

outils open-source

Simulateur d'antennes, gestion des composants, calculatrice, et bien plus

Tam Hanna (Hongrie)

Les programmes à code source ouvert sont amusants, et pas seulement parce qu'ils sont gratuits. Pour les personnes qui s'intéressent aux logiciels, ils offrent la possibilité de jeter un coup d'œil « sous le capot » d'un outil utilisé au quotidien. Voici quelques outils qui sont à la fois open-source et utiles.

Convertir NOW : conversion d'unités pratique

Les problèmes liés à l'utilisation des unités correctes font partie intégrante des équipes internationales, alors que le Britannique aime ses unités anglo-saxonnes, l'Européen continental utilise les unités SI. La perte de la sonde spatiale MCO illustre parfaitement le fait que de telles erreurs peuvent entraîner des problèmes graves et coûtant très chers.

Convertir NOW [1], développé par Damiano Ferrari, apporte une solution, car en principe, il s'agit d'un convertisseur d'unités ordinaire qui fait son travail en douceur, comme le montrent les **figures 1 et 2**. Convertir NOW est également intéressant d'un point de vue logiciel : le produit est basé sur l'environnement multiplateforme Flutter de Google. Toute personne souhaitant s'initier à ce nouveau système serait donc bien avisée d'en étudier le code.

Calculatrice de poche gratuite

Si votre smartphone Android peut être équipé de nouvelles applications, vous pouvez également installer **Calculator N+** [2]. Il s'agit d'un programme basé sur Symja [3] qui émule une puissante calculatrice scientifique sur votre smartphone (**figure 3**). En ce qui concerne l'utilisation pratique de cette application, on peut bien sûr toujours se demander si une calculatrice classique ne serait pas plus pratique à utiliser, et en fin de compte, c'est une question de goût.



Figure 1. Convertir NOW aide les deux parties à surmonter les difficultés liées aux unités SI...

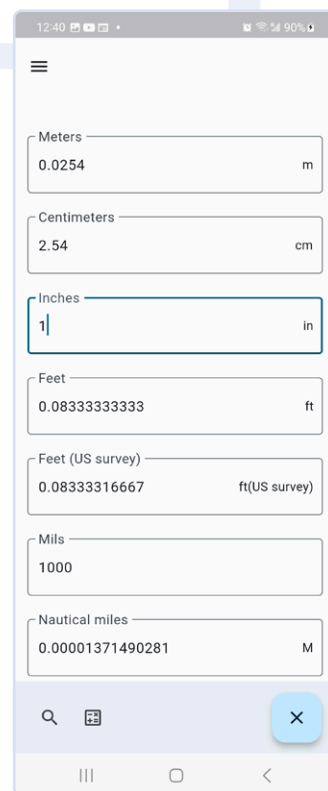


Figure 2. ... ainsi que dans la conversion des longueurs.

Gestion des composants

Autrefois, il s'agissait d'un travail courant dans de nombreux ateliers de développement : un gestionnaire de composants dédié qui s'occupait de l'inventaire et de l'approvisionnement des composants nécessaires. Ceux qui travaillent seuls contournent souvent ce problème par un usage excessif d'un distributeur : chaque nomenclature est achetée neuve, les composants restants sont stockés, donnés ou jetés à la poubelle.

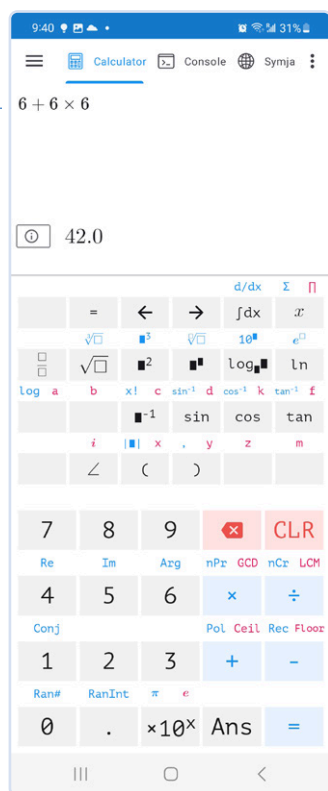


Figure 3. Toute ressemblance avec TI, Casio, etc., est purement fortuite.

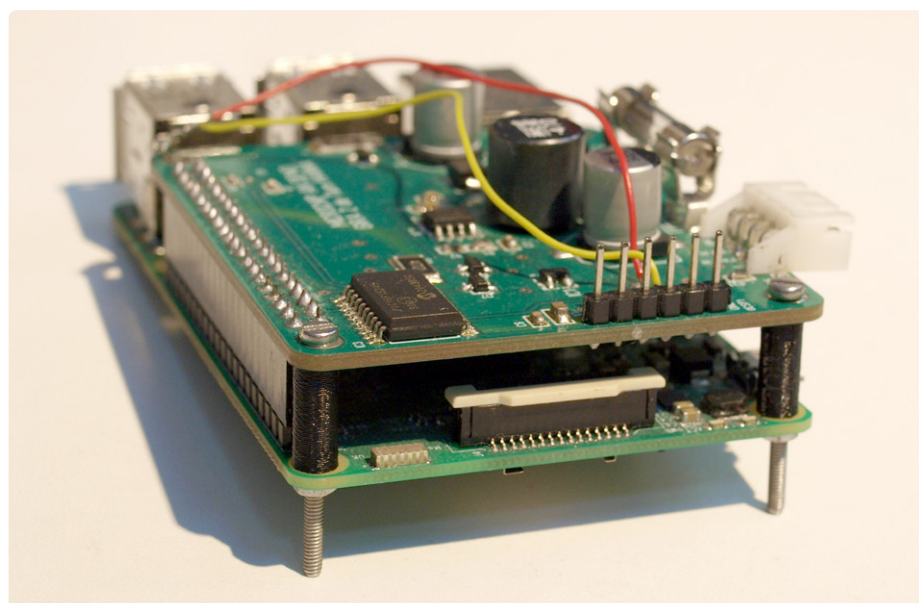


Figure 4. Ce Raspberry Pi obsolète et déclassé fournit divers « services locaux » dans le laboratoire de l'auteur.

Les solutions de gestion des composants sont très répandues dans les logiciels gratuits et payants. **InvenTree** et **PartKeepr** sont tous deux des logiciels libres, ce sont des applications web, ce qui rend leur configuration un peu plus compliquée. Plus précisément, un serveur web est nécessaire, auquel les (divers) clients accèdent ensuite par le biais d'un navigateur web.

Un moyen pratique de contourner ce problème est d'utiliser un Raspberry Pi ou un ordinateur équivalent. Sur Thingiverse, vous pouvez trouver différents boîtiers qui peuvent être vissés sous un bureau dans le laboratoire. Une fois installé, le Raspberry Pi dispose alors d'un logement discret et peu encombrant (figure 4).

La question de savoir si **InvenTree** (figure 5) ou **PartKeepr** (figure 6) est plus pratique peut être discutée en détail. Je vous recommande d'essayer les versions de démonstration [4][5]. Testez simplement les deux et décidez ensuite quel système correspond le mieux à votre style de travail.

Lors de l'utilisation de l'un ou l'autre système, l'effort investi doit être adapté à la situation actuelle. Par exemple, les deux produits offrent une gestion des nomenclatures, ce qui peut être exagéré, en particulier pour les petites entreprises spécialisées dans le prototypage.

Part	IPN	Revision	Description	Category	Stock	Price Range	Link	Last Stocktake
HS [Known]	HS			Bag1	10			
Cgpy [Ba]	Cgpy			Bag1	0			
Silicon Wire 10AWG Red			Silicon wire, 10AWG, red	Electronics/Wire	0	\$1.97 - \$3.5		
Silicon Wire 10AWG White			Silicon wire, 10AWG, white	Electronics/Wire	501	\$2.058 - \$1.029		
Silicon Wire 10AWG Black			Silicon wire, 10AWG, black	Electronics/Wire	203	\$2.5 - \$250		
Pinheader, 1x08x1.27mm			Male pin header connector, 1 rows, 3 positions, 1.27mm pitch, vertical	Electronics/Connectors/Pin Headers	11			
Pinheader, 1x08x1.27mm			Male pin header connector, 1 rows, 2 positions, 1.27mm pitch, vertical	Electronics/Connectors/Pin Headers	0			
Pinheader, 1x08x1.27mm			Male pin header connector, 1 rows, 4 positions, 1.27mm pitch, vertical	Electronics/Connectors/Pin Headers	19			2023-01-06
Pinheader, 1x08x1.27mm			Male pin header connector, 1 rows, 7 positions, 1.27mm pitch, vertical	Electronics/Connectors/Pin Headers	19			
Pinheader, 1x08x1.27mm			Male pin header connector, 1 rows, 5 positions, 1.27mm pitch, vertical	Electronics/Connectors/Pin Headers	25			

Figure 5. Les différences entre les interfaces graphiques d'InvenTree...

Part	Name	Description	Storage Location	Status	Condition	Stock	Min. Stock	Avg. Price	Freight	Internal ID
Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	0	0	0.000	0.000	1808 (1596)
Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	0	0	0.000	0.000	1809 (1597)
Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	0	0	0.000	0.000	1810 (1598)
Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	0	0	0.000	0.000	1811 (1599)
Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	Pin Header 2x4x1.27mm	0	0	0.000	0.000	1812 (1600)

Figure 6. ... et PartKeepr sont vraiment significatives.

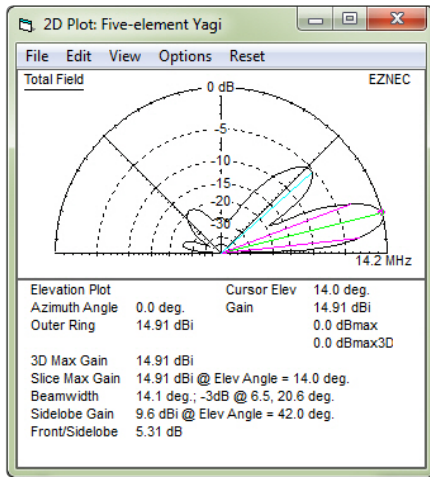


Figure 7. EZNEC facilite le fonctionnement de NEC.

La simulation permet de gagner du temps

Dans le domaine de la conception d'antennes en particulier, il est vrai que la simulation permet d'éviter les recours aux fabricants de matériel de radiofréquence. En général, les propriétaires des différents algorithmes de simulation sont conscients de cette relation, ce qui explique pourquoi les simulateurs d'antennes sont loin d'être bon marché.

Le simulateur d'antennes open source numéro un est **NEC**. Derrière ce terme ne se cache pas le géant japonais de l'électronique du même nom, mais l'abréviation **N**umerical **E**lectromagnetics **C**ode. Il s'agit d'un produit de simulation d'antennes développé par le Laboratoire national Lawrence Livermore, qui est partiellement open source au moment de la publication de cet article. Plus précisément, la version 2 du produit est open source, tandis que les versions ultérieures doivent être achetées.

Cette popularité est due en partie au fait qu'**EZNEC** [6], une interface utilisateur graphique pour le produit, est disponible gratuitement depuis environ un an (figure 7). L'*American Radio Relay League* (ARRL) a utilisé ce programme pendant de nombreuses années dans l'*Antenna Book* en tant que « système de démonstration », de sorte que le maniement du langage de script devrait être au moins quelque peu familier à de nombreux techniciens.

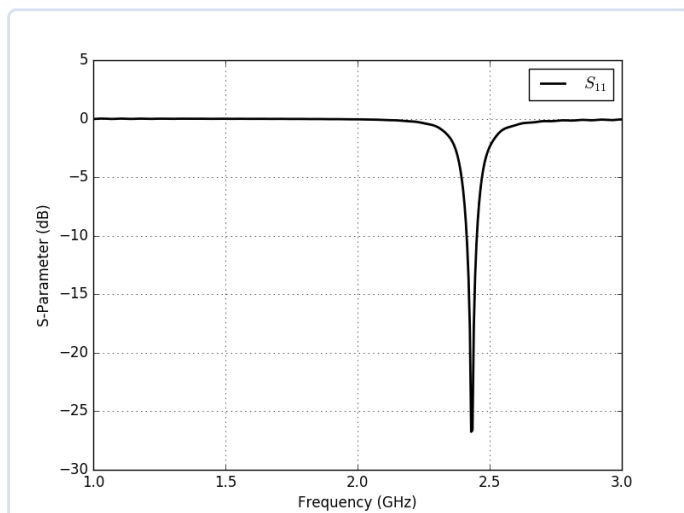


Figure 8. Les résultats graphiques d'Open EMS sont assez impressionnants. (source : [8]).

Avec **openEMS**, un produit développé au sein de l'Union européenne, un simulateur équivalent est disponible [7]. L'argument le plus important en sa faveur est la possibilité d'assembler les scripts de modélisation avec le langage Python, qui est largement utilisé. La figure 8, tirée de la documentation [8], montre une simulation du paramètre S sur une gamme de fréquences.

Enfin, il convient de mentionner brièvement **Elmer** (figure 9). Il s'agit d'un outil abouti [9], qui est optimisé pour la résolution de divers problèmes d'« éléments finis ». Il convient de noter que son utilisation nécessite une formation approfondie.

Bonus : OpenSCAD !

La conception de boîtiers et de moulages techniques dans les programmes de CAO classiques est une tâche que les électroniciens « vétérans », qui ont grandi dans un atelier de fabrication de prototypes, n'apprennent pas facilement au premier coup d'œil. **OpenSCAD** [11] est un système qui utilise une approche « ascendante » pour résoudre ce problème. Plus précisément, il utilise un langage de description rappelant Python et C, et compile les objets à générer à partir de « primitives géométriques » (figures 10 et 11). Le résultat de la compilation est un fichier STL, qui peut être réalisé par exemple avec différentes imprimantes 3D. ◀

VF : Laurent Rauber — 230282-04

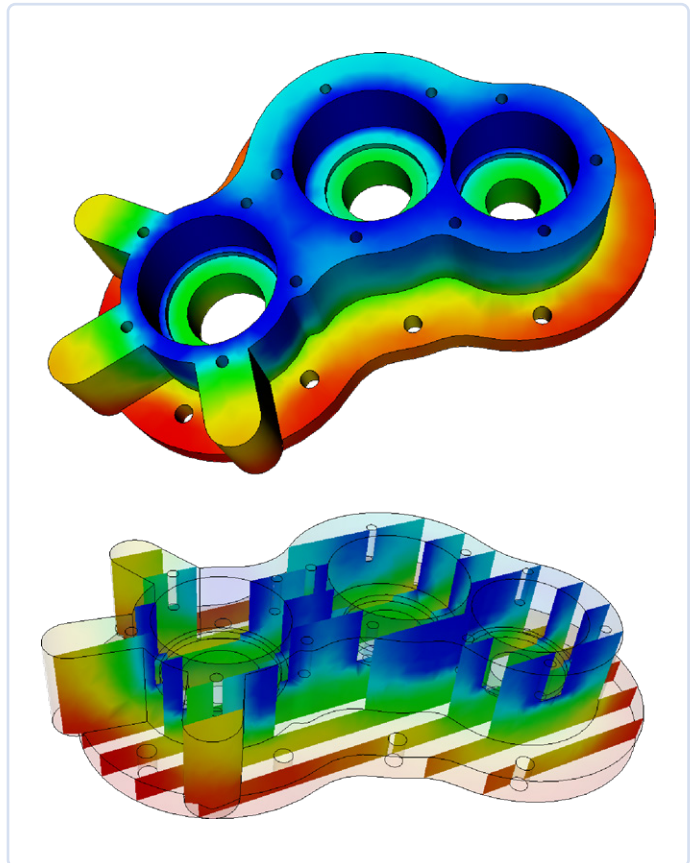


Figure 9. Cet objet n'a pas été conçu avec CATIA, mais avec Elmer. (Source : [10]).

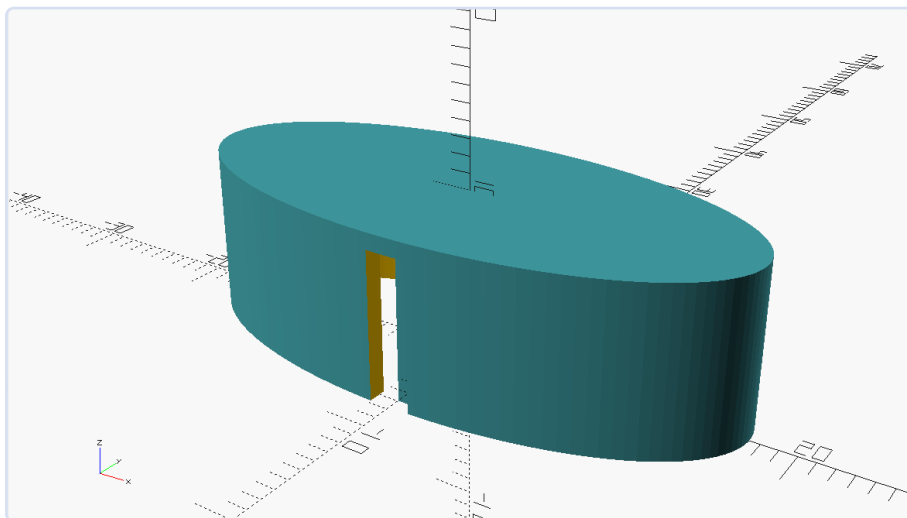


Figure 10. Le bouton de remplacement a l'air décent, tant de l'avant...

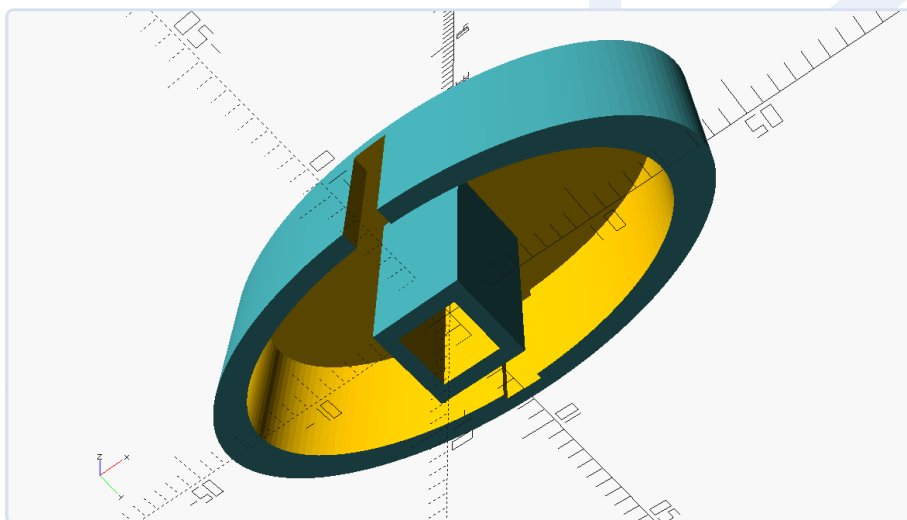


Figure 11. ... que de l'arrière.

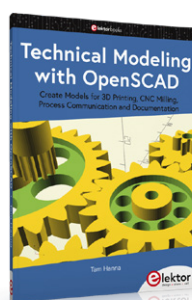
Questions ou commentaires ?

Contactez Elektor (redaction@elektor.fr).



Produits

► **T. Hanna, Technical Modeling with OpenSCAD**
(Elektor 2020)
<https://elektor.fr/19418>



LIENS

- [1] Converter NOW sur GitHub : <https://github.com/ferraridamiano/ConverterNOW>
- [2] Calculator N+ sur GitHub : <https://github.com/tranleduy2000/ncalc>
- [3] Bibliothèque Symja sur GitHub : https://github.com/axkr/symja_android_library
- [4] Démo InvenTree : <https://inventree.org/demo>
- [5] Démo PartKeepr : <https://demo.partkeepr.org>
- [6] Logiciel d'antennes EZNEC : <https://eznec.com>
- [7] openEMS : <https://openems.de>
- [8] openEMS, documentation d'une antenne patch : <https://bit.ly/43W71ak>
- [9] Elmer : <https://csc.fi/web/elmer>
- [10] Exemples d'applications d'Elmer : <https://csc.fi/web/elmer/application-examples>
- [11] OpenSCAD : <https://openscad.org>