



RAPPORT_KNOCKLET

Rapport du système Knocklet

VERSION	01
DATE	01/02/2017
AUTEUR	CATHERIN, CORDONNIER DOFFAGNE, LAW, NAPOLY



Objet

Ce document a pour but de conclure le projet de fin d'études Knocklet. Ce projet s'est déroulé de Septembre 2016 à février 2017. Il a été réalisé par un groupe de cinq étudiants de l'ECE Paris :

- Félicien CATHERIN, Majeure SI
- Paul-Adrien CORDONNIER, Majeure SI
- Flavien DOFFAGNE, Majeure EE
- Nicolas LAW, Chef de Projet, Majeure SE
- Pierre-Yves NAPOLY, Majeure SI



Table des matières

1	Présentation	5
2	Etat de l'art et acteurs du marché.....	6
3	Rappel du cahier des charges.....	9
3.1	Le bracelet	9
3.2	Les modules émetteurs/récepteurs	9
3.3	La centrale domotique	9
4	Réalisations techniques.....	10
4.1	Bracelet.....	11
4.1.1	Décomposition du bracelet	11
4.1.2	Éléments matériels du bracelet.....	11
4.1.3	Éléments logiciels du bracelet	13
4.1.4	Algorithme de fonctionnement.....	13
4.1.5	Résultats	18
4.2	Modules émetteurs/récepteurs	20
4.2.1	Communication avec JEEDOM	20
4.2.2	Résultat.....	20
4.3	Centrale domotique	21
4.3.1	Interface utilisateur	21
4.3.2	Fonctionnement du plugin	23
5	Difficultés rencontrés	25
5.1	Accéléromètre	25
5.2	Transmission sans fil du bracelet	25
5.3	Miniaturisation du système.....	26

Table des figures

Figure 1 : Le marché de la maison connectée	6
Figure 2 : CA de la domotique selon les applications.....	6
Figure 3 : Les acteurs du marché de la domotique	7
Figure 4: Membres de l'alliance EnOcean	7
Figure 5 : Différents constructeurs de centrales domotiques.....	7
Figure 6 : Les concurrents Amazon et Google.....	8
Figure 7 : Smart Remote de SevenHugs	8
Figure 8: Chemin du signal lors de l'envoi d'une commande.....	10
Figure 9: Interface du cœur embarqué du bracelet.....	11
Figure 10: Plateforme NUCLEO-L073RZ	12
Figure 11: Accéléromètre MMA8452QT	12



Figure 12: Shield X-NUCLEO-IDB05A1	12
Figure 13: Séquencement général du bracelet	13
Figure 14 Incidence des différents paramètres.....	14
Figure 15: Pulse Threshold	14
Figure 16:Pulse Time Limit	15
Figure 17:Pulse Latency.....	15
Figure 18: Définition des Knocks valide.....	15
Figure 19: Séquencement du processus Accéléromètre.....	16
Figure 20: Séquencement du processus Bluetooth Low Energy	17
Figure 21: Logs du bracelet	18
Figure 22: 1ère version du bracelet	18
Figure 23: Schémas électriques du bracelet.....	19
Figure 24: 2ème version du bracelet.....	19
Figure 25: Routage du bracelet	19
Figure 26: Deux Knocks reçu par la Raspberry Pi	20
Figure 27: Page du plugin Knocklet sur Jeedom.....	21
Figure 28: Fenêtre de configuration des scénarios	22
Figure 29 : Fonctionnement du plugin	23
Figure 30: Courbe d'accélération des axes X, Y et Z.....	25
Figure 31: Première version du chemin du signal lors de l'envoi d'une commande	25
Figure 32: Essai de braisage pour un accéléromètre	26

1 Présentation

La domotique envahit de plus en plus nos maisons. Il est simple et peu onéreux d'obtenir des lampes et thermostat connectés afin de mieux gérer sa consommation. Néanmoins un grand nombre de ces systèmes ne sont utilisables qu'au travers d'une télécommande ou d'une application mobile. Dans le cas d'un éclairage, il est souvent plus simple de se déplacer pour appuyer sur l'interrupteur plutôt que de chercher son téléphone, le déverrouiller, lancer l'application et éteindre l'appareil.

Knocklet est un bracelet destiné à actionner les différents objets connectés d'un système domotique simplement et intuitivement. En portant ce bracelet dans votre maison vous pourrez par exemple allumer votre télévision, une lampe, retrouver votre téléphone ou encore fermer tous les volets de votre domicile d'un simple tapotement sur n'importe quelle surface.

Le bracelet, lorsqu'il est porté par l'utilisateur, détectera un **Knock** (un rythme tapé sur une surface de la maison) et l'enverra à l'**émetteur/récepteur** Knocklet le plus proche. Ce dernier permettant la **localisation** dans la maison transmettra ensuite l'information à la **centrale domotique** qui déterminera quel objet de la maison il faudra actionner en fonction de quel Knock a été effectué, quel bracelet l'a détecté et de quel E/R a transmis l'information. Ces Knocks **personnalisables** peuvent être associés à une action relative à la pièce dans laquelle vous vous trouvez (taper deux fois peut allumer la lampe la plus proche par exemple) ou effectuer la même action partout dans votre maison (fermer les volets et éteindre toute les lumières avant de quitter le domicile par exemple).

La **personnalisation** des Knocks se fera via l'application de gestion de votre centrale domotique. Un plugin développé par l'équipe Knocklet permettra alors la configuration de vos Knocks de manière simple, ergonomique et correspondant à l'utilisateur. En effet chaque utilisateur peut utiliser les codes qu'il a lui-même configuré.

Knocklet s'inscrira donc en tant qu'actionneur pour système domotique innovant permettant une simplification de l'utilisation de vos différents objets connectés et transformant certaines fonctionnalités considérées comme superflues en réel outil pour votre confort.

Ce projet s'inscrit dans une valorisation partenariat avec le **groupe Ingenico**, leader mondial des terminaux de paiement, qui recherchait au travers de cette réalisation obtenir un **travail de recherche** sur les technologies sans fil actuelles et notamment le Bluetooth Low Energy.

Ce rapport présentera tout d'abord un bref état de l'art des systèmes domotiques existant, que ce soit en termes d'appareils, d'actionneurs et de contrôleurs ainsi que les grands acteurs de ce marché. Nous rappellerons ensuite le cahier des charges puis nous décrirons notre prototype final en expliquant les choix technologiques associés à celui-ci.



2 Etat de l'art et acteurs du marché

La domotique est de nos jours un marché très prometteur, en nette progression au fil des ans. En effet, la France était le deuxième marché mondial (900m d'euros) en 2015 ! Même si elle peine encore à trouver sa place dans les foyers des personnes de plus de 35ans, aujourd'hui 30% des Français possèdent déjà un objet, et 56% des autres prévoient de s'équiper d'un ou plusieurs de ces objets. En effet, le peu de connaissance des objets, ajouté au prix et au manque de compréhension des avantages de ces produits sont des freins au développement de ce commerce.

Quel que soit le pays, pour un tiers des consommateurs, la première motivation à l'achat de produits pour maison connectée est le souhait d'automatiser leur maison. Leur deuxième motivation est la sécurité, suivie par le souhait de faire des économies d'énergie.

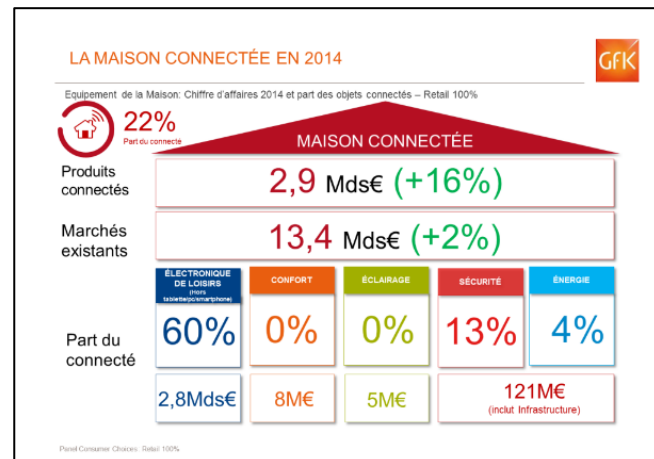


Figure 1 : Le marché de la maison connectée

On peut distinguer trois principaux types d'usages qui motivent les consommateurs dans l'achat :

Le désir d'avoir une maison « prête » dès l'arrivée chez soi : par exemple que le chauffage et l'éclairage soient activés et que les programmes favoris soient déjà sélectionnés sur la télévision

L'activation de la maison en « mode nuit »: extinction automatique des lumières, de tous les dispositifs de coupure du courant, fermeture automatique des portes, des volets, mise en route des caméras de surveillance et de l'alarme etc.

La possibilité d'avoir accès à toutes sortes de divertissements, musique et multimédia disponibles où et quand le consommateur le veut: toute la musique en ligne, TV et profils sur les médias sociaux synchronisés sur tous les appareils, afin que l'utilisateur puisse continuer à écouter la musique qu'il écoutait dans sa voiture en rentrant chez lui ou encore une télévision connectée qui reconnaisse chaque membre de la famille et recommande des contenus appropriés en fonction de chaque

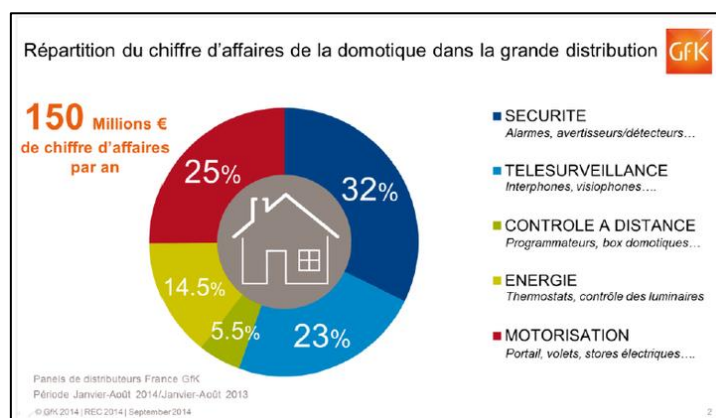


Figure 2 : CA de la domotique selon les applications



personne, etc.

On peut rajouter que les Français manifestent leur goût pour la cuisine par leur intérêt à ce que la maison soit prête pour recevoir les amis. Dans ce type de scénario, les ampoules connectées sont ajustées à l'ambiance souhaitée, la musique est déjà activée, le four est préchauffé et des alertes et des contrôles à distance du frigo permettent de savoir ce qui est présent ou manquant ou quels aliments arrivent à leur date de péremption. (La cuisine est sur la première place du podium des pièces à connecter en premier pour 57% de sondés, et ce, tous pays confondus !)

Ce marché qui semble si prometteur est déjà composé de nombreux acteurs. Nous allons nous intéresser plus précisément aux fabricants de centrales domotiques et d'objets connectés.

Citons en premier lieu l'allemand *Hager*, se spécialisant dans le matériel électrique pour l'installation électrique modulaire, la gestion d'énergie, l'alarme et la domotique de la maison et des bâtiments.

De même, nous pouvons parler de *Legrand* qui lui se spécialise dans les installations électriques et les réseaux d'informations : La programmation des tâches et le pilotage à distance constituent le cœur de cible de Legrand. Enfin, nous pouvons parler d'*Hestia*, qui se concentre plus sur différents actionneurs et télécommandes à application domotique.



Figure 3 : Les acteurs du marché de la domotique



Figure 4 : Membres de l'alliance EnOcean

Il n'est pas facile de parler de tous les fabricants tant leur nombre est important, citons alors en vrac Velux, Somfy, Hörmann, Renson, Vachette, Honeywell...dont la plupart sont membres de l'alliance EnOcean.

En nous concentrons plus précisément sur les différents appareils actuellement disponibles, on peut parler en premier des centrales domotiques, cœur de tout système de maison intelligente. Citons parmi tant d'autres Eedomus, Homecenter, Jeedom, Veraedge et SomfyBox. Nous avons choisi d'utiliser une centrale domotique Jeedom pour notre projet car elle est open-source et l'ajout de modules ou de plugins y est facilité.



Figure 5 : Différents constructeurs de centrales domotiques

Du côté des actionneurs, il en existe de tous les types, avec des formes et fonctionnalités différentes. Présentons en premier lieu *Knocki* : Ce module, une fois posé sur une surface, la rend capable de détecter les coups portés à cette dernière par un utilisateur. C'est de ce module que nous est venue l'inspiration première pour créer Knocklet.

Le marché des actionneurs domotique accueille en son sein de grands noms de milieu divers qui s'attellent à la domotique. Citons Google Home de Google, Amazon Echo d'Amazon ou encore Spot de Nvidia. Tous trois ont développé un actionneur domotique vocal, qu'elles ont présenté lors du CES 2017 à Las Vegas.



Figure 6 : Les concurrents Amazon et Google

Nous avons aussi remarqué *NIU*, un bouton connecté capable de déclencher plusieurs actions selon les clics. Fonctionnant sur le même principe, *FLIC* est aussi un bouton-actionneur, plaçable où vous le souhaitez dans votre maison.

Notons aussi l'arrivée de la Smart Remote par *SevenHugs*, présentée au CES 2017 comme une télécommande domotique universelle, elle pour principal intérêt de pouvoir contrôler n'importe quel objet connecté en le pointant avec la télécommande.



Figure 7 : Smart Remote de SevenHugs

Pour rester dans le domaine des objets connectés, il en existe une pléthore qui ont un lien avec la domotique et dont les applications sont diverses et variées : Economies d'énergie, sécurité (cambriolages, inondations, évanouissements...), gestion du sommeil et des dépenses énergétiques du corps pour les plus classiques. Mais on retrouve des applications moins conventionnelles, allant des pots de fleurs connectés, aux friteuses intelligentes et miroirs analysant votre peau, en passant même par le matelas connecté détectant les adultères ! Et pour les plus tête en l'air d'entre nous, il existe des prises connectées pour retrouver ses objets perdus dans la maison.

Toutes ces informations nous permettent de nous dire que le marché de la domotique est un marché large, ouvert, où il est facile de trouver sa place si l'on propose des systèmes novateurs, des moyens de contrôles originaux, ou simplement, un nouveau design.

3 Rappel du cahier des charges

Voici un rappel du cahier des charges sur chaque section du projet. Tous les détails se trouvent dans les documents « CDC_KNOCKLET_XXXX » prévu à cet effet.

3.1 Le bracelet

Les fonctionnalités principales que doivent effectuer le bracelet sont :

- Le bracelet réagit au tapotement sur une surface.
- Le bracelet reconnaît les différents tapotements.
- Le bracelet est portable et fonctionne sur batterie rechargeable.
- Le bracelet communique avec les émetteurs/récepteurs Knocklet, en permettant une localisation à l'aide de ces modules.
- Le bracelet est configurable.

3.2 Les modules émetteurs/récepteurs

Les fonctionnalités principales des modules émetteurs/récepteurs sont :

- Le module reçoit des messages envoyés par les bracelets.
- Le module communique avec la centrale domotique.

3.3 La centrale domotique

Les fonctionnalités principales de la centrale domotique sont :

- La centrale reçoit des événements transférés depuis les modules. Un événement est la combinaison composé du nombre de Knocks ; d'un identifiant de bracelet ; et d'un identifiant de module.
- La centrale gère une base de données associant les événements Knocks avec une ou plusieurs actions sur le système domotique.
- La centrale fournit une interface de gestion permettant d'associer un événement avec une ou plusieurs actions sur le système domotique.
- La centrale propose d'ajouter un module et un bracelet facilement au système Knocklet.



4 Réalisations techniques

Cette section concerne toutes les réalisations techniques effectuées lors du projet.

Pour rappel, lorsqu'un Knock est détecté par le bracelet, celui-ci est transféré aux émetteurs/récepteurs. Ces derniers ont pour fonction de transférer le message à la station domotique en rajoutant leur ID afin que celle-ci puisse connaître depuis quel salle la commande est envoyée. La station quant à elle n'a plus qu'à déchiffrer la combinaison nombre de Knock + ID afin d'interagir avec le bon l'appareil connecté.

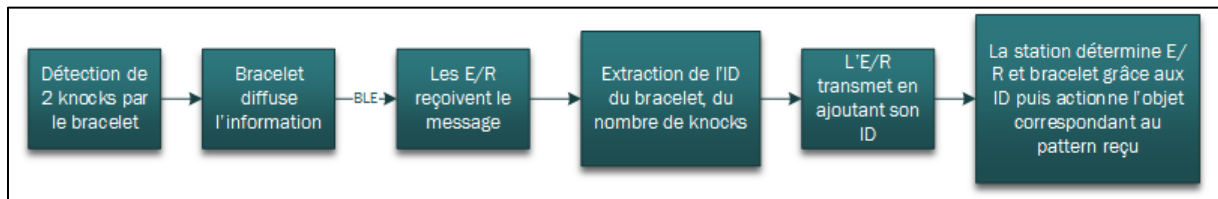


Figure 8: Chemin du signal lors de l'envoi d'une commande

Les réalisations sont décrites ci-dessous dans l'ordre suivant : Bracelet, Emetteur/Récepteur puis Centrale domotique.

4.1 Bracelet

Le bracelet est l'actionneur du système Knocklet. Celui-ci permettra à l'utilisateur de contrôler son système domotique à l'aide de Knock.

Cette partie commencera dans un premier temps par une décomposition du bracelet afin d'éclaircir les différents modules qu'il contient, suivi d'une explication sur les matériaux et logiciels utilisés. Nous détaillerons par la suite le fonctionnement de l'accéléromètre, puis de la transmission sans fil. Enfin, nous montrerons les résultats obtenus.

4.1.1 Décomposition du bracelet

Le bracelet est décomposé de deux fonctionnalités bien distinctes : La détection de Knock et la transmission sans fil.

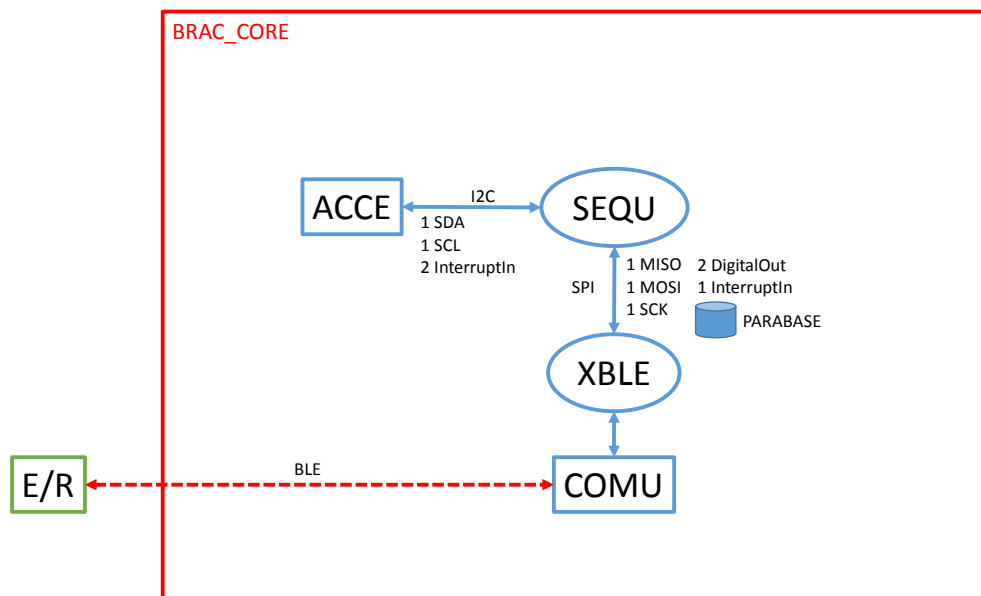


Figure 9: Interface du cœur embarqué du bracelet

Cette première fonctionnalité est effectuée à l'aide d'un **accéléromètre** tandis que la deuxième est fournie grâce à un **module Bluetooth Low Energy**.

4.1.2 Éléments matériels du bracelet

Afin d'effectuer le prototypage du bracelet, celui-ci est composé d'une plateforme de développement, d'un accéléromètre et d'un « shield » Bluetooth. Les détails sont décrits ci-dessous.

4.1.2.1 Plateforme de développement STM32 NUCLEO-L073RZ

La plateforme de développement NUCLEO-L073RZ de la marque *STMicroelectronics* permet le prototypage composé d'un microcontrôleur STM32L073RZ. Celui-ci étant composé d'une architecture *ARM® Cortex®-M0+*, sa faible consommation énergétique fait de lui le processeur idéal pour tout prototypage pour des objets connectés. Cette plateforme inclut un debugger et programmer via le ST-LINKV2-1. Enfin, celle-ci permet l'ajout de nombreux « shield »

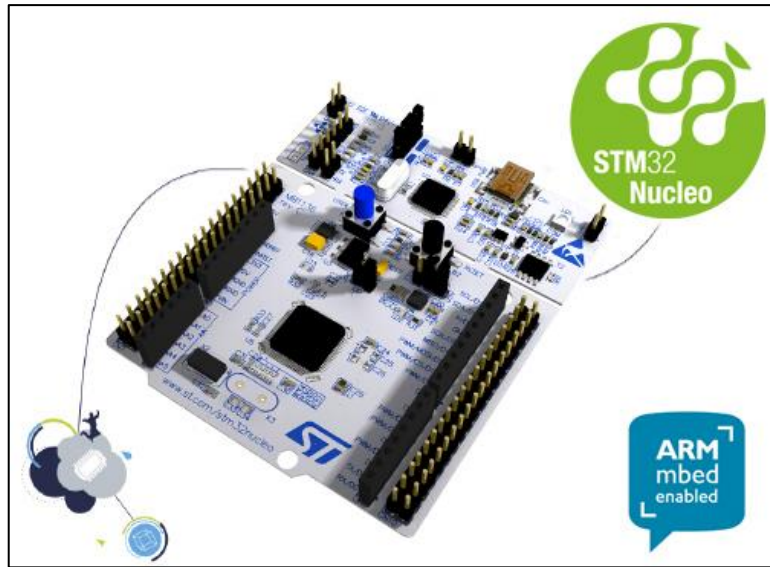


Figure 10: Plateforme NUCLEO-L073RZ

4.1.2.2 Accéléromètre MMA8452QT

L'accéléromètre MMA8452QT de la marque *NXP* est un accéléromètre faible puissance, trois axes, avec une résolution de 12 bits. Cet accéléromètre s'accompagne de fonctions intégrées avec des options flexibles programmables par l'utilisateur, configurables grâce aux deux broches d'interruption. Grâce à des échelles sélectionnables par l'utilisateur de ± 2 / ± 4 / $\pm 8g$, celui-ci nous permettra de détecter le Knock effectué par l'utilisateur. Celui-ci communique via protocole **I2C**.

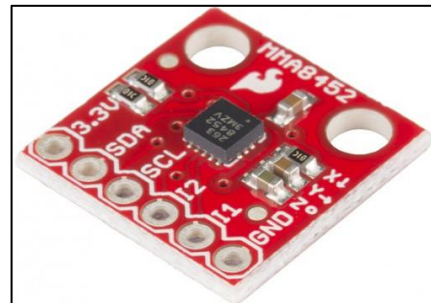


Figure 11: Accéléromètre MMA8452QT

4.1.2.3 Shield BLE X-NUCLEO-IDB05A1

Le shield X-NUCLEO-IDB05A1 de la marque *STMicroelectronics* est un module permettant l'ajout d'une fonctionnalité Bluetooth Low Energy à la plateforme de développement STM32 NUCLEO. Celle-ci communique via le protocole **SPI**.



Figure 12: Shield X-NUCLEO-IDB05A1

4.1.3 Éléments logiciels du bracelet

Le cœur embarqué est développé grâce au logiciel Visual Studio 2013 de *Microsoft* ainsi que l'extension VisualGDB permettant l'ajout du support de nombreux microcontrôleurs. Le langage utilisé est le **C++**.

Le cœur embarqué est programmé à l'aide de 2 librairies bien distinctes :

- **Mbed** : Cette librairie permet de gérer toute les entrées et sorties du microcontrôleur de type ARM Cortex-M. Dans le cas du bracelet, elle permet de communiquer avec le module BLE et l'accéléromètre (Pour en savoir plus, voir <https://www.mbed.com/en/>).
- **BlueNRG** : Cette librairie contient toute les fonctions propres au chip BLE qu'utilise le bracelet. Elle permet au microcontrôleur via une liaison SPI de communiquer avec d'autres appareils Bluetooth auquel il est connecté. Dans le cas présent, il diffusera le message afin que les modules Émetteurs/Récepteurs puissent recevoir le nombre de Knock.

4.1.4 Algorithme de fonctionnement

Le fonctionnement du bracelet est décomposé en deux parties. La première est composée du séquençement de l'accéléromètre. La deuxième est composée du séquençement Bluetooth.

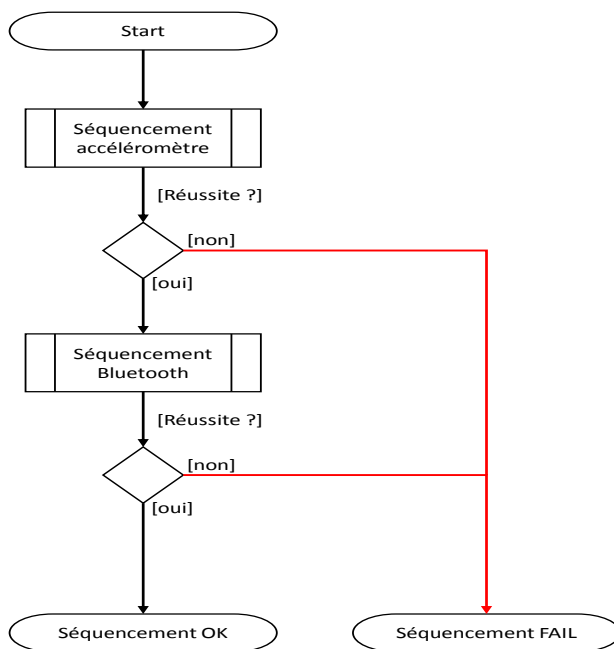


Figure 13: Séquençement général du bracelet

4.1.4.1 Accéléromètre

Nous avons utilisé la librairie suivante afin de contrôler l'accéléromètre :

<https://developer.mbed.org/components/MMA8452Q-Triple-Axis-Accelerometer/>

Afin de bien isoler nos Knocks, une configuration sur l'accéléromètre est nécessaire. Celle-ci se fait principalement à l'aide de 3 paramètres : **Pulse Time Limit**, **Pulse Latency** et **Pulse Threshold**.

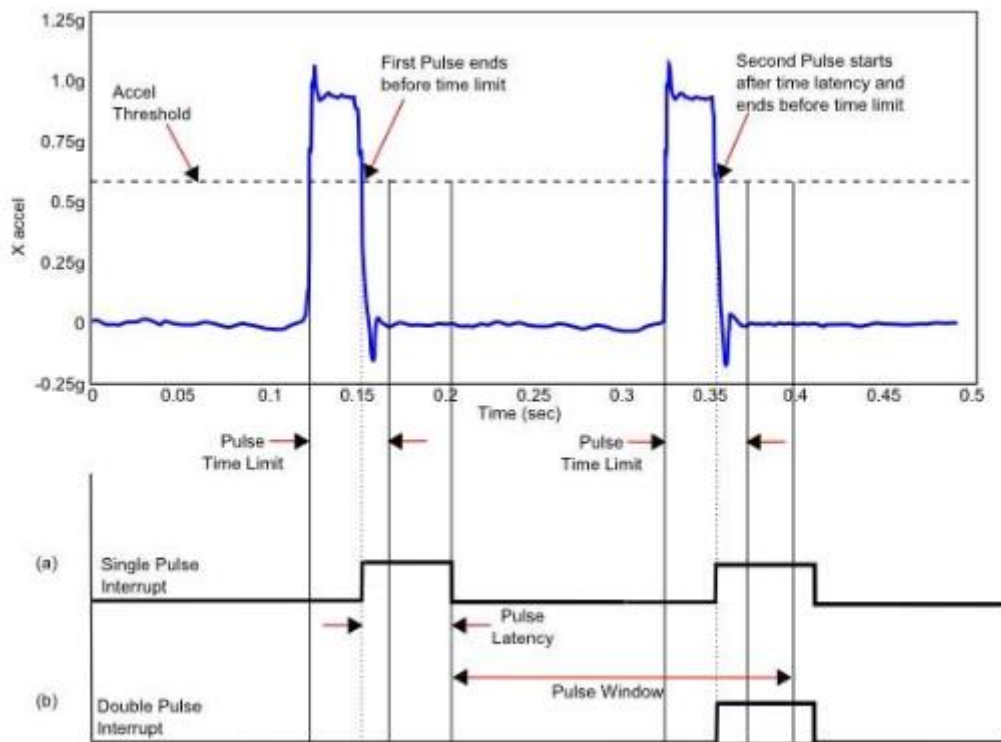


Figure 14 Incidence des différents paramètres

Pulse Threshold configure l'intensité minimal d'une pulsation afin qu'elle soit considérée comme étant un Knock. Dans notre configuration actuelle, l'accélération minimale est calibrée à 0,5g.

```
// Pulse threshold for X,Y,Z
// Threshold X: 0.5g
if (ACX_acce.writeRegister(MMA8452_PULSE_THSX, 0x10) != ACX_ACCE_GOOD)
{
    return false;
}
```

Figure 15: Pulse Threshold



Pulse Time Limit configure la fenêtre de temps dans laquelle un Knock peut se produire. Afin de limiter les faux positifs, nous avons limité ce temps à 20ms.

```
// Pulse Time Window 1: 20 ms, Normal Mode, No LPF
if (ACX_acce.writeRegister(MMA8452_PULSE_TMLT, 0x20) != ACX_ACCE_GOOD)
{
    return false;
}
```

Figure 16:Pulse Time Limit

Pulse Latency configure le temps suivant un premier Knock, où la détection d'un autre Knock ne peut se faire. C'est le temps minimal à respecter entre deux Knocks. Nous avons configuré ce temps à 250ms.

```
// Pulse Latency Timer: 250 ms
if (ACX_acce.writeRegister(MMA8452_PULSE_LTCY, 0x64) != ACX_ACCE_GOOD)
{
    return false;
}
```

Figure 17:Pulse Latency

C'est en modifiant ces paramètres et en faisant de nombreux essais que nous sommes arrivés à une configuration permettant de détecter un Knock lorsque nous le souhaitons, c'est-à-dire lorsque nous faisons un geste qui nous semblait naturel, pas trop forcé. De même, afin de limiter les faux positifs et d'affiner les différentes commandes du bracelet, nous avons choisi de nous limiter à un certain nombre de coups, compris entre 2 et 6.

```
// Blindage de la valeur
if (ACX_ntap > ACX_ACCE_NMAX || ACX_ntap == 0 || ACX_ntap == 1)
{
    // Trop de tap détecté
    Printf("ACX__process: Taps ignore");
}
else
{
    // Tap valide
    Printf("ACX__process: Taps OK");
}
```

Figure 18: Définition des Knocks valide

Une fois cette configuration terminée, le séquençage de l'accéléromètre est décomposé en 3 parties. En effet celui-ci est sensible aux **interruptions**. Un programme fonctionne toujours de façon linéaire hormis lors de la réception d'une interruption. Celle-ci stop le programme actuel, effectue la fonction de l'interruption puis reprend le programme où il s'était arrêté. Ces deux interruptions sont **désactivées** lors de la diffusion du message via Bluetooth.

Le séquençage fonctionnera alors de la façon suivante :

- Si une interruption Knock (tapotement détecté) est reçue, une vérification sur le Timer est effectuée suivi d'une incrémentation du nombre de Knock à envoyer.



- Un Timer fonctionne comme un chronomètre. Celui-ci mesure le temps écoulé lors de son activation. Grâce à celui-ci, lorsqu'un Knock est détecté, si le Timer est inactif (détection du premier Knock), celui-ci s'active ; sinon celui-ci redémarre de zéro.
- Si une interruption Bouton (utilisateur qui appuie sur le bouton du bracelet) est reçue, le bracelet passe en mode configuration.
- Le mode configuration consiste à envoyer une valeur spéciale aux émetteurs/récepteurs. Lorsque ceux-là recevront cette valeur, les modules enverront à la centrale domotique non pas le nombre de Knock reçue mais une commande spéciale permettant la configuration d'un nouveau bracelet/module.
- Le séquençement « normale » de l'accéléromètre est constitué dans un premier temps d'une vérification sur la réactivation des interruptions.
- Dans un second temps, une vérification sur la valeur du Timer est faite. Si celui-ci dépasse le délai maximum, une analyse sur le nombre de Knock reçue suit. A la fin de cette analyse, une remise à zéro sur la valeur du nombre de Knock est effectuée.
- Si le nombre de Knock est valide, le bracelet désactive ses interruptions, une mise à jour Bluetooth et une activation du mode broadcast est demandé.

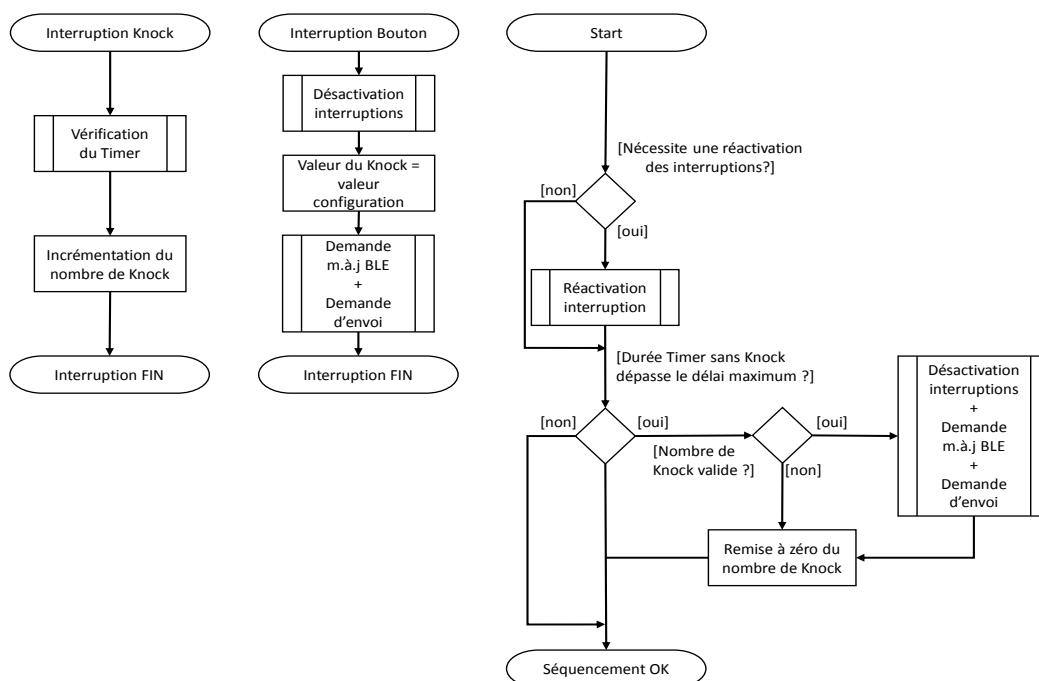


Figure 19: Séquençement du processus Accéléromètre



4.1.4.2 Transmission de l'information (Bluetooth Low Energy)

Le séquençement Bluetooth s'effectue après le séquençement de l'accéléromètre. Celui-ci se déroule de la façon suivante :

- En premier lieu, la fonction HCI_Process() s'effectue. Celle-ci a pour rôle de traiter si présents tous les évènements qui se sont passés entre deux appels de cette fonction. Toutes les évènements concernant les modifications (venant d'autres appareils ou de lui-même) sur les données Bluetooth s'effectueront par le biais de cette fonction.
- Avant d'effectuer une quelconque action, une vérification de procédure est faite afin de confirmer qu'aucune n'est en cours de déroulement. Dans le cas contraire, le séquençement se terminera.
- S'il n'y a pas de procédure en cours, une vérification sur la demande d'activation du mode broadcast est effectuée. Si celle-ci est positive, le bracelet enverra son nombre de Knock pendant une durée de X secondes (en l'occurrence 1.5 secondes).
- Le broadcast est un mode de fonctionnement d'un appareil Bluetooth. Celui-ci permet lors de son activation que tous les appareils à proximité puissent recevoir un message préconfiguré par l'utilisateur sans que celui-ci ait besoin de se connecter à un appareil. En effet si notre bracelet se connecte à un émetteur/récepteur, les informations qu'il contient ne sont accessibles que par le module. Grâce à ce broadcast, nous enverrons le nombre de Knock à tous les émetteurs/récepteurs simultanément.
- Enfin, une mise à jour des données est faite si un changement a été effectué (réception d'un nouveau nombre de Knock).

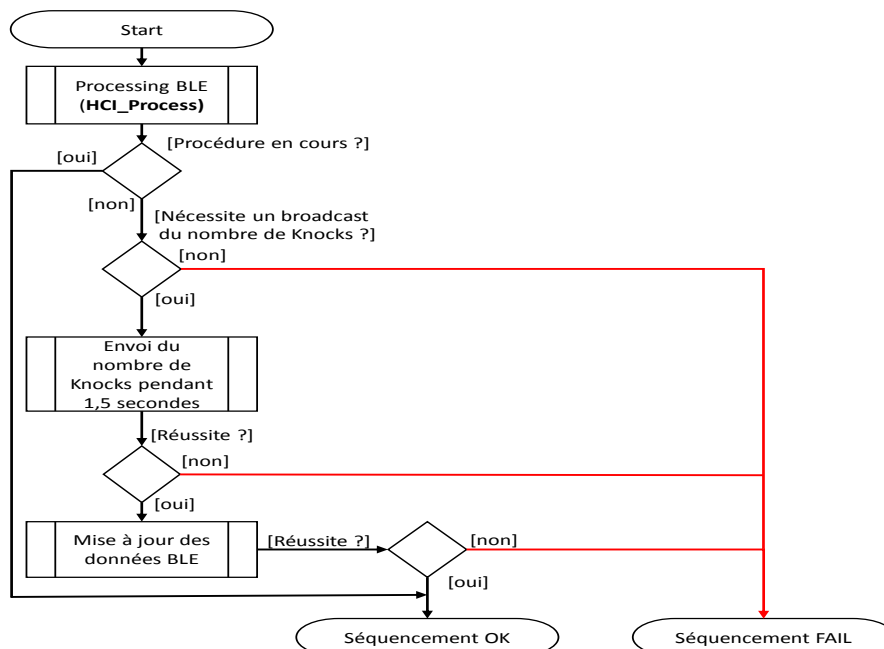


Figure 20: Séquençement du processus Bluetooth Low Energy

4.1.5 Résultats

Une fois tous les algorithmes en place, le bracelet nous retourna les informations suivantes :

```
main: Init et Config OK
ACX_process: Nombre de taps: 1
ACX_process: Taps ignore
ACX_intefonc: Tap detecte
ACX_intefonc: Tap detecte
ACX_process: Nombre de taps: 2
ACX_process: Taps OK
BLP_setbroad: Broadcast Mode ON
BLP_setbroad: Broadcast Mode OFF
BLP_setbroad: Reactivation de l'accelerometre
ACX_intefonc: Tap detecte
ACX_intefonc: Tap detecte
ACX_intefonc: Tap detecte
ACX_intefonc: Tap detecte
ACX_intefonc: Tap detecte
ACX_intefonc: Tap detecte
ACX_process: Nombre de taps: 6
ACX_process: Taps OK
BLP_setbroad: Broadcast Mode ON
BLP_setbroad: Broadcast Mode OFF
BLP_setbroad: Reactivation de l'accelerometre
ACX_intefonc: Tap detecte
ACX_intefonc: Tap detecte
ACX_intefonc: Tap detecte
ACX_intefonc: Tap detecte
ACX_intefonc: Tap detecte
ACX_intefonc: Tap detecte
ACX_intefonc: Tap detecte
ACX_process: Nombre de taps: 7
ACX_process: Taps ignore
```

Figure 21: Logs du bracelet

Concernant l'aspect du bracelet, celui-ci est décomposé en 2 versions :

- Une première version opérationnelle. Celle-ci est composée du microcontrôleur, du shield Bluetooth ainsi que de l'accéléromètre (du bas vers le haut).

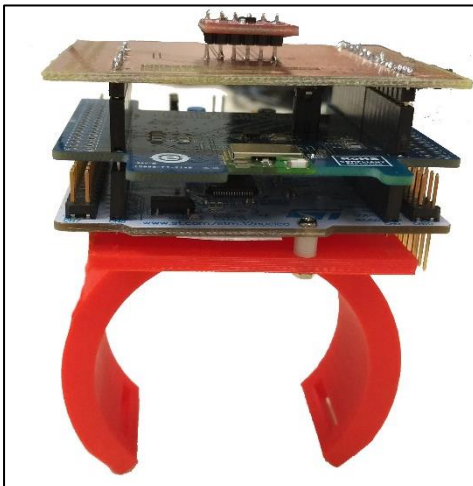


Figure 22: 1ère version du bracelet

- La deuxième version quant à elle est une miniaturisation de la première version. Toutes les fonctionnalités non nécessaires du 1^{er} prototype ont été enlevé afin d'épurer le système et ainsi gagner en place. Les schémas électriques, le routage ainsi qu'un visuel de la 2^{ème} version suivent.

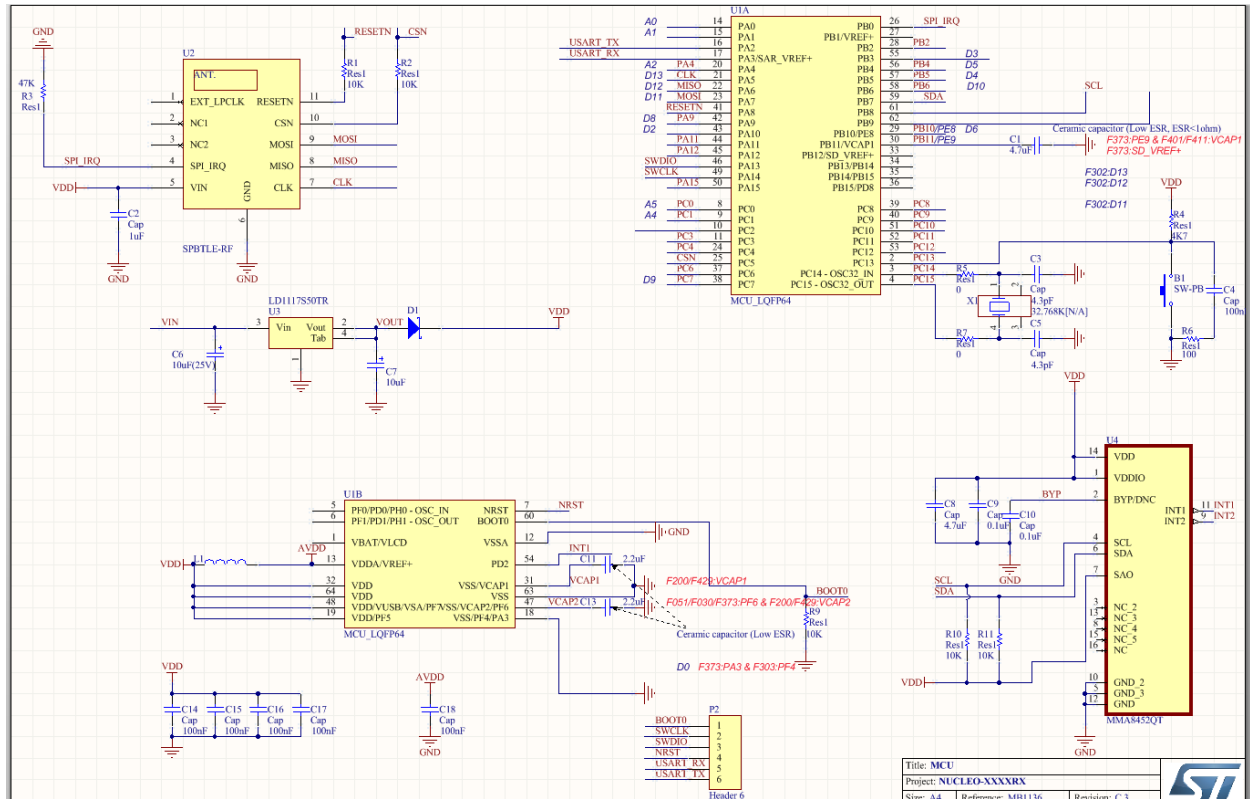


Figure 23: Schémas électriques du bracelet

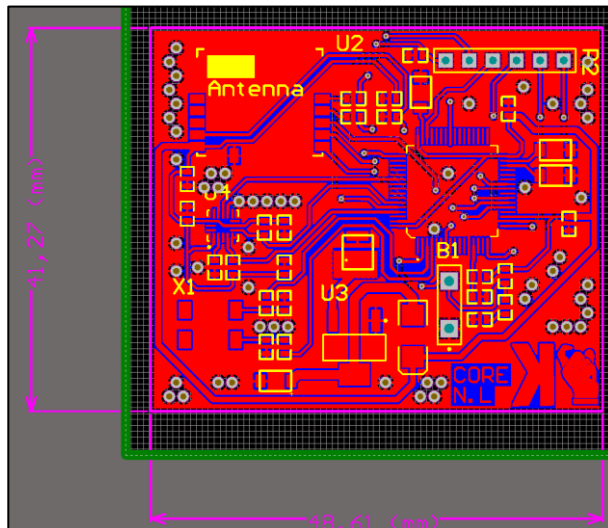


Figure 25: Routage du bracelet



Figure 24: 2ème version du bracelet

4.2 Modules émetteurs/récepteurs

Nos modules émetteurs/récepteurs se composent d'une Raspberry Pi 3 connecté en Wi-Fi au réseau de la maison. Son fonctionnement réside dans un programme (ou daemon) qui lit en permanence les périphériques BLE à proximité. Une fois détecté, le processus lit les informations et les transmet à la centrale domotique.

C'est par l'intermédiaire du RSSI, une valeur correspondant à l'atténuation du signal reçu, qu'est localisé le bracelet. Le module recevant la plus faible atténuation est considéré comme le plus proche. En fonction de cette valeur, un délai est appliqué sur la transmission en HTTP et seule la première requête est traitée par la centrale domotique. Le programme d'écoute utilise une bibliothèque BLE écrite en Go et développé par PayPal. Cette bibliothèque est disponible sur [Github](#). Ce programme se comporte comme un service Linux et est exécuté dès le lancement de la Raspberry.

4.2.1 Communication avec JEEDOM

Le protocole HTTP a été choisi pour communiquer avec la centrale domotique. Il est simple d'utilisation et ne requiert qu'une connexion Wi-Fi qui relie les différents modules Knocklet du domicile. Les informations sur le Knock sont transmises dans une requête POST avec les informations sous forme de JSON RPC. Pour chaque message, le module envoie son identifiant (son adresse Bluetooth) à la centrale domotique. Exemple:

```
{
  "method": "knock",
  "jsonrpc": "2.0",
  "params": {
    "apikey": "maClefDAPI",
    "braceletId": "4B:4E:4F:00:00:01",
    "moduleId": "8B:CE:4F:00:00:01",
    "rssi": -42,
    "knocks": 2
  }
}
```

Dans le cas de l'ajout d'un module ou d'un bracelet, il est possible d'envoyer la méthode "init" qui permet d'ajouter un appareil dans le système Knocklet. Le bracelet possède un bouton permettant d'envoyer une commande d'initialisation.

4.2.2 Résultat

Voici un exemple de registre que reçoivent les modules :

```
Peripheral ID: 4B:4E:4F:00:00:01, NAME: (KNOCKLET1)
Local Name      = KNOCKLET1
TX Power Level  = 0
Manufacturer Data = []
Service Data    = [{626f69726564656c616c6366f6c0000 [2]}]
RSSI            = -50

28
-65
-50
-67
-50
{"method": "knock", "jsonrpc": "2.0", "params": {"apikey": "lbippE2VJkuxVdfI4sKtZWQkCTwNJxH517kYVYEDL8bUe2eD", "braceletId": "4B:4E:4F:00:00:01", "moduleId": "8B:CE:4F:00:00:01", "rssi": -50, "knocks": 2}}
0:27:EB:19:84:89, "rssi": -50, "knocks": 2}}
(2.0 0 OK ! 0 commande(s) et 1 scénario(s) executes {0 })
```

Figure 26: Deux Knocks reçus par la Raspberry Pi



4.3 Centrale domotique

Nos objectifs lors du développement du plugin Knocklet sur Jeedom étaient de recevoir les triplets (bracelet ID, module ID, nombre de Knocks) envoyés par les modules enregistrés automatiquement, de déclencher les commandes et scénarios correspondants et de fournir une interface utilisateur esthétique et ergonomique permettant une personnalisation complète des Knocks.

4.3.1 Interface utilisateur

L'interface de notre plugin sur Jeedom a pour rôle de permettre aux utilisateurs de choisir le nombre de coups à donner pour déclencher les différentes commandes et scénarios. Le plugin doit également fournir une initialisation des équipements Knocklet de manière automatique et une configuration facile et intuitive de ces derniers.

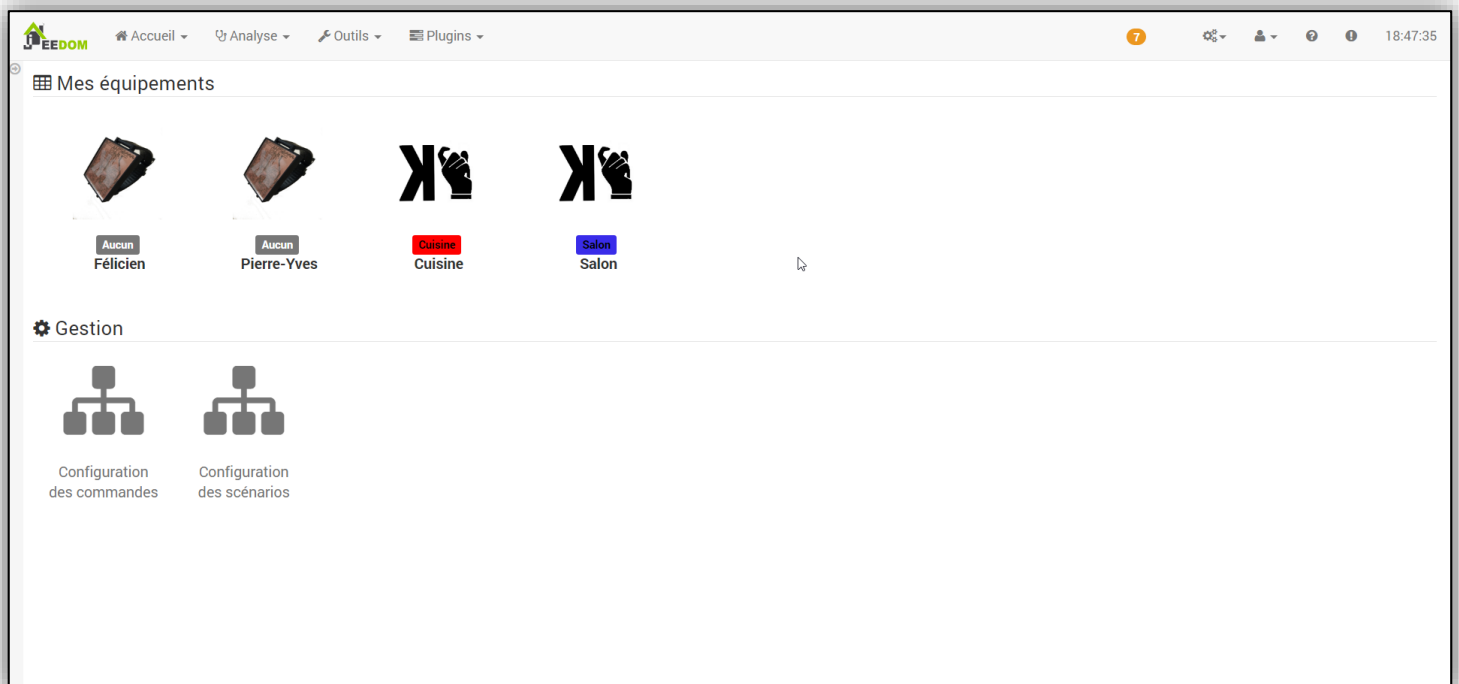
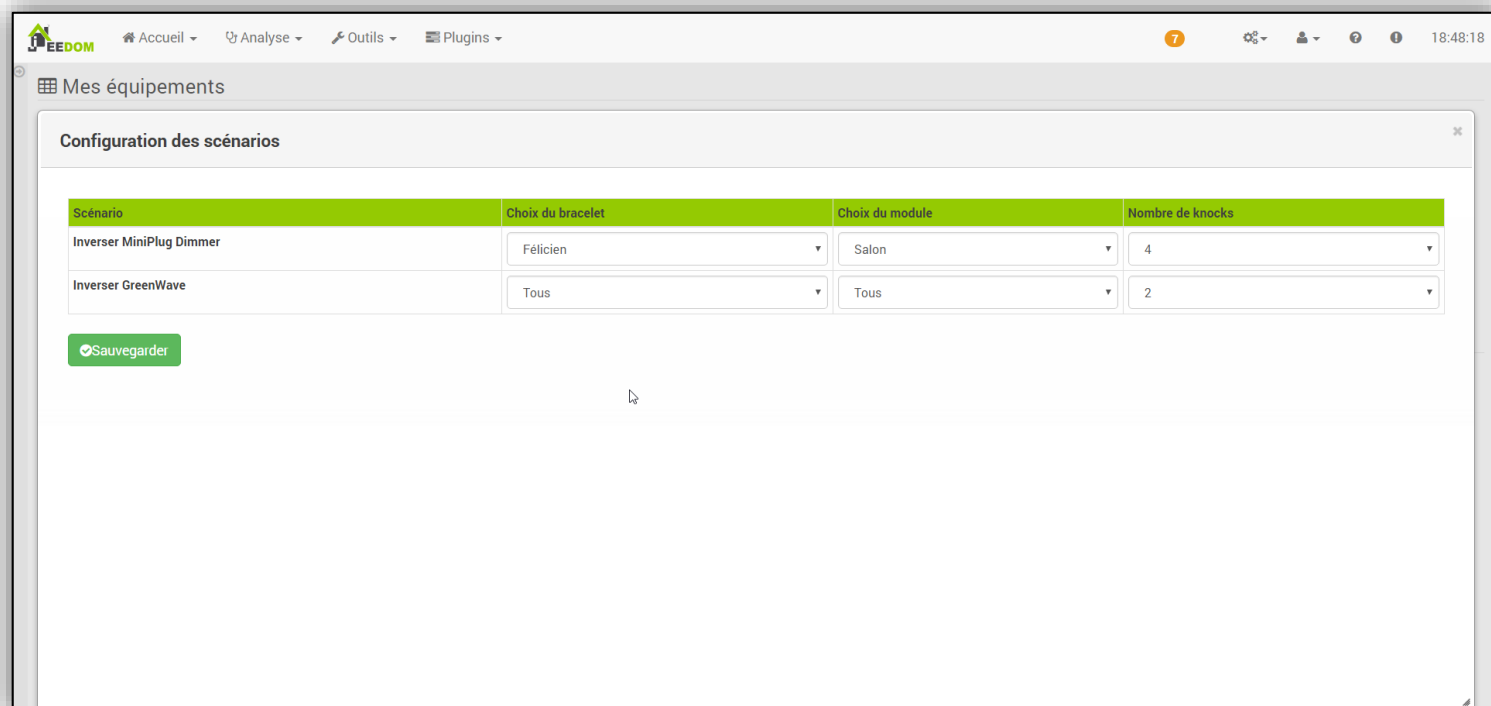


Figure 27: Page du plugin Knocklet sur Jeedom

Nous avons choisi un style graphique similaire à ceux des autres plugins Jeedom. Dans une première section libellée « Mes équipements », une liste des appareils Knocklet est affichée. Cette liste est récupérée dans la base de données à chaque ouverture du plugin. Cliquer sur un des équipements vous redirigera vers sa page de configuration. Vous pourrez ensuite modifier son nom, régler sa visibilité et voir son adresse MAC.



Dans une deuxième section nommée « Gestion », deux boutons de configuration sont présents : « Configuration des commandes » et « Configuration des scénarios ». Cliquer sur l'un d'eux ouvrira une fenêtre en overlay qui vous servira à associer des triplets « Bracelet », « Module » et « Nombre de Knocks » à une commande ou à un scénario. Un tableau qui répertorie respectivement tous les scénarios et toutes les commandes avec leur triplet associé est créé automatiquement lors de l'appel de cette fenêtre. Vous pouvez ensuite changer la configuration de vos Knocks selon vos préférences et appuyer sur le bouton « Sauvegarder » pour l'enregistrer dans la base de données.



The screenshot shows the 'Configuration des scénarios' window in the EEDOM application. The window has a title bar with the EEDOM logo and navigation links: Accueil, Analyse, Outils, and Plugins. The main content area is titled 'Mes équipements' and contains a sub-section 'Configuration des scénarios'. Below this, there is a table with the following data:

Scénario	Choix du bracelet	Choix du module	Nombre de knocks
Inverser MiniPlug Dimmer	Félicien	Salon	4
Inverser GreenWave	Tous	Tous	2

At the bottom left of the window, there is a green button labeled 'Sauvegarder'.

Figure 28: Fenêtre de configuration des scénarios

4.3.2 Fonctionnement du plugin

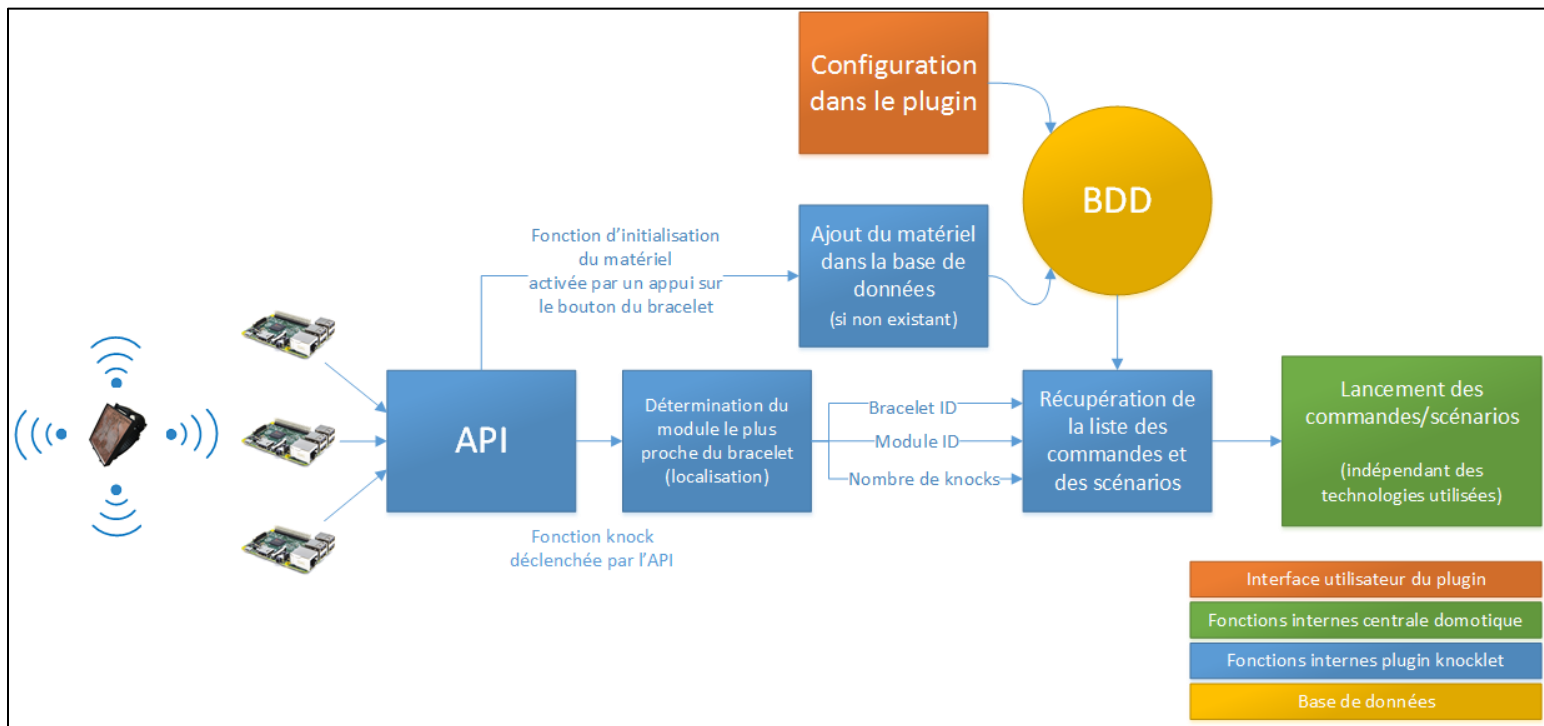


Figure 29 : Fonctionnement du plugin

Dans cette section, nous verrons comment les informations envoyées par les bracelets / modules sont traitées par le plugin. Le schéma suivant n'explique donc pas en détail ce qu'il se passe avant la réception des informations par l'API (voir parties précédentes).

Ce schéma décrit donc la fonction interne du plugin, il existe deux cas différents de fonctionnement :

- Initialisation du matériel :

Lorsque l'utilisateur appui sur le bouton d'un bracelet, tous les modules à portée vont transmettre une demande d'initialisation. L'API (page en php attendant des requêtes de type POST) reçoit donc un fichier json contenant les informations nécessaires pour déclencher la méthode « init » qui va enregistrer dans la base de données les nouveaux équipements qui ont transmis cette demande.

Ainsi, l'utilisateur n'a qu'à brancher tous ses modules, puis à appuyer sur le bouton de son (ses) bracelet(s) pour les voir apparaître dans la page de configuration du plugin.

- Réception de Knocks :

Lorsque des Knocks sont détectés par les bracelets et transmis à la centrale par l'intermédiaire des modules, le plugin reçoit sur son API une requête appelant la méthode « Knock ». Cette méthode utilise les IDs des équipements (bracelet + module le plus proche du bracelet) ainsi que le nombre de Knocks pour déclencher les commandes et les scénarios correspondants dans la base de données (au préalable remplie par l'utilisateur via l'interface du plugin).

Pour rappel, la localisation se fait grâce à l'ordre dans lequel les modules envoient l'information (voir partie sur les modules). Ainsi, seule la première demande venant d'un bracelet donnée est exécutée. Celles provenant d'autres modules (plus lointains) pour le même bracelet sont ignorées pendant un court laps de temps (1,5 seconde).



5 Difficultés rencontrés

5.1 Accéléromètre

Afin de mieux observer le comportement de notre accéléromètre et de mieux calibrer la détection des Knocks, nous avons dû réaliser un montage parallèle, comprenant une carte Arduino UNO, nous permettant de visualiser en temps réel nos valeurs d'accélération sur les trois axes.

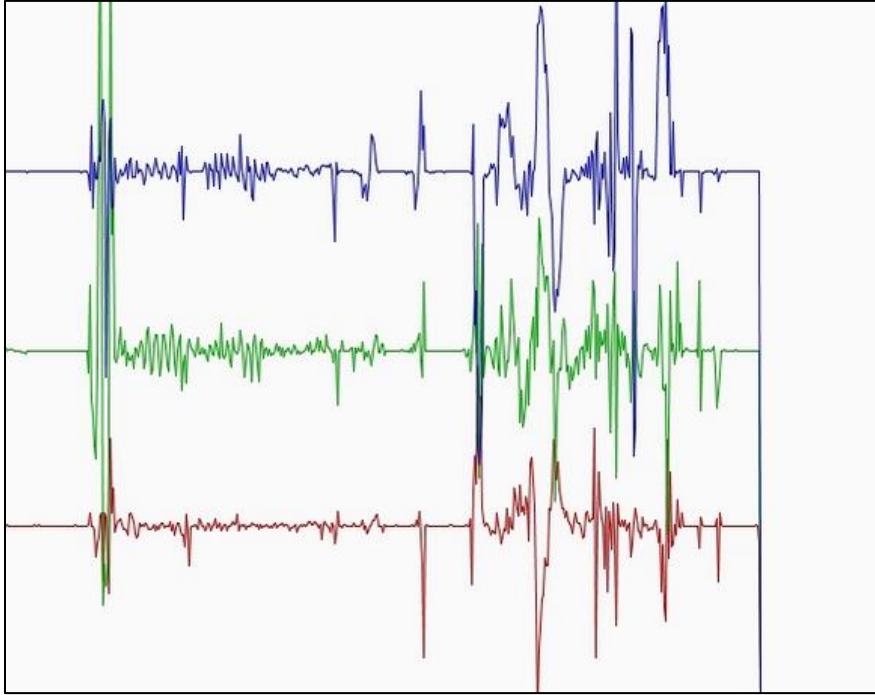


Figure 30: Courbe d'accélération des axes X, Y et Z

5.2 Transmission sans fil du bracelet

Lors de nos premiers tests sur la transmission complète du signal (nombre de Knock du bracelet -> E/R puis E/R -> JEEDOM), nous nous sommes aperçus que le temps de latence entre l'émission de la commande et son déclenchement était très élevé. Par la suite, nous avons remarqué que ce délai était provoqué lors du passage de l'information du nombre de Knock à l'émetteur/récepteur. Ainsi, nous avons changé le fonctionnement du bracelet et des E/R afin que ce transfert soit le plus optimisé possible. Voici notre premier schéma concernant le chemin du signal (la version finale étant montrée [ici](#)).

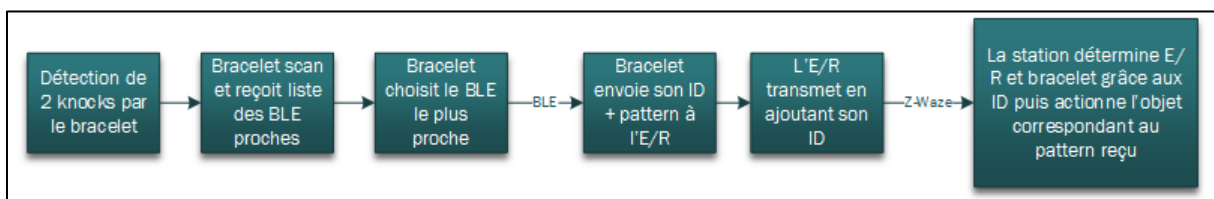


Figure 31: Première version du chemin du signal lors de l'envoi d'une commande

5.3 Miniaturisation du système

La miniaturisation (concernant principalement le bracelet) a été une vraie problématique lors du projet. En effet l'objectif initial sur le bracelet et que celui-ci respecte le cahier des charges concernant ses dimensions et son poids. Cette contrainte en tête, nous avons délibérément choisi des composants de type CMS (composant monté en surface). Ces derniers ont l'avantage d'être souvent plus petits et plus légers. De plus, les circuits imprimés n'auront pas à être percé ce qui facilite le processus d'industrialisation. Enfin, ils sont en général moins chers que des composants « basiques ».

Cependant, un des avantages de ces composants s'est retourné contre nous : la taille. En effet les accéléromètres que nous avons choisi (le plus petit étant de taille 2x2 mm et le plus grand de taille 3x3 mm) n'ont pas pu être exploités de par leur trop petite taille. Nous avons effectué de nombreux essais de carte au Fablab cependant, aucun ne s'est avéré positif... Voici ci-contre un essai à blanc concernant un circuit imprimé comprenant un accéléromètre de taille 2x2 mm Vous pouvez voir que certaines pistes de celui-ci ont fusionné malgré la plus grande prudence exercée.

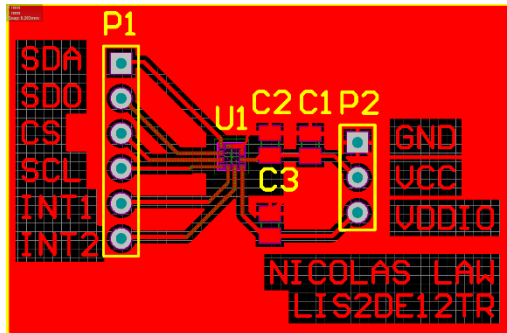
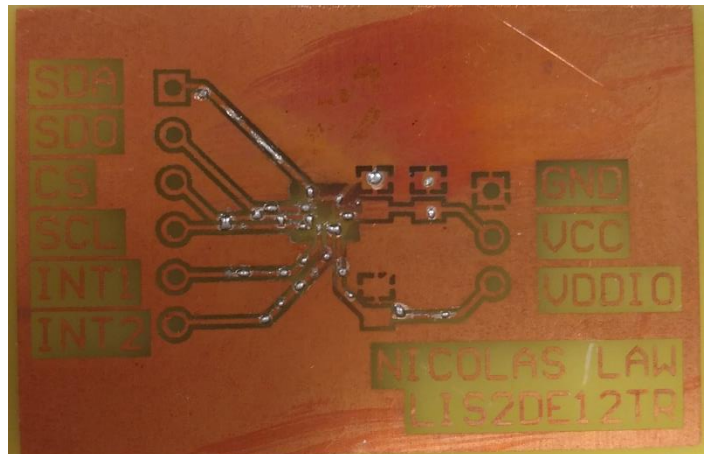


Figure 32: Essai de brasage pour un accéléromètre



Une fois le premier prototype fonctionnelle (bracelet rouge), nous avons tout de même décidé de développer une version miniature du bracelet en sachant que nous ne pourrions pas complètement le terminer afin de montrer qu'il était possible de respecter le cahier des charges.