

Figure 4. Dédoubllement des données de la mémoire des étiquettes.

Des données similaires à celles de la **figure 4** devraient s'afficher. Il s'agit de 1 KB de mémoire de la carte et de son identifiant. La mémoire de 1 KB est organisée en 16 secteurs (0 à 15), où chaque secteur est divisé en quatre blocs (0, 1, 2, 3). Chaque bloc peut stocker 16 octets de données (0 à 15). Par conséquent :

$$16 \text{ secteurs} \times 4 \text{ blocs} \times 16 \text{ octets} = \\ 1,024 \text{ octets de données sur la carte (c.-à-d., 1 KB)}$$

Le troisième bloc de chaque segment (c'est-à-dire le bloc supérieur) est appelé « queue du secteur » (*Sector Trailer*) et contient les bits d'accès. Ceux-ci contrôlent l'accès en lecture/écriture aux blocs restants du secteur. Par conséquent, seuls les trois blocs inférieurs (c'est-à-dire les blocs 0, 1 et 2) de chaque secteur sont disponibles pour le stockage des données utilisateurs. Cela signifie que vous disposez de 48 octets (3 × 16 octets) par secteur de 64 octets.

Le bloc 0 du secteur 0 est appelé *bloc fabricant/données de fabricant*,

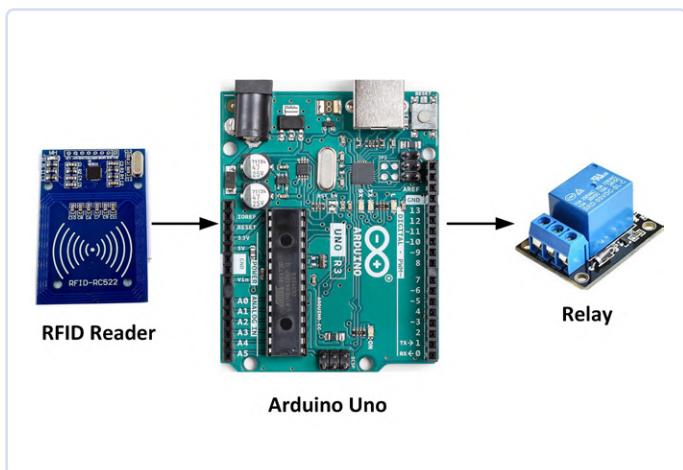


Figure 5. Schéma fonctionnel du projet de contrôle d'accès.

et il comprend les données du fabricant et l'identifiant de l'étiquette. L'ID de l'étiquette est également affiché sous le titre *Card UID* dans la **figure 4**. Dans ce projet, l'ID de l'étiquette est : **23 F0 58 A7**.

Projet 2 : Serrure de porte RFID et contrôle d'accès avec relais

Dans ce projet, le lecteur RFID et le relais fourni (également contenu dans le kit - NDLR) sont tous deux connectés à la carte de développement. On suppose qu'une porte d'entrée sécurisée est commandée par un relais protégé par un système RFID. Pour ouvrir la porte, il suffit de tenir une carte d'identification autorisée à proximité du lecteur RFID. Le relais est activé pendant 15 secondes, puis se désactive, de sorte que la porte se ferme.

Schéma de principe : la **figure 5** présente le schéma de principe du projet. Là encore, des images de cartes réelles sont utilisées - plutôt que des illustrations - avec des blocs fonctionnels.

Circuit : le schéma du circuit est présenté à la **figure 6**. Contrairement aux autres projets de « commutation » proposés dans le livre (non abordés ici), un relais est ajouté au port 2.

Listage du programme : le **listage 1** montre le programme développé pour la serrure de porte commandée par RFID, nommée *RFIDLock*. Au début du programme, les bibliothèques *SPI* et *MFRC522* sont incluses. L'ID de la carte valide est stockée dans la chaîne *ValidCard*, et *RELAY* est assigné au port 2. Dans la fonction *setup()*, *RELAY* est configuré comme sortie et est désactivé. Le bus *SPI* et le *MFRC522* sont également initialisés.

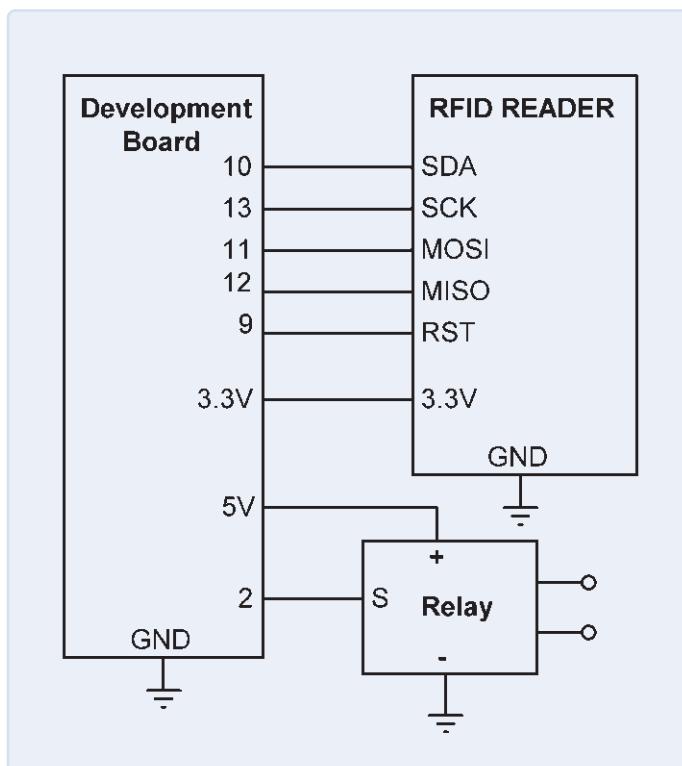


Figure 6. Circuit du projet de contrôle d'accès.



Listage 1. Le programme RFIDLock développé pour le projet.



```
-----  
//          RFID LOCK SYSTEM  
//          ======  
  
// In this program, the RFID card reader is used with a relay. The relay  
// is only activated if an authorized tag is placed near the reader. The  
// relay stays ON for 15 seconds and then turns OFF. The Tag ID of the  
// authorized valid card in this example is: 23 F0 58 A7  
//  
// Author: Dogan Ibrahim  
// File : RFIDLock  
// Date : May, 2022  
-----  
#include <SPI.h>  
#include <MFRC522.h>  
  
#define SS_PIN 10  
#define RST_PIN 9  
MFRC522 mfrc522(SS_PIN, RST_PIN);      // Create MFRC522 instance  
String ValidCard = "23F058A7";           // Valid Tag ID  
String TagID = "";  
int RELAY = 2;                            // RELAY at port 2  
byte i;  
  
void setup()  
{  
    pinMode(RELAY, OUTPUT);                // RELAY is output  
    digitalWrite(RELAY, LOW);              // Deactivate RELAY  
    SPI.begin();                         // Initiate SPI bus  
    mfrc522.PCD_Init();                  // Initiate MFRC522  
}  
void loop()  
{  
    if (!mfrc522.PICC_IsNewCardPresent()) // Look for card  
    {  
        return;  
    }  
  
    if (!mfrc522.PICC_ReadCardSerial())    // Select the card  
    {  
        return;  
    }  
    TagID = "";  
  
    for (i = 0; i < 4; i++)               // Read 4 byte Tag ID  
    {  
        TagID.concat(String(mfrc522.uid.uidByte[i], HEX));  
    }  
  
    TagID.toUpperCase();                 // Convert to upper case  
    mfrc522.PICC_HaltA();               // Stop reading  
    if(TagID == ValidCard)              // Valid card?  
    {  
        digitalWrite(RELAY, HIGH);       // RELAY ON  
        delay(15000);                  // Wait 15 seconds  
        digitalWrite(RELAY, LOW);        // RELAY OFF  
    }  
    else  
        digitalWrite(RELAY, LOW);       // RELAY OFF  
}
```



Le reste du programme s'exécute dans la boucle `main`. Ici, le programme attend qu'une carte soit placée près du lecteur. Elle est alors sélectionnée et une boucle `for` est créée pour lire l'identifiant de 4 octets dans la variable `TagID`. Ensuite, `TagID` est comparé à l'identifiant autorisé dans `ValidCard`. S'il y a correspondance, l'utilisateur est autorisé et le relais est activé pendant 15 secondes. Si la carte n'est pas valide, le relais reste désactivé. Ce processus se répète indéfiniment.

Enfin, la **figure 7** montre le projet construit sur la plaque d'essai. ↵

230066-04

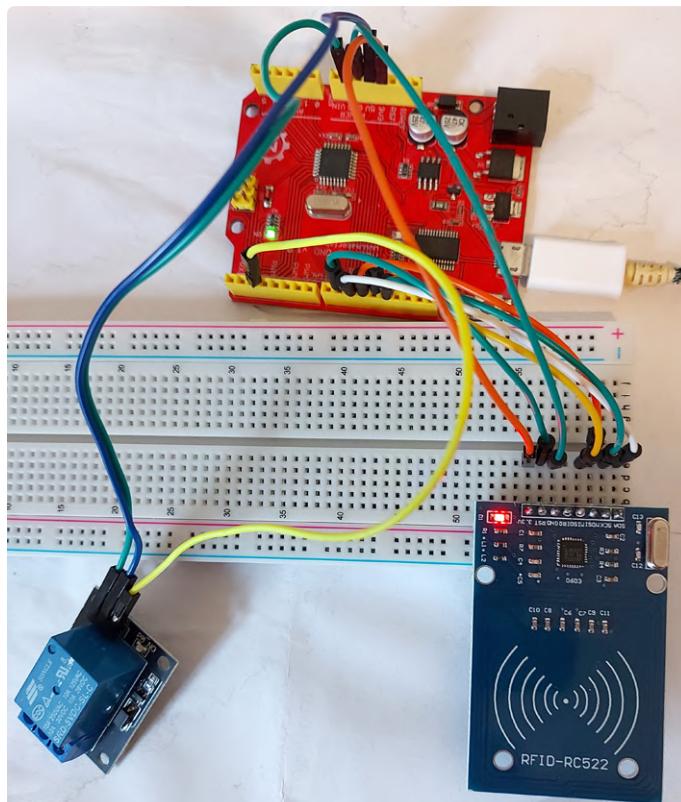


Figure 7. Projet de contrôle d'accès réalisé sur une plaque d'essai.



Produits

➤ Arduino Uno Experimenting Bundle (SKU 20340)

Inclus :

- Livre d'Elektor (en anglais) : *Arduino Uno Experimenting Kit - Programming & Projects* (238 pages)
- Kit de composants de Makerfabs avec un Arduino Uno R3.

www.elektor.fr/arduino-uno-experimenting-bundle



LIENS

[1] Téléchargements : <https://elektor.fr/arduino-uno-experimenting-bundle>

sondes de courant

RF pour oscilloscope

la mesure de courants RF sans peine

Roberto Visentin (Italie)

Avez-vous besoin de mesurer le courant d'antenne de votre émetteur HF ? Ou, ce qui est mon cas, le courant au primaire de votre bobine Tesla ? Ces situations nécessitent quelques considérations préliminaires. Vous trouverez ici quelques recommandations de conception, assorties d'un exemple concret.



Figure 1. Une photo en gros plan de ma sonde de courant RF.

Il peut être utile, dans de nombreux cas, de mesurer un courant sans composante continue. Cette approche concerne surtout les transformateurs de courant (TC) sur secteur en alternatif. Le présent article aborde la conception de transformateurs de courant pour fréquences moyennes et hautes, très

simples à construire. Les formules proposées s'appliquent également aux appareils alimentés sur le secteur en alternatif.

Principe de la sonde

La sonde de la **figure 1** est conçue pour mesurer jusqu'à 50 A crête dans une gamme de fréquences comprise entre 7 kHz et plusieurs dizaines de MHz. Le schéma de la **figure 2** est assez simple : le fil dont le courant doit être mesuré traverse le tore, constitué d'un noyau Amidon FT 82-43 ordinaire dont le fonctionnement est satisfaisant jusqu'à au moins 50 MHz.

L'enroulement du secondaire est constitué de dix spires de fil, réparties uniformément sur le noyau. Si possible, utilisez un fil toronné de calibre moyen, mais ce n'est pas obligatoire. En raison du rapport de bobinage de 1:10 (tours), le courant maximal dans le secondaire est de $5 A_p$.

Le côté secondaire est chargé avec une résistance de $0,2 \Omega$, ce qui a été réalisé par la connexion en parallèle de cinq résistances de 1Ω . Pour un courant crête de $5 A_p$ la tension crête aux bornes de ces résistances est de $1 V_p$ ce qui est très pratique pour les mesures

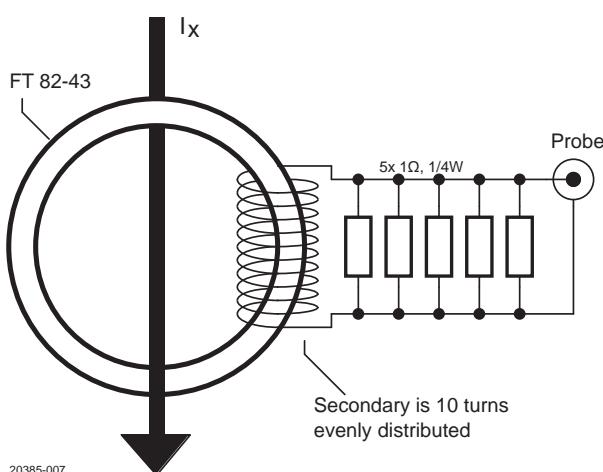


Figure 2. Le circuit simple de ma sonde de courant RF.

avec un oscilloscope. Pour un courant sinusoïdal, la puissance moyenne dissipée par les résistances est de $R \cdot I_p^2 / 2 = 2,5 \text{ W}$, soit $0,5 \text{ W}$ par résistance. Un courant sinusoïdal continu de 50 A_p ne peut être mesuré qu'avec des résistances de $0,5 \text{ W}$, ou plus. Pour autant, avec des formes d'onde pulsées ou des mesures très courtes à effectuer, des résistances de $\frac{1}{4} \text{ W}$ feront l'affaire. C'est le choix que j'ai fait car je voulais préserver le côté compact du dispositif pour une meilleure performance en RF. Je dois tout de même admettre que c'étaient les résistances que j'avais sous la main...

Utilisation

La figure 3 représente une utilisation typique avec une sonde d'oscilloscope, dotée d'un adaptateur BNC pour oscillo. Il est également possible d'utiliser le dispositif avec une connexion directe par câble coaxial à l'entrée de l'oscilloscope, la tension de 1 V_p étant idéale pour un fonctionnement en mode $1\times$. Dans ce cas, il est recommandé d'utiliser un câble court pour éviter les réflexions dans la bande concernée, car le coaxial sera désadapté des deux côtés. Mieux encore, il est possible d'adopter une terminaison du câble coaxial à son impédance caractéristique du côté de l'oscilloscope. De nombreux oscilloscopes modernes offrent la possibilité de

régler l'impédance d'entrée sur 50Ω , ce qui est donc particulièrement facile. Dans ce cas précis, il faut se rappeler que la mesure sera légèrement hors échelle, en raison de la mise en parallèle de la charge de 50Ω avec les $0,2 \Omega$ incorporés dans la sonde (la résistance totale atteint $0,1992 \Omega$, ce qui donne un facteur d'échelle de $50,2 \text{ A/V}$).

Ce qu'il faut éviter, c'est de fixer la sonde de l'oscilloscope directement avec les résistances à l'aide des pinces et de contourner le connecteur BNC, car lors de la mesure de courants RF élevés, même une minuscule boucle non blindée dans les sondes pourra ajouter des artefacts aux mesures.

Calculs

La conception du transformateur de courant n'est pas compliquée, mais elle nécessite l'application de quelques formules électromagnétiques. Tout d'abord, la résistance de charge RL doit être aussi faible que possible pour minimiser la perte de puissance introduite. En effet, le circuit à mesurer « verra » au moins $R \cdot n^2$, où $1:n$ est le rapport de bobinage en tours (1:10) et R est la somme de RL ($0,2 \Omega$) et de la résistance du fil secondaire (quelques $m\Omega$). Comme nous l'avons déjà dit, il est très important que le côté secondaire soit enroulé de manière uniforme, car sinon le circuit testé présentera une certaine inductance parasite

en série. À l'autre extrême, si nous choisissons une valeur trop faible pour RL , nous aurons également une très petite tension à mesurer, ce qui provoquera du bruit sur les traces. Enfin, RL doit être supérieure à la résistance du fil secondaire.

Dans mon cas, j'ai choisi $0,2 \Omega$ afin de pouvoir obtenir 1 V pour 5 A (50 A sur le primaire), ce qui ajoute $2 \text{ m}\Omega$ au circuit testé.

Le nombre de spires secondaires, n , détermine le rapport de courant. Dans le cas d'un TC en haute fréquence, ce nombre doit rester faible pour éviter l'auto-résonance résultant d'une capacité parasite associée à une inductance élevée. Dans le cas des TC sur secteur, la fréquence est assez basse (50 ou 60 Hz), et donc une valeur $n = 1000$ est courante. Les puissances de 10 sont classiques, afin que le rapport de conversion du courant soit simple, mais d'autres valeurs sont possibles.

La plus haute fréquence utilisable pour un TC toroïdal à ferrite dépend des facteurs suivants :

- les performances du matériau en ferrite ;
- l'auto-résonance de l'enroulement secondaire ;
- l'inductance de la résistance (la résistance elle-même et les connexions).

Un dispositif comme le mien peut facilement fonctionner jusqu'à plusieurs dizaines de MHz si l'on utilise une ferrite appropriée, comme le matériau 43 d'Amidon/Fair-Rite. Il est possible d'employer des noyaux à haute perméabilité pour la suppression des interférences électromagnétiques, mais seulement jusqu'à des fréquences beaucoup plus basses. Les noyaux à faible perméabilité utilisés pour les bobines de puissance et les inductances à facteur Q élevé ne sont pas recommandés, car leur inductance par tour est trop faible, ce qui conduit au point suivant.

Le choix du noyau de ferrite a également un impact sur la plus basse fréquence utilisable, pour deux raisons :

- Pour obtenir de bonnes mesures, nous voulons que la réactance du secondaire soit très supérieure à R , car elle agit également comme une charge. Comme la réactance a pour valeur est $X = 2\pi f L$, il y a un impact sur l'extrémité basse fréquence. Dans la pratique, nous voulons $X > 10 R_L$ à l'extrémité inférieure.
- Nous devons éviter la saturation du noyau, qui se produit pour un courant élevé et une fréquence basse. La saturation des ferrites se produit pour un champ $B = 0,25 \dots 0,3 \text{ T}$. Cependant, pour

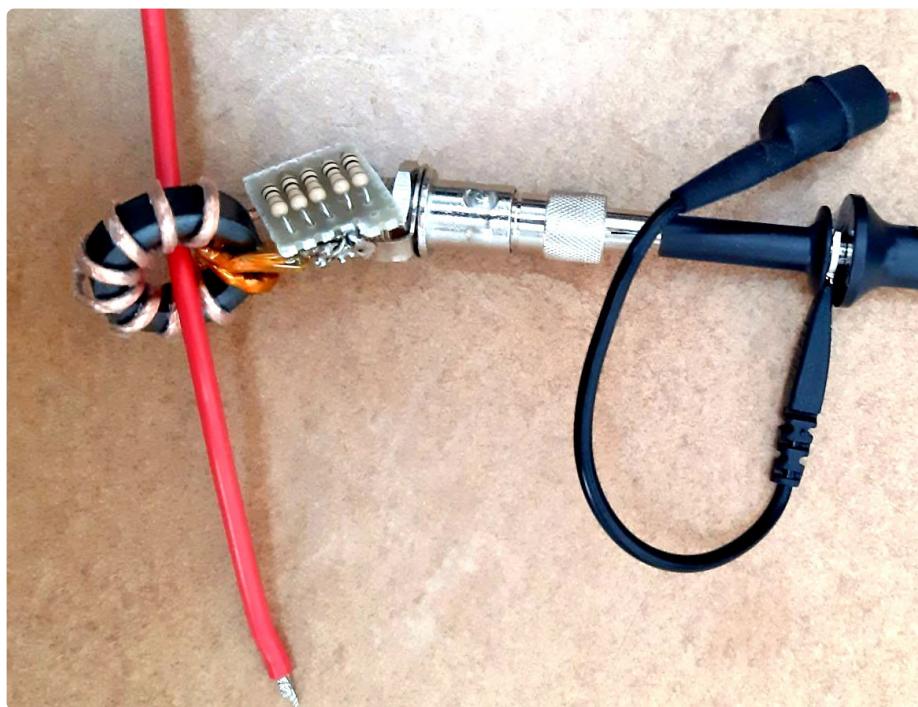


Figure 3 . Sonde de courant RF en utilisation pratique.

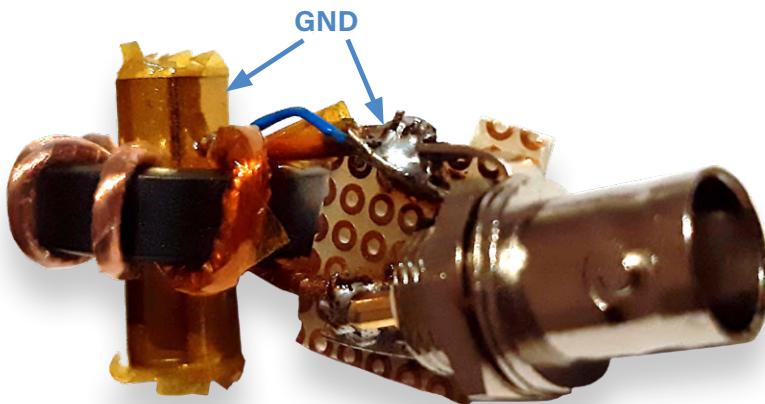


Figure 4 . Sonde améliorée avec blindage électrostatique.

éviter les phénomènes non linéaires, nous devons rester au-dessous de 0,2 T pour le courant maximum et la fréquence minimum.

Pour les deux raisons ci-dessus, les basses fréquences utilisables avec un petit nombre de spires et un courant élevé ont tendance à nécessiter des noyaux de grande taille. Les paramètres clés du noyau sont A_L , l'inductance par spire, généralement exprimée par nH/n^2 pour les noyaux RF, et la section transversale du noyau S, qui peut être facilement calculée à partir des dimensions du noyau. Dans le cas du noyau Amidon FT 82-43, $A_L = 470 \text{ nH/mm}^2$ et la section transversale est de $24,6 \text{ mm}^2$ (facile à calculer à partir du diamètre extérieur, du diamètre intérieur et de la hauteur).

L'inductance correspondant à 10 tours est donc de $470 \text{ nH} \times 10^2 = 47 \mu\text{H}$ (en gardant à l'esprit que l'inductance dépend de n^2). Si nous acceptons que $X = 10 \cdot R \approx 10 \cdot R_L = 2 \Omega$, en inversant la formule de la réactance, nous obtenons $f > 6,8 \text{ kHz}$ comme fréquence de coupure inférieure. Cette fréquence est acceptable ou non, selon l'application, et un noyau plus grand, dont la valeur A_L est plus importante, bénéficiera d'une coupure plus basse. Dans mon cas, pour mesurer une bobine Tesla musicale, j'étais intéressé par des fréquences supérieures à 500 kHz, et ce noyau était donc parfait.

En ce qui concerne la saturation, nous avons le fait qu'en régime sinusoïdal, l'amplitude du champ B (densité de flux magnétique) vaut $\text{abs}(B) = V_p / (n 2\pi f S)$, où V_p est la tension de crête aux bornes de l'enroulement. Cette expression peut être dérivée en mettant en équation les deux expressions du flux

magnétique lié $\Phi_c = L \cdot I = n B S$ et en remplaçant I par la valeur absolue du courant alternatif de magnétisation $\text{abs}(I) = V / (2\pi f L)$. Si V et n sont fixés lors des phases de conception précédentes, la seule façon d'atteindre une valeur f plus faible est d'adopter une valeur S plus grande, c'est-à-dire un tore de plus grandes dimensions.

Dans mon cas, $V_p = 1 \text{ V}$, $n = 10$, $S = 24,6 \text{ mm}^2$ (attention à utiliser des unités correctes), nous avons donc $B < 0,2 \text{ T}$ pour $f > 3,2 \text{ kHz}$, une valeur encore une fois idéale, du moins pour mon application. L'absence du courant dans le calcul peut prêter à confusion : en fait, il est caché dans la tension V, car nous savons que 1 V correspond à 50 A.

Améliorations

Le couplage capacitif entre les enroulements primaire et secondaire peut perturber les mesures aux plus hautes fréquences utiles, ou même à des fréquences modérées si le conducteur primaire est soumis à une tension RF élevée.

Il est possible d'améliorer le dispositif en ajoutant un blindage électrostatique destiné à éviter ce couplage capacitif : en pratique, le

conducteur primaire passe à l'intérieur d'un petit morceau de tube métallique (généralement en cuivre ou en laiton) connecté à la sortie de masse (GND) du secondaire, comme le montre la figure 4. Ce système n'altère pas la liaison magnétique et agit comme un bloqueur de champ électrique.

Conclusion

Cet exemple de transformateur de courant RF, mais aussi les principaux critères de conception, prouvent que ce domaine est moins complexe qu'il n'y paraît au premier abord. J'espère que les considérations et les formules présentées ici vous seront utiles pour simplifier vos expérimentations avec les noyaux toroïdaux, et qu'elles serviront de base à vos propres développements. ↗

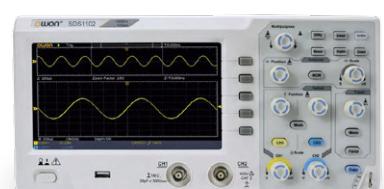
VF : Pascal Godart — 200385-04

À propos de l'auteur

Ingénieur électronicien récemment retraité, Roberto Visentin a travaillé sur des systèmes électroniques et de contrôle destinés à des applications marines et des robots sous-marins. Aujourd'hui consultant indépendant, il apprécie de réservé du temps pour développer des projets pour son plaisir dans le laboratoire électronique qu'il possède chez lui.

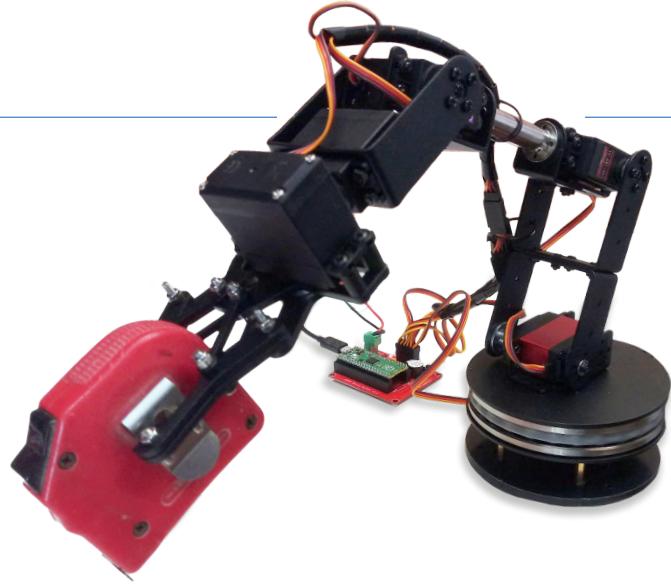
Des questions, des commentaires ?

Contactez Elektor à l'adresse redaction@elektor.fr.



Produits

- OWON SDS1102 oscilloscope à 2 voies (100 MHz) (SKU 18782)
<https://elektor.fr/18782>
- OWON XSA810 analyseur de spectre (1 GHz) (SKU 19714)
<https://elektor.fr/19714>



pas pour les mauviettes -

kit bras robotique

avec Raspberry Pi Pico et MicroPython

Clemens Valens (Elektor)

Un bras robotique capable de se mouvoir dans de nombreuses directions et de saisir des objets grâce à six articulations est un jouet amusant, d'autant plus lorsqu'il est facile à contrôler avec MicroPython fonctionnant sur un Raspberry Pi Pico. Avec ce kit, vous pouvez construire vous-même un puissant bras tout en métal.

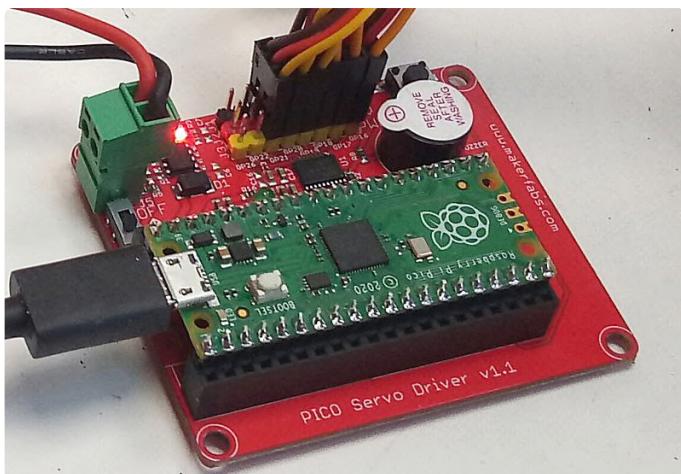


Figure 1. Une carte Raspberry Pi Pico contrôle les six servos du bras robotique.

Qu'est-ce que c'est ?

Le bras robotique 6 DOF avec Raspberry Pi Pico [1] est un bras robotique contrôlé par une carte microcontrôleur Raspberry Pi Pico. DOF est l'abréviation de *Degree(s) of Freedom* (Degré(s) de Liberté), ce qui, dans le cas d'un bras robotique, correspond à une articulation. Le bras examiné ici en possède six, ce qui lui permet de se déplacer à six endroits différents, indépendamment des autres. Chaque DOF correspond à un servo.

Le bras est contrôlé par une carte Raspberry Pi Pico montée sur une petite carte d'extension (**figure 1**) qui fournit les connexions aux servos, l'alimentation électrique, un buzzer et un bouton poussoir. Le Pico, ou plutôt le microcontrôleur RP2040 qu'il contient, est destiné à être programmé en MicroPython, mais ce n'est bien sûr pas une obligation. Cependant, le pilote et les exemples sont écrits en MicroPython.

Tout est inclus

Le bras robotique est livré sous la forme d'un kit complet (**figure 2**). En dehors d'un ordinateur sur lequel effectuer la programmation et d'un câble micro-USB, rien d'autre n'est nécessaire. Une alimentation de 5 VCC, 6 A est incluse. Un cordon d'alimentation est également fourni, mais avec une fiche de type américain.



Figure 2. Le bras du robot 6-DOF est livré sous forme de kit de pièces.



En fonction de votre lieu de résidence, vous devrez peut-être le remplacer ou ajouter un adaptateur (comme je l'ai fait). Il y a même un petit tournevis, mais j'ai préféré utiliser mes propres outils pour assembler le bras.

Il y a pas mal de pièces et encore plus d'écrous et de boulons, mais tous ne sont pas utilisés. À l'exception de la plaque de base en acrylique et de l'os du bras en acier, toutes les pièces sont en aluminium. En les assemblant correctement, vous ne vous retrouvez pas avec un jouet bancal en plastique, mais avec un bras en métal solide et résistant. Bien que tout soit en aluminium, le bras assemblé pèse 1,3 kg (sans l'alimentation électrique).

Outil pédagogique

Le bras robotique 6-DOF est principalement un outil éducatif pour enseigner la programmation dans MicroPython. Il s'agit d'un objet amusant qui peut se mouvoir de nombreuses façons et qui peut ramasser de petits objets comme sa propre alimentation (environ 6 cm × 4 cm × 8,5 cm). Comme le montre en quelque sorte une vidéo de démonstration [2], vous pouvez par exemple l'utiliser pour jouer au jeu Tours de Hanoi. Il n'est pas adapté aux applications industrielles ou aux tâches de haute précision. Il est doté d'une pince, mais je ne lui ferais pas confiance avec un outil dedans (**figure 3**).

Ce que vous devez savoir avant tout

L'assemblage du bras robotique n'est pas pour les renoncateurs. Le manuel [3] (non inclus, à vous de le trouver) est uniquement en chinois, et, en plus, peu clair et incomplet. Il existe également une courte vidéo d'animation de l'assemblage [4], mais elle est également incomplète. Pourtant, il est possible de le faire avec un peu de patience et de bon sens, car il ne s'agit pas d'un kit très compliqué. La seule partie complexe, la pince, est pré-assemblée (**figure 4**). Cependant, il y a certaines choses que vous devez savoir à l'avance (et que j'ai découvert à la dure) :

- Soyez prêt à démonter les choses, afin de pouvoir les réassembler de la bonne manière. Le construire deux fois donnera probablement les meilleurs résultats.
- Commencez par le bas, c'est-à-dire la plaque de base avec la plate-forme rotative, et remontez.
- Le grand roulement de la base rotative est en trois parties. J'ai remarqué une légère différence de qualité de rotation entre les deux configurations d'assemblage possibles, mais en fin de compte, cela ne semble pas avoir d'importance du tout, car il fonctionne simplement comme un poids mort.
- Utilisez d'abord les deux servos rouges YF-6125MG (**figure 5**). Mon kit en comprenait trois (un de rechange), mais le manuel et la vidéo les utilisent partout. Ces servos sont les plus forts et peuvent supporter le plus de poids, ils doivent donc être placés sur la partie inférieure du bras.
- Assurez-vous que tous les servos sont dans leur position médiane avant de les monter. Pour les servos noirs MG 996R, vous pouvez le faire manuellement/visuellement avec une sorte de pointeur/bras de servo (inclus dans le kit). Les servos rouges YF-6125MG, par contre, n'ont pas de butée de fin de course, donc vous ne pouvez pas voir quand ils sont dans leur position médiane. J'ai fini par utiliser le Pico pour mettre à zéro les servos à la volée, c'est-à-dire en assemblant le bras. Cela peut sembler compliqué, mais c'est en fait assez facile

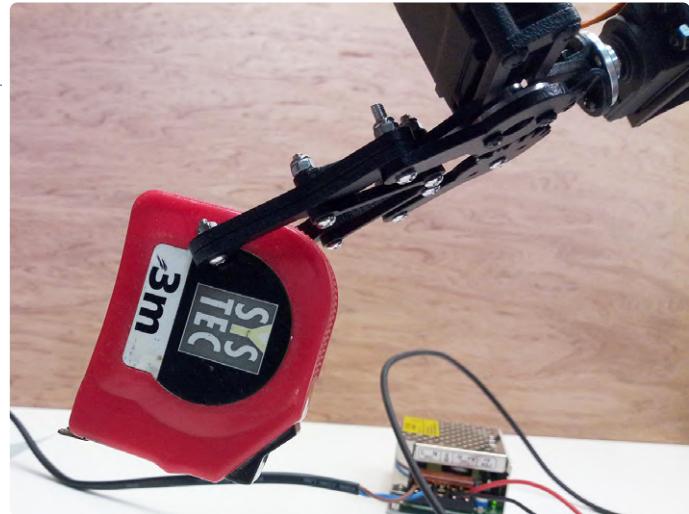


Figure 3. Le bras peut tenir un outil, mais je ne lui ferais pas confiance avec.

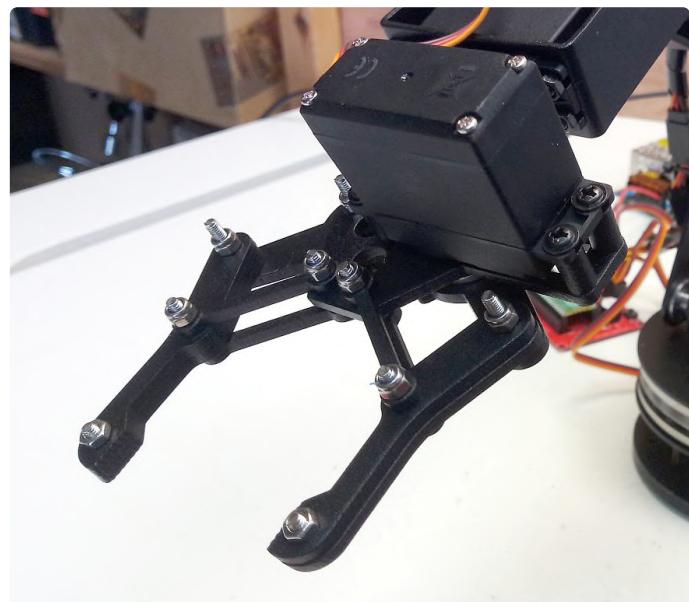


Figure 4. La pince est pré-assemblée, vous devez seulement monter le servo.

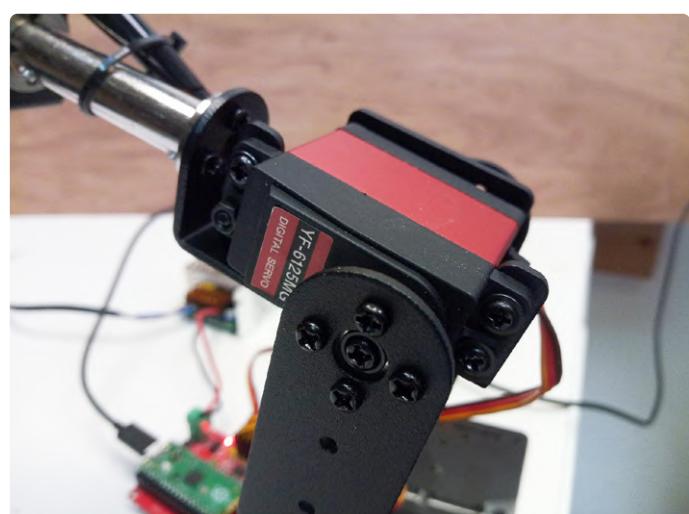


Figure 5. Les servos YF-6125MG sont puissants et doivent supporter la majeure partie du poids du bras.

à faire. Je vous suggère même de commencer l'assemblage en configurant l'environnement de développement MicroPython et en essayant d'exécuter les démos avec les servos non montés [5]. Une fois que vous aurez compris comment contrôler les servos depuis MicroPython, vous pourrez les mettre à zéro et les monter dans le bras.

- Dans tous les sous-ensembles, le servo est monté en dernier. Ils doivent généralement être fixés avec les vis à tête en forme de chapeau).
- Veillez à bien serrer le tout, car c'est difficile à faire une fois le bras assemblé.
- La pince est préassemblée et branlante, et j'ai donc serré un peu les écrous autobloquants. Le résultat a été que le servo contrôlant la pince a grillé car la pince était apparemment devenue trop serrée pour être manipulée (même si je pouvais la mouvoir manuellement). Heureusement, le kit comprend deux servos de rechange.
- Il n'y a pas de schéma de câblage, il y a six servos, et l'embase de connexion des servos a huit positions. Heureusement, il y a un code de couleur, donc la polarité est facile à respecter. Connectez les servos en commençant par le bas. Le servo le plus bas (la base rotative, « O » ou « A » dans le logiciel) se connecte à GPIO16, le plus haut (la pince, « 5 » ou « F » dans le logiciel) va sur GPIO21. Le logiciel permet de remapper les broches GPIO, et la direction de rotation de chaque servo peut être définie aussi, donc vous avez une certaine liberté ici pour corriger les choses.
- La plaque de base comporte de nombreux trous, mais pas pour fixer la carte Pico ou l'alimentation.
- Assurez-vous que l'alimentation produit 5 V avant de la connecter au bras. La tension de sortie peut être réglée entre 4,5 V et 6,5 V environ.

Soyez prudent !

N'assemblez pas naïvement le bras robotique et ne lancez pas la démo sans avoir testé chaque servo au préalable. Le bras peut se mettre à tourner et à bouger dans tous les sens, faire tomber des objets du plan de travail ou vous frapper au visage. Ses servos sont puissants ! J'ai réussi à faire en sorte que le bras casse sa propre plaque de base acrylique de 4 mm d'épaisseur à cause d'un servo qui a poussé vers le bas sur le plan de travail au lieu de tourner vers le haut. En particulier, les servos rouges YF-6125MG tournent assez vite, alors faites attention. Je recommande donc fortement de fixer le bras au plan de travail (**figure 6**) ou sur une plaque lourde, et de garder l'espace autour libre.

Notez également que les servos rouges YF-6125MG n'ont pas de butée de fin de course et peuvent se tromper de sens pour atteindre leur position cible. Cela peut se produire lorsque le bras est replié puis remis sous tension. Je recommande donc vivement d'étirer le bras jusqu'à sa position verticale avant de le mettre sous tension (voir **figure 7**).

Pas pour les mauviettes

Le bras robotique 6 DOF avec Raspberry Pi Pico de Makerfabs [6] n'est pas un kit pour les mauviettes. L'assembler correctement demande de la patience et de la persévérance. Même si le bras est assez facile à utiliser grâce à son pilote et à sa démo écrite en MicroPython, il doit être apprivoisé pas à pas, sous peine d'accidents.



Figure 6. Fixer le bras du robot au plan de travail permet de rendre les expériences plus sûres.

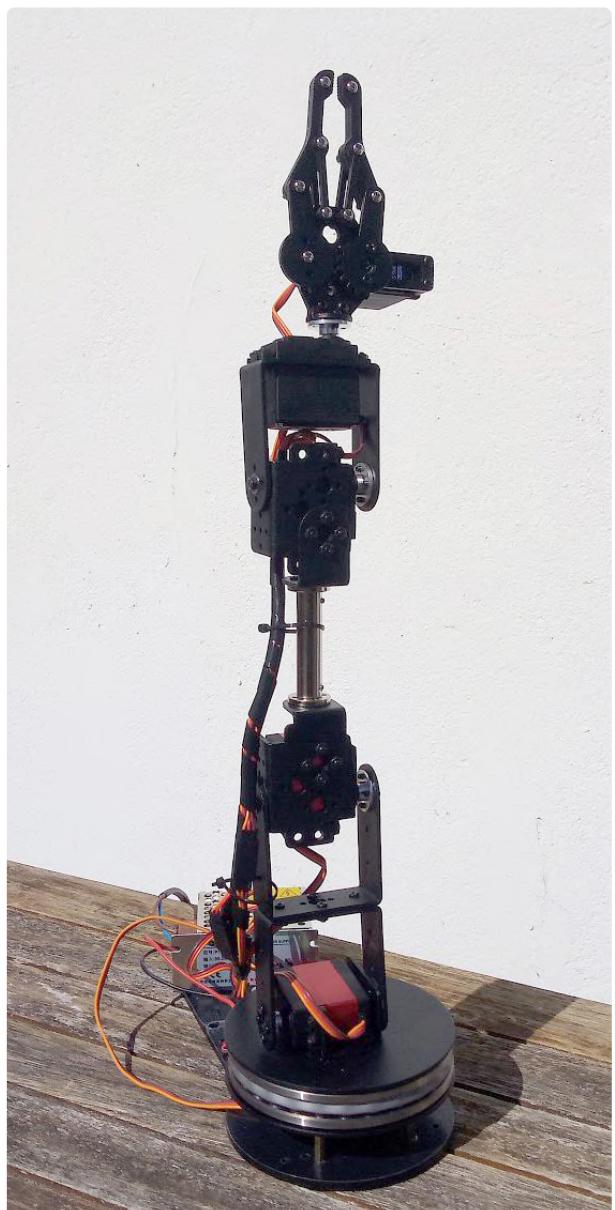


Figure 7. La position verticale est la position de départ la plus sûre.



Ses servos sont rapides et puissants, et le bras mesure 44 cm de long lorsqu'il est entièrement déployé (50 cm en incluant la base rotative). Il est fortement recommandé de le fixer au sol au centre d'un mètre carré vide de tout objet. Vous ne voulez pas qu'il écrase votre ordinateur portable ou qu'il fasse tomber votre tasse à café de la table, n'est-ce pas ? ↪

VF : Maxime Valens — 220232-04



Des questions, des commentaires ?

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



Produits

- Bras robotique 6 DOF avec Raspberry Pi Pico de Makerfabs (SKU 20130)
www.elektor.fr/20130
- Kit d'expérimentation Raspberry Pi Pico d'Elektor (SKU 19834)
www.elektor.fr/19834

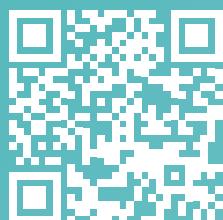
LIENS

- [1] Le bras robotique 6 DOF sur Elektor : <https://www.elektor.fr/makerfabs-6-dof-robot-arm-with-raspberry-pi-pico>
- [2] Vidéo de démonstration : <https://youtu.be/8dB1W6Jd07g>
- [3] Manuel [Zip file] : <https://www.makerfabs.com/desfile/files/Raspberry-Pi-PICO-6-DOF-Robot-Arm.zip>
- [4] Vidéo d'animation de l'assemblage : <https://youtu.be/NqL9n3avBbY>
- [5] Pilote et exemples écrits en MicroPython : https://github.com/Makerfabs/PICO_Mechanical_Hand_Driver
- [6] Site web de Makerfabs : <https://makerfabs.com/raspberry-pi-pico-6-dof-robot-arm.html>



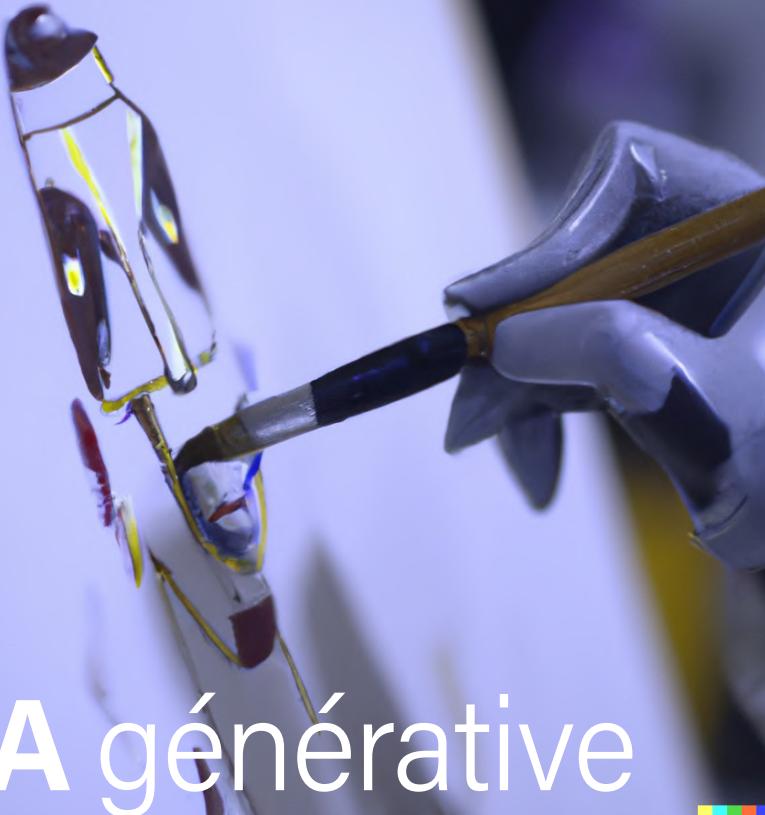
Tirage limité !

Disponible sur www.elektor.fr/special
ou dans votre kiosque préféré.



15,50 €





IA générative

mais qui a fait ça ?

Priscilla Haring-Kuipers (Pays-Bas)

DALL-E et ChatGPT – tous deux créés par OpenAI – fascinent et pénètrent tous les secteurs. L'IA générative devient un débat de société.

IA comme Inspiration Artificielle ?

Une des critiques adressées aux IA génératives d'images telles que DALL-E [1] est qu'elles ont été entraînées avec des œuvres sans que leurs auteurs aient donné leur autorisation. Les artistes ont toujours été inspirés par d'autres artistes. Copier une œuvre est même considéré comme une pratique formatrice, à l'artiste ensuite de trouver sa propre voie. Alors pourquoi une IA ne pourrait-elle pas faire la même chose ? Une des raisons est que ces IA n'existaient pas au moment où l'artiste a mis ses œuvres en ligne, qu'il ne l'avait pas fait pour entraîner une IA, et que personne ne lui a demandé son accord pour ça. Les règles ont été modifiées du jour au lendemain à son insu, et nous devons maintenant déterminer ensemble quelles règles sont acceptables et comment les appliquer.

Une autre raison est que l'IA ne s'inspire pas de l'art. Elle collecte un jeu de données à partir duquel elle construit un modèle. Nous apprenons l'intelligence à travers le prisme de nos propres expériences, et nous nous attendons à ce que l'IA fonctionne plus ou moins comme nous. Or, ce n'est pas le cas. Nous

Source : P. Haring-Kuipers et DALL-E, avec l'invite AI working on a painting.

sommes des êtres multimodaux traitant l'information de façon directe et indirecte, notre apprentissage est associatif, et notre pensée est influencée par des facteurs tels que les filtres émotionnels et la neurochimie. L'IA peut traiter plus de données que nous et plus vite, mais elle le fait différemment. Il faut des années à l'humain avant que ses acquis intellectuels et émotionnels ne lui permettent de créer ce que nous jugeons être de l'art. L'IA ne se développera pas comme nous si nous ne faisons que lui fournir plus de temps ou de puissance de calcul. Nous façonnons l'IA.

Une IA à notre image

Les techniques d'exploration de données (de *data mining*) servent à construire des modèles capables de mimer nos comportements. Dans le cas du modèle linguistique sous-jacent à ChatGPT [2], les données sont le gigantesque corpus textuel laissés par vous et moi sur l'internet. Si ces modèles vous intéressent, Hannah Fry les explique dans l'épisode *Speaking of Intelligence* de sa série de podcasts DeepMind [3].

Un des problèmes de ChatGPT est d'être (involontairement) raciste, agressif et odieux. Il n'a aucun système de valeurs intégré pour juger du contenu servant à son entraînement. Alors d'où viennent ses « valeurs morales » ? D'humains. De travailleurs (sous-)payés à visionner des images et des textes immoraux, et à les étiqueter comme tels afin que le modèle apprenne à les reconnaître et cesse de les produire [4]. Le contenu des médias sociaux est modéré de la même façon. L'amélioration d'un modèle passe donc par un incessant travail d'étiquetage avec des mots comme « viol », « contenu raciste », etc. L'intelligence de l'IA est en partie la nôtre.

Autre problème, les faits énoncés par ChatGPT sont souvent incorrects. Le site Stack Overflow interdit de poster des réponses fournies par GPT, car « même si celles-ci ont un taux élevé d'erreurs, elles donnent souvent l'impression d'être correctes et sont très faciles à produire par quiconque. » [5] Nous interactons avec l'IA sur le mode de la conversation, et avons donc tendance à traiter ses réponses de façon relâchée et à les juger plus vraisemblables – si tant est que nous les évaluons vraiment. En utilisant ChatGPT, gardez à l'esprit que vous parlez à la médiane de l'internet, et ne lui faites confiance qu'en conséquence.

Un revenu universel comme solution
Nous pourrions atténuer les effets d'une mainmise de l'IA sur le travail humain en versant à chacun un revenu universel. Assez d'argent pour vivre. Sans conditions, sans questions.

Is it safe to use ChatGPT for your task?

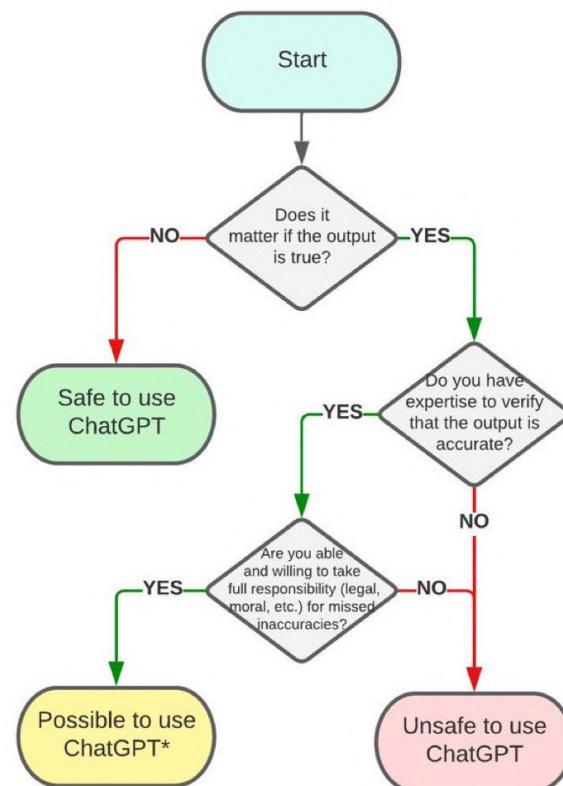
Aleksandr Tiulkhanov | January 19, 2023

Les IA génératives ne menacent pas la créativité de l'artiste, elles menacent ses revenus. Je pourrais utiliser l'IA pour créer des visuels accompagnant mes articles – et non pas me les procurer auprès d'une banque d'images – mais je ne veux pas qu'une IA les écrive. Je veux les écrire moi-même. Par souci d'économie, mon éditeur pourrait toutefois décider l'inverse. Ce genre de décision peut affecter une vie professionnelle et personnelle à des degrés divers.

Ne vous y trompez pas : aucune profession n'est à l'abri de l'IA. Graphistes, auteurs, avocats, codeurs, ingénieurs, tous verront bientôt une partie de leur travail remplacée par une IA générative. Nous pourrions utiliser l'IA comme outil, et nous concentrer avec profit sur ce qui nous motive réellement en garantissant nos moyens de subsistance par un revenu universel. Nous aurions alors tout loisir de réfléchir à une division du travail entre ce qu'humains et IA font de mieux chacun de leur côté.

Face à l'IA comme outil, il est clair que ce qui nous revient à nous, humains, c'est la prise de décision. Décision d'appliquer (ou non) ce que l'IA produit, ou de relancer le programme après avoir modifié l'invite ou certains paramètres. Nous sommes également responsables de l'exploitation des résultats d'une IA, des informations qu'elle livre et de leur partage, p. ex. dans un cadre médical. ↗

VF : Hervé Moreau — 230048-04



Logigramme de ChatGPT selon Aleksandr Tiulkhanov, chercheur en IA.

L'auteure

Priscilla Haring-Kuipers est titulaire d'un master en Psychologie des médias et directrice générale de *This Is Not Rocket Science*. Convaincue des bienfaits de la recherche et de la technologie sur le devenir de l'humanité, elle aborde ses articles sous l'angle des sciences sociales.

Forum mondial de l'électronique éthique

Le *Forum mondial de l'électronique éthique* (WEEF) se veut un lieu d'inspiration, de débats et de publications sur l'éthique et les objectifs de développement durable. Pour en savoir plus, visitez worldethicalelectronicsforum.com.



LIENS

- [1] OpenAI, DALL-E 2 : <https://openai.com/dall-e-2/>
- [2] OpenAI, ChatGPT: Optimizing Language Models for Dialogue, 30 novembre 2022 : <https://openai.com/blog/chatgpt/>
- [3] H. Fry, Speaking of intelligence, podcast DeepMind (S2, Ep2), 25 janvier 2022 : <https://youtu.be/21JSKHR7KWw>
- [4] B. Perrigo, OpenAI Used Kenyan Workers on Less Than \$2 Per Hour to Make ChatGPT Less Toxic, 18 janvier 2023 : <https://time.com/6247678/openai-chatgpt-kenya-workers/>
- [5] Stack Overflow, Temporary Policy: ChatGPT Is Banned, 5 décembre 2022 : <https://meta.stackoverflow.com/questions/421831/temporary-policy-chatgpt-is-banned>

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.

Archives 1978-2022 d'Elektor sur clé USB

Plus de 40 ans d'Elektor (de 1978 à 2022) sur clé USB

Cette clé USB (32 Go, USB 3.0) contient tous les numéros d'Elektor en français des années 1978 à 2022. Elektor propose à ses lecteurs des montages électroniques de conception professionnelle et aisément reproductibles, dans les domaines de l'électronique et de l'informatique appliquées. Il leur apporte également des informations sur l'évolution technologique et les nouveaux produits.

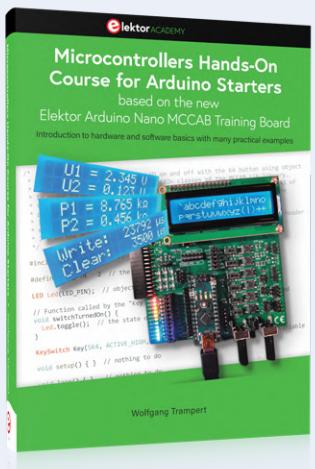
Prix : ~~199,95 €~~

Prix spécial : 99,95 €

 www.elektor.fr/20375



Offre groupée : Microcontrollers Hands-on Course for Arduino Starters



Ce cours pratique sur les microcontrôleurs pour les débutants Arduino montre comment vous pouvez réaliser vos propres projets avec un microcontrôleur même sans grande expérience en électronique et en langages de programmation.

Prix : ~~109,95 €~~

Prix spécial : 94,95 €

 www.elektor.fr/20440



Arduino Giga R1 WiFi



Prix : 89,95 €

Prix (membres) : 80,96 €

www.elektor.fr/20436

SDRplay RSP1A – Récepteur SDR 14 bits (1 kHz à 2 GHz)



Prix : 139,95 €

Prix (membres) : 125,96 €

www.elektor.fr/20421

Raspberry Pi Pico W en un tournemain



+
carte Pico W
GRATUITE



Prix : 47,95 €

Prix spécial : 39,95 €

www.elektor.fr/20424

Kit d'initiation au soudage Whadda WTS100



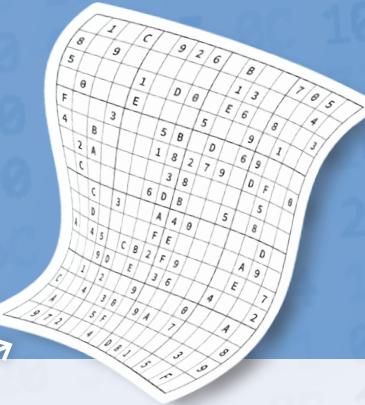
Prix : 49,95 €

Prix (membres) : 44,96 €

www.elektor.fr/20426

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 juin 2023** à l'adresse hexadoku@elektor.fr

LES GAGNANTS

La solution de la grille du numéro de mars/avril 2023 est **F13AB**.

La liste des gagnants est publiée ici : www.elektormagazine.fr/hexadoku

Bravo à tous les participants et félicitations aux gagnants !

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

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

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.



Équipement de test et de mesure fabriqué par des techniciens pour des techniciens avec plus de 40 ans d'expérience

100% fabriqué en Allemagne

Kurth Electronic offre des solutions pour la localisation de câbles dans presque tous les types d'installation, des instruments pour identifier et protocoler les réseaux de données (LAN), des instruments de haute qualité pour l'analyse des défauts de câbles et la qualification des lignes ainsi que des plateformes de mesure pour les réseaux à large bande sur la base du cuivre (xDSL), de la fibre optique (xPON / ETH P2P) ou sans fil (WiFi), y compris les services triple-play.

Notre kit mis en lumière : KE7207

KE7200 Testeur de réseau

- Localisation des défauts de câble
- Mesure de la longueur de câbles avec TDR
- Profils de câblage à partir d'une paire
- Base de données de câbles modifiable
- Test de charge de PoE / + / ++
- Scan du réseau / Ping stress test
- Enregistrer et consigner
- Jusqu'à 32 unités de KE7010 prise en charge
- En option : Détection de ponts (Bridge Tap)

KE701 Traceur de lignes

- Localisation de câbles, de fils et paires de fils sans contact
- Fonction de Portfinder ou Link-Blink en réseaux de LAN
- Signal-LED avec un filtre des signaux brouilleurs
- Protection contre les surtensions jusqu'à 500 V



896,- €

en plus des taxes et des frais de livraison

Vidéo KE7200



Article 0.49420

Vidéo KE701



Article 0.49567



Un cadeau !
Pour les premières
25 commandes !



Kurth Electronic GmbH | Membre du groupe GMC-Instruments

Équipement de test et mesure fabriqué en Allemagne

Tel. +49 (0) 7212 9755 0 | Email: sales@kurthelectronic.de | www.kurthelectronic.de

De nombreux outils de développement en un seul endroit

Provenant de centaines de fabricants fiables



Choisissez parmi une vaste sélection de produits sur mouser.fr/dev-tools



**MOUSER
ELECTRONICS**