

SophiaConf 
2013 Du 1^{er} au 4 juillet

Open Source Engine Robotique Google
Web Informatique Cloud Utilisateurs
Open Hardware Communautés Saas Electronique
Arduino Développement Frameworks

4^{eme} édition

Open Source, Open Hardware, DIY, les nouveaux visages de l'industrie électronique.

L'électronique est-elle plus libre qu'avant ?

Jean DEMARTINI
PDG, DEMTECH SAS

`jean.demartini@demtech.net`

SophiaConf
2013

 **DEMTECH**.net
Remote Flexible Supervision

DEMTECH



Jean DEMARTINI – PDG
DEMTECH SAS
WTC2 – 120, route des Macarons
06560 SOPHIA-ANTIPOLIS

T. +33(0)493 746 289
M. jean.demartini@demtech.net

DEMTECH SAS a été créée in 2007 par Jean DEMARTINI alors Professeur des Universités à l'Université de Nice Sophia Antipolis et enseignant l'électronique, l'automatique, le traitement du signal, les architectures de calcul embarqué et la programmation ultra-basse consommation au département d'électronique de l'École Polytechnique Universitaire de Nice-Sophia Antipolis.

DEMTECH développe, produit et commercialise des systèmes de supervision flexible à distance pour la supervision de grands parcs de machines et d'équipements industriels en s'appuyant sur le réseau de téléphonie mobile.

Préambule

- La formule Open Hardware a le vent en poupe depuis quelques temps :
 - un nouveau concept ?
 - la résurgence de concepts anciens ?
- Le terme « Open » peut-il s'appliquer à tout ? Logiciels (libre) ? Objets manufacturés ? Molécules chimiques ? Produits naturels ? etc. ?
- Nous allons nous poser quelques questions concernant le Open Hardware (électronique).
 - Le nom « Arduino » doit parler à beaucoup.

Électronique ?

- Objet = Composants + Circuit
- Objet = Composants + Composants Programmables + Circuit
 - PAL, PLA, FPGA, PLD, EPLD, CPLD, etc.
 - apparition des HDL : Verilog, VHDL, System C, ...
 - apparition de l'IP (forme de software)
- Objet = Composants + Composants Programmables + μ Contrôleur + Circuit
 - apparition du firmware (logiciel enfoui)
 - apparition des noyaux d'exécution

Open Hardware électronique ?

- Une nouveauté ?
- Une « extension de l'Open Source » ?
 - issu lui-même de concept de « Logiciel Libre »
- Le logiciel est essentiellement *immatériel* ; le hardware est essentiellement *matériel*.
 - avec l'existence du logiciel enfoui (firmware) qui complique le paysage depuis environ 40 ans.
- Quant au Droit de propriété d'une chose :
 - en ce qui concerne son aspect matériel ?
 - en ce qui concerne son aspect immatériel ?

Droit de propriété

- Pour le côté matériel d'une chose achetée, 3 aspects :
 - **usus** : son usage est libre (utiliser, céder, ...),
 - **fructus** : on peut tirer profit de son usage,
 - **abusus** : on peut la détruire.
- Pour protéger l'investissement du fabricant de la chose, l'acquisition de la chose ne concerne que son aspect matériel. L'aspect immatériel, description de la chose, n'est pas cédé.
 - Le brevet protège l'aspect matériel,
 - Le droit d'auteur (copyright) protège l'aspect immatériel dont la copie est facile (depuis l'invention de la photocopie).

Open Source

- La copie et la distribution d'un logiciel sont faciles.
- La production d'un programme exécutable à partir de ses sources est facile (en général).
 - d'autant plus que le nombre des plateformes à supporter est (assez) faible (PC, Macintosh, Smartphone).
 - Les outils de productions de programme peuvent être également Open Source.
 - `gcc` et son environnement de travail est un bon exemple.
- L'ouverture d'un logiciel-outil largement transversal :
 - est utile à un grand nombre,
 - peut être soutenu par un large communauté.

Open Hardware

- Dans le cas du matériel,
 - la chose, aspect matériel, peut être protégée par des brevets,
 - la documentation de la chose, aspect immatériel, peut être protégée par un Copyright, une marque, un modèle, etc.
- Seul l'aspect immatériel peut être couvert par une licence de type Copyleft.
- Tentons une définition :
 - un matériel ouvert est un matériel suffisamment documenté pour qu'on soit capable :
 - d'en réaliser des copies, et les vendre (attention aux brevets éventuels),
 - de le modifier (hardware et software),
 - de l'améliorer (hardware et software).

Licences

- Tous les documents textuels ou graphiques peuvent être couvert par une licence de type Creative Commons.
 - schémas,
 - fichiers Gerber, BOM, Fichiers de positionnement, etc.
 - documentation
- Des licences mieux ciblée sur le hardware sont en cours de rédaction. En voici 2 à titre d'exemple :
 - TAPR OHL : Tucson Amateur Packet Radio Open Hardware Licence
 - OSHW : Open Source HardWare

TAPR OHL

(Open Hardware Licence)

- You may modify the documentation and make products based upon it.
- You may use products for any legal purpose without limitation.
- You may distribute unmodified [copy of the] documentation, but you must include the complete package as you received it.
- You may distribute [sell] products you make to third parties, if you either include the documentation on which the product is based, or make it available without charge for at least three years to anyone who requests it.
- You may distribute modified [copy of the] documentation or products based on it.

Former OSI (Open Source Initiative) president Eric S. Raymond expressed some concerns about certain aspects of the OHL. According to Raymond, the license has "lots of problems" and that it "strips the word 'distribution' of its normal meaning, assuring lots of contention over edge cases." There are also concerns that the Open Hardware License may merely place the design and/or idea into the Public Domain by effectively publishing it before securing the benefits of patent protection.

Questions ouvertes

- Open Hardware concerne un équipement assez bien documenté pour qu'on puisse le programmer à sa guise ?
 - où est l'électronique dans cette définition ?
 - le firmware, s'il y en a, est-il nécessairement Open Source ?
- Open Hardware concerne un équipement assez bien documenté pour qu'on puisse :
 - en produire des copies (pour les vendre),
 - le modifier,
 - l'améliorer.
- ... et assez simple pour que cela soit possible avec des moyens légers et sans être obligé d'utiliser des outils logiciels propriétaires ?
 - il est interdit de verser du perchlorure de fer dans son évier !
 - le montage de surface est à la portée d'un amateur.

Brève histoire de l'Open Hardware électronique

- de 1915 à 1955 : la Radio.
 - Tous les aficionados ont construits leur propre récepteur de radio (poste à galène avec ou sans amplificateur). Les schémas circulent librement. Ce n'est pas encore l'époque du circuit imprimé, on utilise les « arêtes de poisson ». On fabrique les composants à la main.
- de 1950 à 1970 : la HiFi et la Radio-Commande
 - Tous les aficionados ont construit leur propre chaîne HiFi. Les schémas circulent librement. Les boutiques de composants au détail fleurissent (et la VPC aussi). On fabrique son circuit imprimé dans une cuvette. Les kits apparaissent : Radioshack, Heathkit, etc.
- de 1965 à 1975 : l'ordinateur personnel – Homebrew Computer Club
 - Tous les aficionados ont monté un Altair (celui pour lequel Bill Gates a développé son premier interprète Basic et Gary Kildall CP/M), un H8, un Apple I, ...

Et aujourd'hui ?

- Deux pistes dans la tradition :
 - les kits pour l'électronique environnementale : l'éco-système Arduino est l'exemple cité à chaque fois.
 - les systèmes de maquettage et d'expérimentation.
- Je lance un troll :
 - Raspberry PI est un merveilleux nano-PC, mais j'ai de la peine à considérer que le programmer est « faire de l'électronique ».
 - À propos, Raspberry PI est-il Open Hardware ?
- Développer et construire des cartes électroniques interconnectables pour constituer des ensembles pour le loisir ou l'expérimentation relève de l'électronique.

Produits électroniques commercialisés ?

- Un environnement normatif assez lourd modère la distribution de produits Open Hardware sans précaution :
 - Sécurité des personnes (directive UE basse-tension),
 - Absence de nuisances et d'interférences (directives UE CEM et R&TTE)
 - Labels et Certifications (marquage CE)
 - Autorisation de mise sur le marché
- Produire des objets pour les commercialiser est vite assez compliqué.
 - C'était beaucoup plus facile dans les années (19)60. On mettait dans un sachet tout ce qu'il fallait, on rédigeait une petite documentation en précisant que l'électricité c'est dangereux, on l'envoyait par la poste contre remboursement et tout était dit.



- DEMTECH a entrepris, il y a deux ans la conception d'une plateforme de développement d'objets communicants autonomes avec l'intention de « l'Open-Sourcer »
 - pour développer des réseaux de capteurs sans fil par exemple.
 - La philosophie Arduino est bien sympathique. Elle nous a servi d'inspiration.
- Questions :
 - y-a-t-il bénéfice à Open-Sourcer ?
 - quoi Open-Sourcer ?
 - quoi ne pas Open-Sourcer ?
 - comment Open-Sourcer ?

Objectifs OCCOX

- Il s'agit de développer une plateforme permettant de développer rapidement de nouveaux modules
 - pouvant fonctionner en stand alone éventuellement,
 - pouvant être assemblés pour construire un système opérationnel,
 - pouvant être intégrés sous forme de combos pour créer un produit spécifique.
- Il ne s'agit pas de reproduire une variante Arduino en changeant simplement de processeur (pour conserver la compatibilité avec les shields existants)
 - Architecture mono-processeur à périphérie extensible.

Architecture OCCOX

- Architecture SMP parallèle
 - autour d'un bus standard : RS485 – protocole Modbus RTU,
 - permettant le développement de systèmes ultra-basse consommation,
 - conçue pour un développement **bare-metal** des firmwares.
- Une première version a été développée autour de processeurs à cœur AVR (Atmel) puis remplacée par un version développée autour de processeurs à cœur ARM Cortex
 - Cortex M3 en général,
 - Cortex M4 pour le traitement du signal.
- Parmi les raisons du choix Cortex, on trouve :
 - le nombre des sources disponibles,
 - le nombre des environnements de développement Open Source disponibles.

Et alors ?

- La mayonnaise est difficile à faire prendre :
 - OCCOX est très (trop) ciblé ?
 - OCCOX est réservé aux professionnels ?
 - Pas assez de communication ?
- Pourquoi Arduino est un (grand) succès ?
 - Arduino est très simple ?
 - Arduino ne nécessite que de la programmation ?
 - La communication est bonne ?

Retour d'expérience

- Au moins en électronique, l'engouement pour un objet (récepteur radio, chaîne HiFi, micro-ordinateur, ...) a cessé dès lors que l'objet correspondant du commerce était devenu (par les progrès de son industrialisation) trop économique pour justifier l'effort de sa construction en mode DIY.
 - sauf pour les passionnés qui les construisent pour les étudier, les améliorer, les modifier, les échanger, mais pas pour les utiliser.
 - pour eux, l'aspect « Open Hardware » pourrait être important, mais n'est pas nécessaire dans le cadre d'une copie personnelle.

Open Hardware

- Remarque : Open Hardware n'est finalement revendiqué que par ceux qui proposent des objets programmables (au sens software).
- Open Hardware recouvre surtout la documentation qui va permettre :
 - de modifier le logiciel enfoui,
 - de développer un logiciel enfoui qui crée de nouvelles fonctionnalités, un nouvel usage.
 - N'est-on pas revenu, de fait, au logiciel Open Source et à la documentation libre ?

Open Hardware ?

source : <http://www.raspberrypi.org/faqs>

What hardware documentation will be available?

Broadcom don't release a full datasheet for the BCM2835, which is the chip at the heart of the Raspberry Pi. We will release a datasheet for the SoC which will cover the hardware exposed on the Raspi board e.g. the GPIOs. We will also release a board schematic later on.

But I want documentation for <hardware X>!

Other documentation may be released in future but this will be at the Foundation's discretion.

But I demand the documentation for the chip. Give it to me!

To get the full SoC documentation you would need to sign an NDA with Broadcom, who make the chip and sell it to us. But you would also need to provide a business model and estimate of how many chips you are going to sell.

Open Hardware

En guise de conclusion

- **Pro :**
 - Une modularité extrême permet de réaliser des assemblages étonnants.
 - La programmabilité des composants permet de créer des fonctionnalités par le software.
 - Des composants puissants permettent d'utiliser les mêmes outils de développements que ceux utilisés traditionnellement.
- **Cons :**
 - La généricité entraîne une consommation énergétique élevée.
 - Raspberry PI sans accessoire consomme 3,5W.
 - La conformité CE peut être difficile à établir
 - par manque de documentation.

Conclusion

provisoire et ouverte

- Les nouveaux composants, cartes, modules :
 - déclenchent l'enthousiasme,
 - renforcent la créativité,
 - accélèrent l'expérimentation.

- Ces nouveaux éléments ne facilitent pas spécialement l'industrialisation.
 - En fait c'est l'industrialisation qui a permis de les démocratiser.
 - Il constitue alors un vrai marché.

Merci pour votre attention.

La discussion est ouverte.