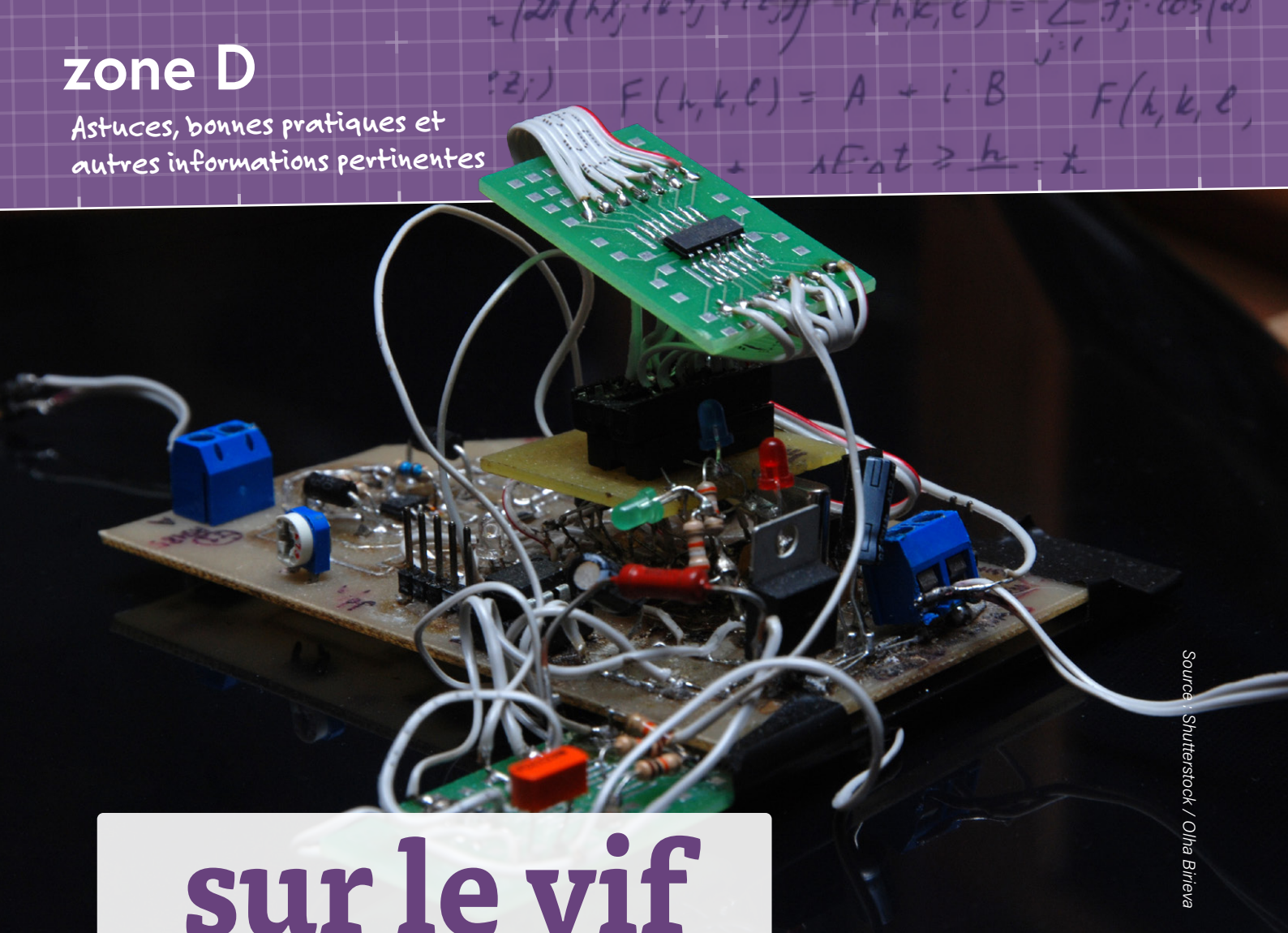


zone D

Astuces, bonnes pratiques et autres informations pertinentes



Source : Shutterstock / Olha Brieva

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'existait 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 réfléchi. J'ai récemment joué avec ChatGPT [3][4], l'agent conversationnel capable de pondre à la demande du code et des descriptions de schémas électroniques. 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, fussent-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.



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 !



Source : Shutterstock / jivacore

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>