

# Raspberry, Arduino, ESP32, Nucleo, oh my ! Vers la domotique et au delà !

Arnaud Février, Anouch Hovsepian  
IUT Aix-Marseille, 163 avenue de Luminy, 13288 Marseille cedex 09  
Courriel : arnaud.fevrier@univ-amu.fr  
Courriel : anouch.hovsepian@univ-amu.fr



*Résumé* : La domotique, les objets connectés et les smart cities sont de plus en plus présents chez les opérateurs télécoms, les stages de nos étudiants et dans la vie de tous les jours. Nous mettons en place, à Luminy, une extension d'un Fablab pour motiver les étudiants et leur permettre de maîtriser ces nouveaux domaines. Nous introduisons ces éléments dans les cours ce qui permet d'établir une relation étroite entre les enseignements d'électronique, de transmission, de traitement du signal et de télécoms.

## 1 INTRODUCTION

Le mot domotique commence à être un mot du passé! Maintenant, nous parlons d'Internet des objets, des objets connectés, des smart-cities. Le programme pédagogique des IUT Réseaux et Télécoms, va probablement intégrer ces notions d'une manière ou une autre. Au delà de l'intérêt de surfer sur des mots nouveaux poussés par la propagande, ces concepts peuvent permettre de redynamiser nos enseignements classiques.

Nous nous appuyons sur un fablab local, situé à Aix en Provence. Ce fablab fournit un environnement de développement de qualité pour des makers. Il dispose de nombreuses machines outils (imprimantes 3D, Fraiseuses et tours numériques, découpeuse laser...) et d'outils de développements sur cartes micro-contrôleurs et leurs capteurs et actionneurs.

La distance est trop grande entre le Fablab et l'IUT à Luminy, aussi nous avons proposé un essaimage. Nos étudiants disposent donc d'un accès favorisé au FabLab, mais la distance reste gênante. Donc, nous avons mis en place une extension qui propose une orientation réseaux et télécoms. Le Fablab n'a que peu de spécialistes dans ces domaines et ont demandé de l'aide, en particulier sur MQTT et le Lorawan.

Pour nos étudiants, nous mettons en place plusieurs axes :

- Utilisation des GPIO;
- Interaction entre une carte et un PC;
- Installation d'un programme;
- Utilisation des périphériques radio (Modulation AM, FM);
- Utilisation de Gnradio et des logiciels dérivés.

## 2 LE JEU DE CARTES

Nous avons commencé à utiliser plusieurs types de cartes. Nous pouvons classer les cartes que nous avons utilisées en deux grandes familles. Leur point commun, c'est le prix modéré : entre moins d'un euro à moins de cinquante euros.

Nous avons testé deux types de cartes : les cartes supportant un environnement GNU-Linux et les cartes moins puissantes soit sans système d'exploitation, soit avec un RTOS.

## 2.1 Les cartes GNU-Linux

Le prix des cartes GNU-Linux commencent à 10\$ pour l'OrangePI Zero (Quad-core Cortex-A7, 512Mo RAM), passe par 35Euro pour un raspberry classique et il n'y a pas vraiment de limite supérieure.

Les critères de choix vont être : popularité, performance, nombre d'entrée sorties, compatibilité avec un brochage populaire (Raspberry), libre hardware, compatibilité du matériel avec une distribution majeure.

Le développement est extrêmement simple sur ces cartes, c'est un système GNU-Linux classique avec une faible consommation et des entrées sorties en nombre.

## 2.2 Les cartes microcontrôleur

Le prix des cartes à microcontrôleur est aussi très variable. Il commence à 4\$ (ESP32S WIFI Bluetooth, Dual Core CPU 520Ko RAM) et monte à 10Euros pour un compatible Arduino.

Les critères d'achat sont assez similaires que pour les cartes GNU-Linux.

Pour développer une application sur ce genre de cartes, il y a plusieurs possibilités :

**IDE Arduino** : cet environnement a été conçu pour faire des applications simplement sur Arduino. Beaucoup de cartes sont compatibles avec cet environnement. Cet environnement est facile d'accès, mais limité. Il ne permet pas d'exploiter à fond les possibilités des cartes. Pour nos étudiants, c'est largement suffisant.

**Eclipse** Il est possible d'utiliser Eclipse pour certaines cartes. Pour que ce ne soit pas trop facile, il faut parfois télécharger une version modifiée spécifiquement pour une carte. Bref, c'est lourd, mal conçu et peu adapté pour nos étudiants.

**build-essential** La façon optimale pour développer des applications complexe, c'est d'utiliser les outils classiques de développement GNU-Linux (gcc, make, bison, autoconf. . .)

**IDE spécifiques** Certaines cartes viennent avec un environnement de

développement spécifique, privatif, qui impose de nombreuses restrictions, ceci ne convient pas à nos étudiants.

## 3 GNU RADIO

Il existe des composants à très bas coût, comme le couple récepteur RXB6 et émetteur JMR-TX-1. Ce couple permet de réaliser des transmissions numériques en AM très facilement. Le prix est d'environ un euro par couple.

D'autres matériels existent, comme les clef TNT RTL2832 qui permet de recevoir n'importe quel signal (ou presque) de moins de 2GHz et de réaliser la démodulation en logiciel à l'aide de GnuRadio.

D'autres composants existent, comme le hckrf, qui permet d'émettre et de recevoir les signaux hertzien jusqu'à 6GHz.

Gnu radio permet d'assembler des briques permettant de réaliser toutes les fonctions de transmission en utilisant de nombreux périphériques. Il existe aussi de nombreux logiciels, depuis les rtl\_sdr en ligne de commande (analyse spectrale, réception FM. . .) jusqu'aux logiciels graphiques d'analyse de signaux comme Universal Radio Hacker qui facilite le décodage de signaux de télécommandes de garage (AM ou FM).

## 4 CONCLUSION

L'accueil des étudiants devant les possibilités offertes par un FabLab est très bon. Nous souhaitons que les étudiants puissent avoir des envies pour réaliser des montages personnels. Ces montages peuvent être l'occasion pour montrer de nombreux aspects du programme pédagogique actuel et proposer des évolutions pour le futur.