



HAL
open science

Exemple de réalisation d'objets connectés avec outils open source Éclairage de vélos (Arduino, forum)

Arnaud Sivert, A. Faqir, B. Vacossin, F. Betin

► To cite this version:

Arnaud Sivert, A. Faqir, B. Vacossin, F. Betin. Exemple de réalisation d'objets connectés avec outils open source Éclairage de vélos (Arduino, forum). Colloque sur les Objets et systèmes Connectés - COC'2021, IUT d'Aix-Marseille, Mar 2021, MARSEILLE, France. hal-03593066

HAL Id: hal-03593066

<https://hal.science/hal-03593066>

Submitted on 1 Mar 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Exemple de réalisation d'objets connectés avec outils open source Éclairage de vélos (Arduino, forum)

A. SIVERT, A. FAQIR, B. VACOSSIN, F. BETIN arnaud.sivert@u-picardie.fr

Laboratoire des Technologies innovantes (L.T.I), équipe Énergie Électrique et Systèmes Associés (EESA)

U.P.J.V. Université de Picardie Jules Verne, Institut Universitaire de Technologie de l'Aisne GEII, 02880 SOISSONS

RESUME : Est-il possible de réaliser des objets connectés ou des systèmes avec des outils informatiques « open source » alors qu'à ce jour il n'existe aucun appareil électroménager commercialisé et libre de droit pour qu'il soit facilement modifiable ou réparable par l'utilisateur ? Quelle pourrait être la participation des écoles ingénieurs, des IUT informatiques, électriques, ... pour élever le niveau des applications présentées sur des forums, applications qui font appel à du codage de processeurs et à des outils « open source » ? L'article suivant présente la réalisation d'un éclairage de vélos avec de multiples options dont un tracker. Tous les programmes, les schémas électriques, les PCB (Printed Circuit Board : circuit imprimé) sont disponibles sur le forum Arduino et sont donc partageables par le plus grand nombre. L'article montre qu'il n'est pas si facile de faire des choix techniques et de code. Il faut, en effet, maîtriser de nombreux savoirs que les « makers » ne maîtrisent pas toujours (temps de calculs, consommation, batterie, Android). En effet, les projets sont souvent pluridisciplinaires et demandent de mettre en œuvre de nombreuses compétences. Enfin, les auteurs présentent les avantages et les inconvénients de partager ces connaissances sur des forums.

Mots clés : objet connecté, éclairage vélo, Arduino IDE, ESP32, Atmega 328, forum, outils open source.

1 INTRODUCTION

Dans un monde où tout doit être évalué, justifié, économisé, les données techniques sont cruciales pour faire des analyses, réguler, optimiser, rentabiliser des systèmes complexes. Dans ce contexte, les objets connectés permettent de recueillir ces données et d'optimiser un système pour obtenir une valeur ajoutée. Pour réaliser un objet connecté, il faut faire des choix de capteurs, d'actionneurs, de processeur, de communication, de cloud, de méthode d'analyse, de batterie ...



Fig 1. Contexte technique de cette étude

Dans ce cadre et depuis une quinzaine d'année, des outils « open source » [17] et « free » (Arduino [10]) existent, cependant ils manquent encore, à ce jour, de rigueur et d'options de débogage. Ces outils utilisaient seulement de petits microcontrôleurs tel que l'ATMEga328 qui commandait de petites cartes optionnelles appelées shields qu'il est nécessaire de câbler. Aujourd'hui, le microcontrôleur ESP32 est fortement utilisé grâce à ces communications intégrées en wifi, LoRa, Bluetooth. D'ailleurs, des fabricants tels que Heltec et TTGO distribuent une pléthore de cartes avec différentes options intègres autour de ce processeur tel que le GPS, carte SIM, support carte SD....

Les fabricants de shields créent des libraries pour mieux vendre leurs produits et faire gagner du temps aux développeurs.

On peut ajouter que la vérification de ces sous-programmes (Library) est rarement faite et il n'y a pas encore de certification pour minimiser les conflits entre ces programmes. Les

outils de simulations pour déboguer (point d'arrêt, visualisations de la mémoire...) commencent à être réalisés avec le nouveau compilateur « Arduino pro ». Cependant à ce jour, la phase de fiabilisation de ce compilateur n'est pas encore achevée.

De nombreux systèmes intégrés à notre quotidien, ne demandent pas un nombre d'entrées et de sorties important et des vitesses de calculs rapides... (Exemple : tout ce qui est électroménagers, chargeur de batteries NiMh avec afficheur, régulateurs de chaudières ou de radiateur électrique, domotique, instrumentation et commande de véhicules électriques tels que les trottinettes, les vélos électriques...). Pourtant, aucun de ces objets ne repose sur une solution « open source » pour être réparable pour laquelle, les cartes pourraient être à prix modiques.

Un autre exemple, toutes les plaques de cuisine vitrocéramique pourraient être régulées en température comme un four. Des capteurs de température à distance existent depuis plus de 10 ans avec des prix très modiques. On pourrait donc connaître les températures de cuisson, aisément, à l'aide de ce type de systèmes et mieux partager des recettes. Mais il serait aussi possible de réguler la puissance d'un four à micro-onde en fonction de la température.

Enfin, une certaine éthique relative à ces produits conduirait à retarder leur obsolescence au maximum notamment en facilitant leur maintenabilité car le coût des pièces serait faible.

Pourtant, à ce jour, il est possible de constater qu'aucun appareil électro-ménager ne fait appel à des supports de type « open source ».

Fort de ces constatations, depuis 2017, à l'IUT génie électrique de Soissons, de nombreux projets ont été réalisés avec une pédagogie discursive par des étudiants sur le forum Arduino [1].

Il est possible de citer des projets réalisés tels que la modification de systèmes électroménagers [2] avec du matériel « open source », la réalisation de la commande et de l'instrumentalisation de trottinettes électriques. Des « posts » ont été rédigés sur le site en français [3, 5] mais aussi en anglais [4].

De nombreux livres sur les outils et les cartes Arduino existent mais ils sont très vulgarisés et ne permettent pas de finaliser des systèmes. Ils sont très bien pour des collégiens.

En revanche, il n'apporte rien à des étudiants de DUT. De plus, les livres ne permettent pas de partager facilement du code.

Très peu d'articles sur l'Arduino entrent dans le détail des temps de calculs, de la gestion des timers, du chien de garde, de la précision des calculs qui est limitée par le nombre significatif de chiffres pris en compte avec la virgule flottante du compilateur, de la précision des Libraries (mathématiques, statistiques, calculs complexes, filtres digitaux, ...) [6].

Pour optimiser un système et vérifier son code, de nombreuses données doivent être mesurées pour vérifier la programmation.

La communication la plus simple pour vérifier le code est le moniteur et le traceur série via l'USB d'un PC. Mais aussi d'utiliser un afficheur LCD ou OLED. Le Bluetooth permet au processeur de communiquer sans fil à un PC ou un smartphone. Le wifi permet de dialoguer et de sauvegarder les données sur une borne et d'envoyer des mails ou des notifications. Le LoRa ou la carte SIM permet d'envoyer des données via un abonnement mais avec une limitation de données. Les fichiers sont souvent en mode .CSV (Comma-Separated Values) ce qui leur permet d'être traité aisément dans un tableur.

L'analyse ou la gestion des données n'est pas simple pour les personnes peu formées. Donc, avec un objet connecté sur le réseau, il est possible qu'un sous-traitant puisse traiter les données à distance avec une supervision et une télémaintenance. Cette supervision permet de vulgariser les données à des personnes utilisatrices.

L'avantage de la communauté « open source » en programmation est la communication via des blogs, des forums. Elle permet donc aux étudiants (et aux enseignants) de se former par eux-mêmes assez facilement.

L'« open source hardware » n'est pas cher car ceux sont des produits diffusés en très grande série. En revanche, si la programmation est assez facile, l'électronique est plus difficile à comprendre pour les makers [14]. Donc en général, les « makers » se limitent à la réalisation de petits systèmes, un peu gadget. Souvent aussi, cette communauté a aussi de grosses lacunes en automatique et en mathématique, ... Cela limite la complexité des réalisations. Il n'en reste pas moins que certains makers passionnés réalisent aussi des produits remarquables.

Pour motiver les étudiants, les projets doivent être utiles, innovants et avec une certaine puissance. Nous nous proposons de présenter un projet sur l'éclairage arrière des vélos. L'intérêt de ce projet est qu'il propose notamment de multiples possibilités électroniques, des développements en termes de communication et donc de codage. De plus, à ce jour, c'est un thème très tendance avec l'augmentation des vélotourneurs qui ont un besoin d'éclairage autant tôt le matin que tard le soir mais aussi en plein jour pour être bien vu alors que le marché propose des éclairages peu performants [8].

Ce projet utilise un microcontrôleur qui doit gérer la led en fonction de la charge de la batterie. Lors de la décélération, la led arrière peut indiquer que l'on va s'arrêter de façon automatique. Cet éclairage peut faire aussi office optionnellement de tracker GPS contre le vol.

Notons qu'il existe de nombreuses Start-up qui opèrent sur le sujet. Cependant, pour ces dernières, le design, le marketing passent souvent avant les performances de l'éclairage. En effet, la conception de l'environnement physique des leds et des optiques n'est pas si facile à maîtriser [13]. Enfin, la

réalisation est souvent complexe à cause du compromis masse, volume, autonomie, prix qu'il faut prendre en compte.

2 REALISATION ECLAIRAGE LED ARRIERE

L'utilisation d'un processeur pour un éclairage permet réguler facilement le courant dans la led pour avoir un courant constant dans la led donc un flux lumineux constant malgré la chute de tension de la batterie en fonction de la décharge.

D'ailleurs un ATMEGA328 peut être alimenté directement par une batterie lithium 18650 entre 4.2V et 3V.

De plus, le processeur peut gérer différents modes d'éclairage clignotant, pour augmenter l'autonomie et augmenter l'intensité lumineuse sans utiliser de dissipateur.

Enfin, ce processeur peut gérer un capteur d'accélération pour réaliser un feu stop et une alarme et gérer un GPS pour donner sa position en cas de vol.

Dans un premier temps, un Arduino pro mini qui consomme peu car il n'y a pas de driver USB, a été choisi. Un intérêt supplémentaire de ce choix vient de son encombrement : c'est la plus petite des cartes « open source »

2.1 Éclairage LED avec un Pro mini Atmega 328

La régulation du courant dans la led se fait avec une PWM à 64 kHz. La mesure de courant se fait avec des résistances shunt. Cette régulation permet de changer de fabricant de led et de modifier la consigne en fonction des désirs de l'utilisateur.

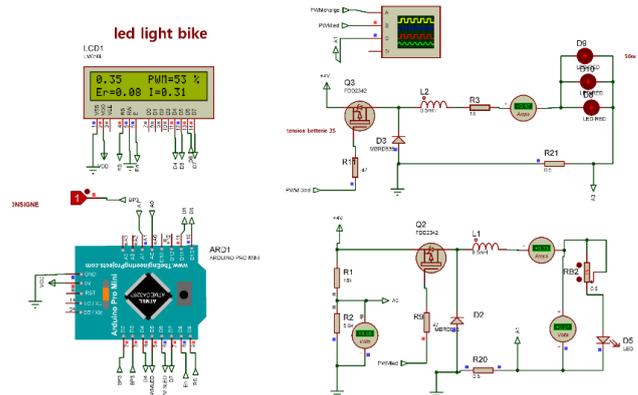


Fig 2. Schéma électrique d'une led de puissance de 1Watt (90lumen) et de 3 petites leds (65Cd) pour être vue sur le côté.

L'éclairage doit être testé car les caractéristiques limites des leds ne sont généralement pas données par les fabricants. De plus, il faut vérifier l'optique qui concentre l'éclairage en diminuant l'angle de diffusion. Voici les performances de la led pour 0.7 Watt qui a un demi-angle de 60° mais avec une optique obtient un demi-angle de 8° pour être vue à plus de 150 m.



Fig 3. Test de l'optique avec une LED CREE de 1W.

La puissance de la led est limitée en fonction de sa déperdition thermique et donc va dépendre du dissipateur et de la température ambiante.

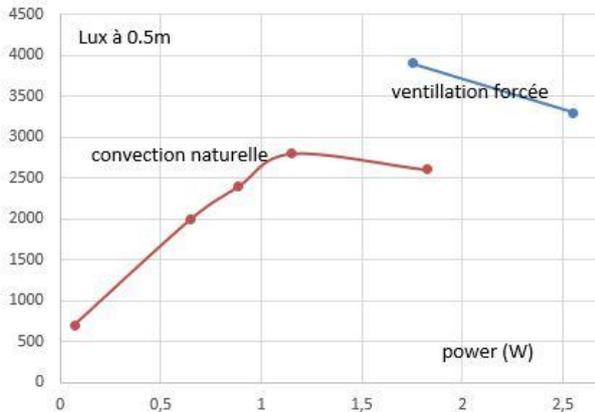


Fig 4. Performance de la LED CREE 1Watt en fonction de la puissance pour une résistance thermique de 21°C/W.

La fédération des usagers de la bicyclette (FUB) fait régulièrement des tests consommateurs d'éclairages de vélo [8] via les étudiants du département GEII (Génie Électrique et Informatique Industrielle de Soissons. Ces tests sont aussi présentés dans un forum de vélos [7] pour faire une communication entre les étudiants.

On peut noter que certains éclairages peuvent envoyer des SMS via le Bluetooth et le smartphone lors d'un accident donc pour une décélération brutale (il est possible d'utiliser un capteur accéléromètre mais aussi l'accéléromètre du smartphone car dans tous les cas il faudra qu'il y ait une communication entre le smartphone et l'éclairage)

L'outil de simulation de la régulation de courant est ISIS qui n'est pas un « open source ». Ce programme a peu d'évolution ces dernières années. D'ailleurs, il n'y a pas de possibilité à l'heure actuelle de simuler un ESP32 pour la version 8.7. Dans ISIS, s'il y a un certain nombre de capteurs qui sont simulables pour permettre de debugger, nombreux ne sont pas simulables.

Mais il est possible d'utiliser des outils « open source » (Fritzing, Virtronics, Simduino ...). Mais là encore c'est très limité.

L'histoire a aussi démontrée que les logiciels « open source » quand il commence à être au point deviennent payant ou doivent être utilisés en ligne avec un abonnement.

Les outils de simulation demandent une 20 heure minimum pour le maîtriser et pour connaître ces possibilités.

Dans ISIS, il est possible de simuler la communication série et de traiter les données via un terminal et un tableur.

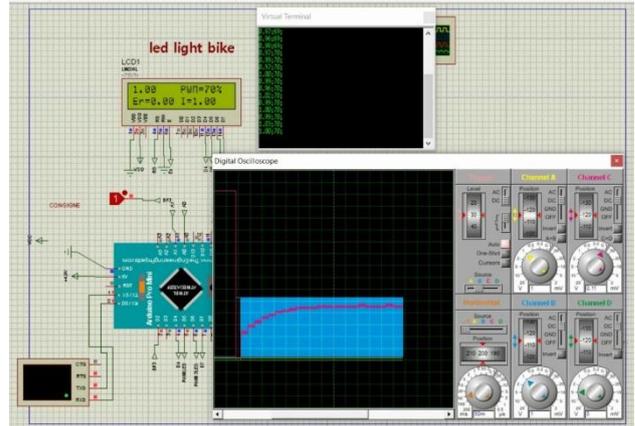


Fig 5. Schéma de simulation de la régulation de courant.

Pour le tracker, le système GPS est le NEO-6M. L'accéléromètre est un MPU 6050 pour réveiller le processeur si l'objet bouge et pour mesurer la décélération du véhicule pour que l'éclairage arrière devienne un feu stop automatique.

Le capteur d'accélération a un filtre interne mais, il doit être aussi filtré numériquement par un filtre passe bas du second ordre de 1 Hz, pour éviter des déclenchements intempestifs.

Le routage est assez facile par les étudiants car il n'y a pas beaucoup de composants. Tout est en CMS pour gagner de la place mais il faut créer de nombreux packages de composants dont ceux de l'Arduino dans ISIS et ARES.

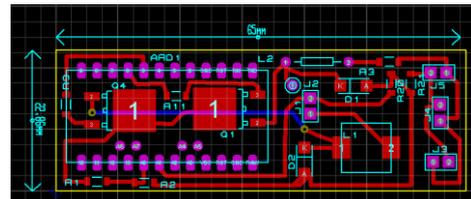


Fig 6. PCB à l'échelle 1 des nombreuses versions d'éclairages réalisés en fonction des options souhaitées

L'utilisation de ISIS permet de créer n'importe quels composants et de partager ces bibliothèques assez facilement.

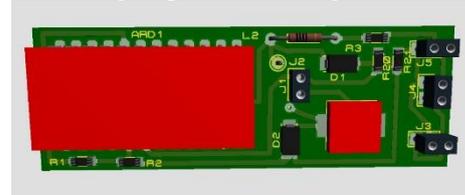


Fig 7. Le PCB en 3D de l'éclairage avec Arduino pro Mini

Une communication Bluetooth HC06 (10m) permet de changer la consigne, de connaître l'état de charge de la batterie, d'estimer le temps de fonctionnement. Le HC08 est légèrement plus cher mais consomme moins et sa distance de transmission est d'environ 60m [16] mais pas utile pour notre Le PCB de l'électronique va dépendre du boîtier et de l'accrochage. L'accrochage se fera sur la tige de selle du vélo via un boîtier fait à l'imprimante 3D comme on peut l'observer sur la figure suivante.



Fig 8. La réalisation avec communication par Bluetooth, pour paramétrer l'éclairage et vérifier la régulation

Pour gagner du temps, l'application « keuwl bluetooth electronics » est souvent utilisé par les étudiants car elle permet de réaliser très facilement le panneau et les commandes sur un smartphone et de s'en servir comme une supervision.

Il est aussi possible de réaliser une application avec Visual ou Android Studio en collaboration avec l'enseignant qui effectue ce module d'enseignement.

Pour connaître l'autonomie de l'IoT, il est facile de faire des mesures. Cela va dépendre du nombre d'octets transmis par le bluetooth

Pour diminuer le câblage et remplacer le HC06, un BLE NAN0 avec un Atmega 328 [9] a été testé. Cependant, la communication dans cette configuration, utilise seulement sa propre application et ne permet pas de gérer des données avec Android. De plus, cette configuration ne doit pas être confondue avec celle du Nano 33 BLE qui a un processeur Cortex-M4 et qui va consommer beaucoup plus que la carte Pro mini.

Dans tous les cas, il est tentant de concevoir un objet multi service (éclairage, compteur, GPS, tracker contre le vol....)

Dans ce cas, cette configuration sera intéressante pour gagner du temps de câblage et pour utiliser un processeur avec GPS intégré, un capteur accélération et une communication pour recevoir des notifications. Notons ainsi que l'ESP 32 a un wiki, le Bluetooth intégré et que des fabricants ajoutent aussi du LoRa ou une carte SIM...sur la même carte et évite du câblage.

2.2 Éclairage avec tracker et processeur ESP32

Le tracker doit envoyer la position toutes les 5 minutes pour connaître la position du véhicule en cas de vol. En effet, si l'on met le véhicule dans une cave ou sous un pont, le GSM ne fonctionne plus et il est même difficile de savoir où le vélo a été dissimulé pour la dernière fois.

La consommation de la partie communication doit être faible pour maintenir la connaissance de la position pendant au moins 2 jours.

Aujourd'hui, seul le fabricant, TTGO réalise des cartes toutes intégrées, GPS, OLED. Il faut encore faire un choix pour la communication

Il est toujours possible d'utiliser un petit schield de communication tel que la SIM800L ou le GPS sensor gy-neo6mv2 pour le GSM ou le Module LoRa SX1272 si l'on désire.

L'antenne qui permet d'avoir une bonne communication, est souvent problématique pour minimiser l'encombrement du système.



Fig 9. TTGO avec ESP 32 avec GPS, OLED et Lora intégré

Le schéma électrique de l'éclairage avec juste un TTGO Oled et le schield GPS GY-neo6mv2 qui a une antenne plate carré est le suivant.

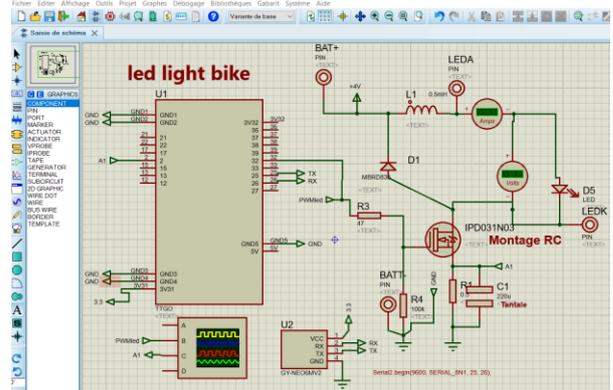


Fig 10. Schéma éclairage avec TTGO+GPS

Etant donné que l'éclairage arrière est alimenté par une batterie avec un mode clignotant de 50ms allumé et 950ms éteint.

A la place d'utiliser une routine d'interruption, qui provoque des temps d'échantillonnage et que le programme tourne en boucle pour pas grand choses. Un codage avec un mode veille pendant 950ms a été utilisé pour augmenter l'autonomie.

Mais comment est gérer la charge et la décharge des batteries par les modules TTGO ou Heltec ?

2.3 Consommation, Autonomie et Batterie

La consommation est très importante pour un IoT alimenté par une batterie. La carte TTGO et Heltec peuvent gérer une batterie lithium est mesurée sa tension mais ne peut mesurer son courant pour estimer son autonomie. Un ESP32 à 240 MHz consomme environ 120mA et le OLED 20mA.

La recharge de la batterie lithium est gérée via l'USB par un régulateur de courant. Mais l'arrêt de la décharge n'est pas géré par le micro qui passera en veille pour une tension batterie inférieure à 3.2V mais consommera toujours 5mA et va détruire la batterie en la déchargeant complètement.

Donc pour ne pas détruire la batterie, il faut utiliser un BMS (Batterie Management Security) et pour ne pas avoir de décharge profonde (d'ailleurs des notifications sont envoyées pour demander de recharger la batterie de l'IOT).

Donc, il faudra donc avoir des connaissances sur les batteries lithiums pour réaliser un IOT.

D'ailleurs, pour gagner en autonomie, il faut souvent mettre le processeur en veille profonde et le réveiller le micro grâce à une routine d'interruption.



Fig 11. ESP 32 TTGO OLED batterie, GPS

La programmation d'un ESP32 n'utilise pas le même codage que l'ATMEGA 328, il est donc nécessaire de refaire tout une partie du programme.

3 DIFFICULTES PEDAGOGIQUES

Les étudiants ont de nombreuses difficultés car ils ont une expérience faible des objets connectés et des processeurs. Voici une liste de leurs difficultés obtenues sur le forum Arduino où on peut observer l'évolution des choix, des résultats et des problèmes rencontrés, en analysant les posts envoyés par les étudiants.

Les difficultés sont nombreuses :

- connaître toutes les possibilités des cartes processeur ainsi que des différents shields pour faire les bons choix,
- maîtriser les outils de simulation, les outils du compilateur....
- maîtriser la communication des processeurs
- maîtriser le temps pour faire la réalisation car le module n'est que de 40 heures en IUT et limite la finalisation de leur projet.
- maîtriser le vocabulaire et la communication technique des datasheets.
- maîtriser la communication sur le forum qui demande une rhétorique concise. Cette communication oblige les étudiants à être rigoureux et l'enseignant à vérifier ce qui va être transmis.
- le forum permet de faire un état de l'art. Mais, la digestion de ce qui est écrit n'est pas facile surtout s'il y a beaucoup de pages. L'enseignant doit donc permettre d'aboutir à la version la plus élaborée et encadrer les explications. Dans tous les cas, l'étudiant doit faire plusieurs essais pour bien comprendre ce qui a déjà été fait auparavant.
- Enfin pour toucher un auditoire plus important, le projet doit être écrit en anglais.

4 CONCLUSION

Le projet d'éclairage vélo est intéressant car il est très pluridisciplinaire et de nombreuses options peuvent être mises en pratique. Le budget de réalisation est très modéré et peut être prêté aux étudiants pour pouvoir avancer même en distanciel. De plus, les outils étant open source, l'étudiant n'a pas de besoin de cracker certains softs pour pouvoir avancer sur le projet.

Mais, tout peut être simulé pour que les étudiants avance sans la partie matériel en distancielle.

Aujourd'hui, il y a très peu d'écoles qui réalisent des libraries (open source sur github) et qui partagent leur réalisation ou leurs projets sur un forum. Pourtant, un partage du travail permettrait d'augmenter le niveau de ces forums.

Aujourd'hui, il existe de nombreuses carte Arduino [10] qui ne sont pas tous compatibles entre elles et qui doivent être programmées différemment ce qui complique le codage et pour trouver l'information.

L'abonnement à des outils de programmation et de simulation propriétaire coûte relativement cher car on paye leurs développements et les nouvelles versions mais cela permet de programmer assez facilement sans perdre du temps avec des bugs.

Des questions restent en partie en suspens :

Est-ce que les outils « open source » s'imposeront grâce à des communautés de bénévoles et de passionnés ?

Certains fabricants de composants et de processeurs, ont aussi compris que pour mieux vendre leurs produits, ils doivent développer des programmes « open source ».

Des encyclopédies libres (Wiki) et de nombreux logiciels libres (open office, linux, VLC) ont aussi été un jour des utopies. Ils sont désormais devenus des références.

Quelle pourrait être la participation des écoles ingénieurs, des IUT informatiques, électriques, ... pour élever le niveau des applications présentées sur des forums, applications qui font appel à du codage de processeurs ou à des outils « open source » ?

Si les colloques permettent de s'auto former et de communiquer entre enseignants, est ce que des formations, des MOOC en distanciel se développeront sur les outils et sur les processeurs « open source » ?

De nombreux Fab-lab existe dans le monde entier, est-ce qu'ils contribueront au développement de solutions « open source » pour le codage des processeurs ?

En effet, ces structures peuvent apporter une valeur ajoutée, la connaissance et une maîtrise des différents composants facilitant les réalisations finales et d'éventuelles commercialisations.

Faut-il attendre des magazines d'électronique pour démocratiser les outils « open source » ?

Les magazines électroniques utilisent souvent des outils open source pour leur lecteur. Mais souvent ils reprennent des choses classiques qui sont déjà sur internet et il faut être membre pour accéder aux ressources complètes [15]. Ces magazines vendent des kits (composant et PCB) pour faire des réalisations avec déjà le codage tout fait.

Bibliographie

- [1] SIVERT, B. VACOSSIN, F. BETIN "Pédagogie communautaire discursive, application à l'éclairage LED de vélos », *CETIS Valencienne 2021*
- [2] [open source hardware electromenagers \(four électrique, micro onde.....\)](#)
- [3] [Commande et instrumentation de trottinette électrique 500W avec Arduino méga](#)
- [4] [500W electric scooter control and instrumentation with Arduino mega](#)
- [5] Arnaud Sivert, Vincent Boitier, Abdel Faqir, Franck Betin, Thierry Lequeu «Trottinettes électriques: Instrumentation, modélisation, simulation et contrôle sous Arduino » *Revue 3EI N 91°*, Janvier 2018, <https://hal.laas.fr/hal-01695380/document>

- [6] A. Sivert, A. Faqir, F. Betin, « Performance Arduino avec Outils (IDE, Matlab, ISIS, ...) pour Commander un éclairage arrière autonome de led de puissance » *Revue 3EI* N 97°, *Septembre 2019*
<https://www.fichier-pdf.fr/2019/03/06/arduino--led-eclairage-velo-v2/preview/page/3/>
 temps d' instruction atmega328, DUE, ESP32 compiler IDE
- [7] <http://velorizonta1.1fr1.net/t16874p925-eclairage-a-del-pour-velo-led-light-for-bike-light-emitting-diode>
- [8] <https://www.fub.fr/tests-eclairages>
- [9] <https://github.com/keywish/keywish-nano-plus/blob/master/BLE-Nano/Ble-Nano%20operation%20manual%20V.1.6.pdf>
- [10] <https://fr.wikipedia.org/wiki/Arduino>
- [11] [Filtre numerique RII, RIF....digital filter...FFT...atmega328, ESP32](#)
- [12] Arnaud Sivert, Abdel Faqir, franck Betin « TRAITEMENT NUMERIQUE DU SIGNAL (PETIT PROCESSEUR OPEN SOURCE ATMEGA 328, ESP 32) », *REVUE 3EI* N°103 2021
- [13] A. Sivert, F. Betin , B. Vacossin , P. Dondon “Innovative sustainable development teaching at university: Study of lighting systems for safety bicycle rides” *WSEAS TRANSACTIONS on ADVANCES in ENGINEERING EDUCATION* 2019
<https://www.wseas.org/multimedia/journals/education/2019/a145110-212.pdf>
- [14] <https://paris.makerfaire.com/maker/entry/33/>
- [15] [ESP32 - Getting Started | Elektor Magazine](#)
 Résultats de recherche pour "ESP32" - 226 résultats | [Elektor Magazine](#)
- [16] [HC-08 USER MANUAL V2.4](#)
- [17] [Open source — Wikipédia \(wikipedia.org\)](#)
- [Xin Yuan Electronic Technology Co., Ltd \(lilygo.cn\)](#)
[Shenzhen Xin Yuan Electronic Technology Co., Ltd \(lilygo.cn\)](#)