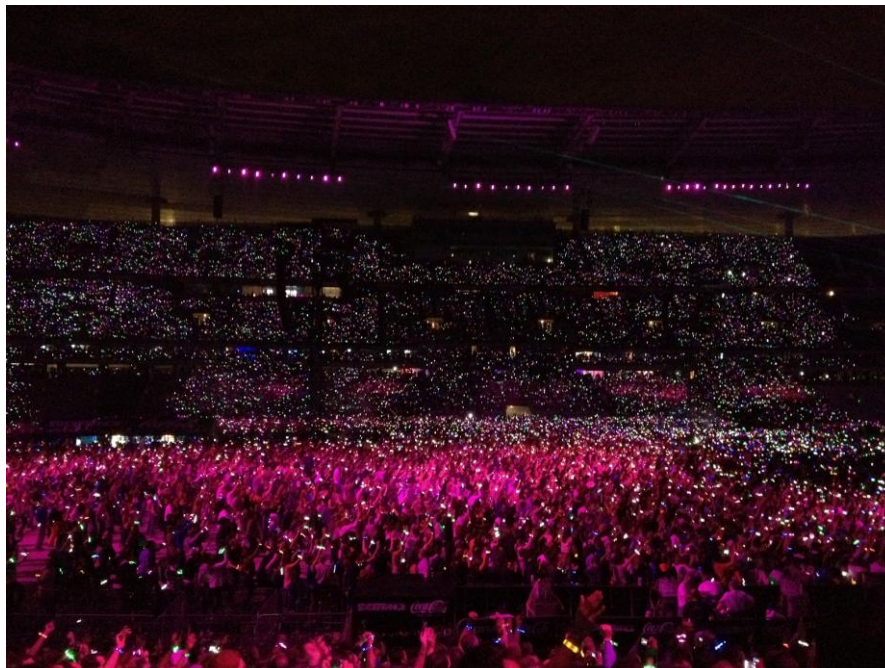


*Projet de Fin d'Etudes :*

## **Transformation des spectateurs d'un concert en afficheur géant interactif**

Ecole Polytechnique universitaire de Lille

Département Informatique, Microélectronique et Automatique



Auteur : Mélanie Hauteceur

Encadrants : Alexandre Boé / Thomas Vantroys

Année : 2014/2015

## Sommaire

Introduction.....	3
1. Présentation du projet .....	4
1.1 Objectifs.....	4
1.2 Cahier des charges.....	4
2. Localisation absolue .....	5
2.1 Principe.....	5
2.2 Protocole de communication .....	6
2.3 Tests réalisés .....	7
3. Localisation relative.....	8
3.1 Principe.....	8
3.2 Protocole de communication .....	9
5. Travail à poursuivre .....	10
Conclusion .....	10

## Introduction

Ce rapport relate le projet réalisé pendant les soixante derniers jours, qui constitue mon projet de fin d'études. Celui-ci intervient pendant la dernière année du cycle d'école d'ingénieur dans le département informatique, microélectronique et automatique de l'école Polytech Lille, et est en collaboration avec le laboratoire de l'IRCICA.

Nous verrons dans ce rapport l'étude et le développement d'un bracelet interactif destiné aux publics de concerts, afin que ces derniers puissent participer à l'évènement en devenant les pixels d'une image géante. Mais ce projet consistera avant tout à réussir à déterminer quelle personne correspond à chaque pixel.

Nous commencerons par étudier les objectifs de ce projet ainsi que le cahier des charges établi avec les encadrants du projet. Nous nous intéresserons ensuite aux différentes manipulations et aux développements réalisés grâce à un prototype de carte créé pour ce bracelet. Nous aborderons alors les améliorations envisageables, puis nous terminerons par conclure sur les avancées effectuées et étudiées, et les apports de ce sujet.

# 1. Présentation du projet

## 1.1 Objectifs

Ce projet fait suite à un projet réalisé à Polytech Lille l'année précédente, 'Capteur Communicant Intelligent' par Bastien Chalaux et Nicolas Husse, et utilise la carte créée par les encadrants pour le finaliser. Il consiste à créer une interaction entre un artiste et son public via un bracelet affichant une couleur. Ainsi, le spectateur fait lui même parti du spectacle. Nous aurons comme objectif d'afficher une image géante en faisant de chaque bracelet, donc chaque personne, un pixel de cette image.

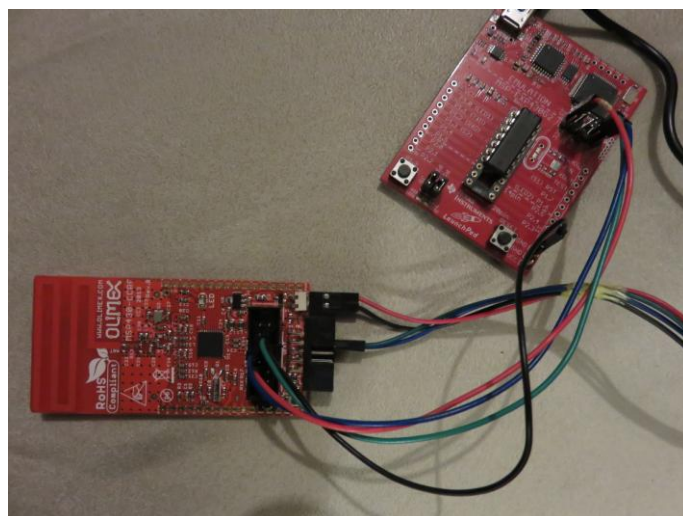
Pour cela, nous aurons plusieurs défi : le premier consiste à trouver la position de chaque bracelet afin de déterminer quelle couleur il doit afficher, ce qui constitue le second défi. Ce deuxième défi possède plusieurs solutions, mais étant limitée par le temps, je me suis consacrée au premier défi, nous étudierons donc la localisation de la carte en 'indoor' par radiofréquence, et plus précisément à l'aide du RSSI (Received Signal Strength Indication) du signal qui correspond à la puissance reçu du signal émis.

Afin de localiser la carte le plus précisément possible et en limitant les erreurs, nous envisagerons d'effectuer deux localisations successives : une localisation absolue grâce à trois antennes fixées autour de la zone du public, puis une localisation relative des bracelets les uns par rapport aux autres.

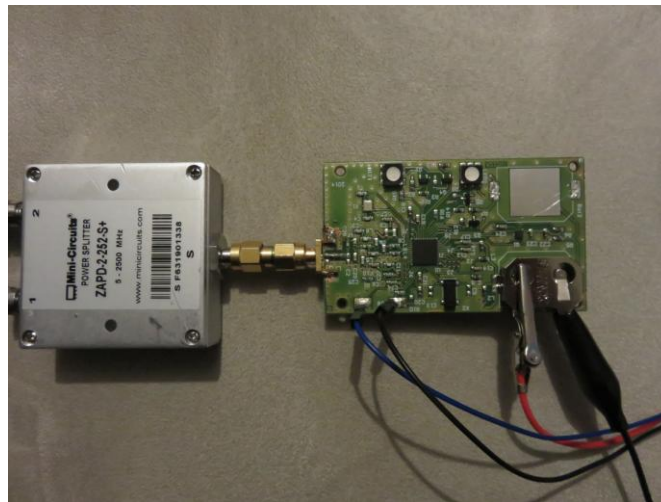
## 1.2 Cahier des charges

Tout d'abord, pour atteindre ces objectifs, je dispose du matériel suivant :

- une carte MSP430-CCRF Olimex, équipée d'un microcontrôleur CC430F5137, d'une antenne et d'une communication UART permettant de récupérer les données en USB.



- plusieurs cartes de l'IRCICA équipées d'un microcontrôleur CC430F5137 et d'une antenne.



- un splitte, permettant de connecter une carte sur deux antennes simultanément.
- deux launchpads MSP430
- les codes de transmission et réception, à étudier mais donc pas à refaire.

Après avoir rencontrés les encadrants, j'ai pu établir un cahier des charges des attentes du projet : on veut donc une localisation précise à 80cm près au maximum, qui correspondent à l'espace entre les sièges. Il faut également que cette localisation soit assez rapide, afin de ne pas devoir attendre que cela se termine avant que le spectacle ne commence.

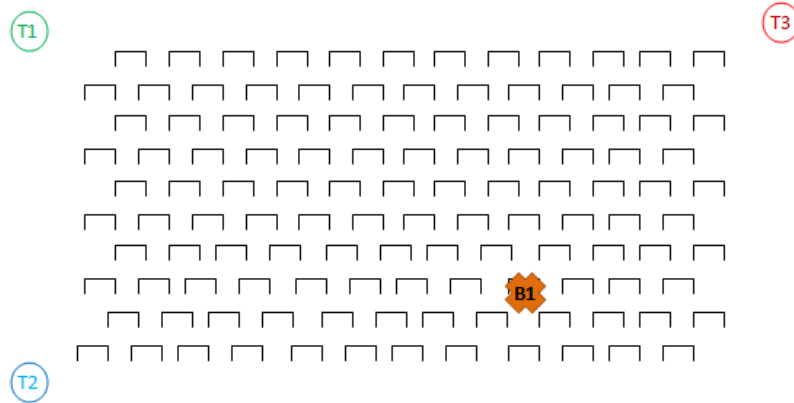
On considèrera néanmoins que les spectateurs sont statiques, par conséquent une seule localisation est nécessaire à l'arrivée du public.

## 2. Localisation absolue

### 2.1 Principe

La localisation absolue signifie que l'on trouve la position du bracelet par rapport au repère de la salle toute entière. Nous utiliserons pour cela la puissance du signal reçu, alias le RSSI du signal. Pour cela, nous transmettrons à plusieurs fréquences une trame à partir de trois tourelles disposées à divers endroits de la salle. Emettre à plusieurs fréquences nous permet d'avoir plus de relevés et donc d'éliminer au mieux les erreurs et d'augmenter la précision.

Nous relèverons ensuite son RSSI, puis nous utiliserons un algorithme développé par des chercheurs de l'IRCICA permettant la mise en adéquation RSSI-distance, puis la localisation du récepteur.



## 2.2 Protocole de communication

Les tourelles émettent sur différentes fréquences successives, que l'on appellera 'chaînes', simultanément grâce à un splitter. L'émission est programmée telle que la fréquence est définie de la façon suivante :

$$f = \text{base frequency} + \text{CHANNR} * \text{channel\_spacing}$$

On a fixé *base frequency* et *channel spacing* dans le code, puis on modifie *CHANNR* afin d'augmenter la fréquence à chaque émission. La trame transmise est de la forme :



Exemple de trame émise par les tourelles

- La première partie est un préambule
- Le **troisième octet** indique l'ID de l'émetteur
- Le **quatrième** correspond à la commande, ici 'f' pour fréquence, car on change de fréquence après chaque trame
- Le **cinquième** est la chaîne sur laquelle la prochaine trame sera émise, afin que le récepteur écoute à cette fréquence
- Le **sixième** indique le 'link quality'
- La **dernière** partie donne le RSSI du signal

Une fois l'émission réalisée, on peut alors récupérer les différents RSSI puis les analyser grâce à l'algorithme. On peut résumer ce protocole de la façon suivante :

- Emission des trames à différentes fréquences par les tourelles
- Récupération des RSSI en fonction des chaînes dans un tableau par les bracelets
- Traitement des résultats par les bracelets, obtention de la position
- Emission des résultats par les bracelet tour par tour (ID par ID)
- Récupération et stockage des résultats envoyés par les bracelets

## 2.3 Tests réalisés

Tout d'abord, j'ai réalisé plusieurs tests afin d'étudier le fonctionnement du matériel et la forme des trames obtenues. J'ai alors changé les paramètres et le programme afin d'obtenir une émission sur les différentes chaînes et une réception permettant une lecture sur le port USB d'un ordinateur.

Une fois en possession des codes et du matériel nécessaire, j'ai effectué différents tests afin de soumettre les résultats à l'algorithme et les vérifier. J'ai alors fait les manipulation suivantes pour estimer les résultats:

1. Emission avec une première antenne
2. Emission avec une seconde antenne
3. Emission avec les deux antennes réunis

Premièrement, ce test devait permettre de prouver que le récepteur pouvait recevoir des trames des deux antennes simultanément, or je suis parvenue à collecter toutes les données, ce qui confirmait cette théorie.

J'ai alors continuer ces tests pour différentes plages de fréquences et différentes configurations.

J'ai ensuite refait ces essais avec trois antennes, mais il semblait qu'il fallait une plage de fréquence plus grande, j'ai alors presque doublé la bande. Cependant la réception était très mauvaise, et le nombre de données transmises ne correspondait pas à celle que je m'attendais à avoir, elles étaient bien plus nombreuses.

Je suis parvenue à résoudre les problèmes de réception en changeant l'une des antennes qui était défaillante, puis j'ai résolu le second problème. L'émission s'effectue de cette manière :

```
for(i = CHANNEL_START; i <= CHANNEL_STOP; i += CHANNEL_STEP){           //defined in RFLA.h

    ChangeChannel(i);                //Set channel

    ReceiveOff();

    __delay_cycles(1000000);        //Let the receiver to change frequency

    send_cmd('f', i+CHANNEL_STEP); //Advertise receiver to change channel

    __delay_cycles(DELAY);

}
```

'i' étant codé sur un octet, lorsqu'il arrivait à 255, il retournait à 0, donc si CHANNEL\_STEP était plus grand que 1, et que CHANNEL\_STOP était de 255, alors la boucle se faisait plusieurs fois.

Cependant, ces tests ne sont pas toujours efficaces : en effet, ils sont très long à réaliser car les interférences et multi-trajets empêchent d'obtenir suffisamment de trames dans beaucoup de cas. Il faut donc les répéter plusieurs fois.

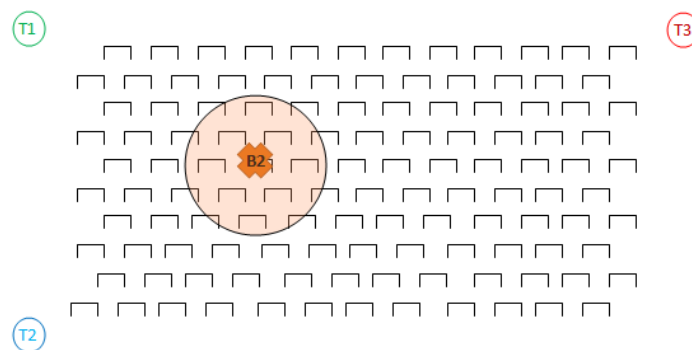
Une fois la localisation absolue réalisée et renvoyée aux tourelles, on peut effectuer une localisation relative afin de confirmer/rectifier la position du bracelet.

### 3. Localisation relative

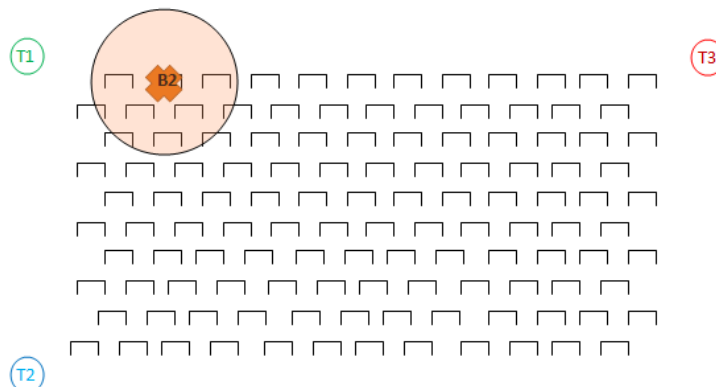
#### 3.1 Principe

La localisation relative consiste à trouver la position des bracelets les uns envers les autres. Cela signifie que l'on va faire émettre les bracelets chacun leur tour. Premièrement j'avais envisager de le faire suite au renvoi de la position, mais j'ai finalement décidé de faire ces deux tâches en même temps, ce qui permettra d'optimiser les envois et surtout d'économiser du temps.

Chaque bracelet émettra donc tour à tour, pendant que l'ensemble des autres bracelets écoutent. On déterminera alors quels sont les différents bracelets situés dans la zone d'un autre bracelet, en croisant les résultats, on pourra déterminer la position relative des bracelets.



*Exemple 1 : le bracelet en position provisoire (4,5) avec l'ID 1 émet sa position, les bracelets dans la zone autour reçoivent son signal*



*Exemple 2 : c'est ensuite le bracelet avec l'ID 2 en position provisoire (1,2) qui émet*

Une fois que chaque bracelet a émis, la tourelle a récupéré chacune des positions provisoires, et les bracelets émettent à nouveau les différents ID qu'ils ont reçu. Pour



émettre tout cela, il faut de nouveaux formats de trames. Observons le protocole adopté pour cette partie.

### 3.2 Protocole de communication

J'ai donc adopté de nouvelles trames contenant de nouvelles commandes. La trame émettant la position provisoire est la suivante :



*Exemple de trame émise tour à tour par les bracelets contenant la position provisoire*

- La première partie contient le préambule
- Le **troisième** octet est l'identifiant du bracelet
- Le **quatrième** est la commande 'p' pour position
- Le **cinquième** est la position provisoire en x
- Le **sixième** est la position provisoire en y

On a donc l'identifiant permettant de savoir quel bracelet émet, ce qui sera garder en mémoire pour déterminer la position finale.

Premièrement, j'ai penser émettre dans l'ordre des positions provisoires. J'ai ensuite réalisé qu'il serait possible que deux bracelets pensent être au même endroit et donc qu'ils émettent en même temps. Il serait donc préférable qu'ils émettent selon leur ordre d'identifiant, ceux-ci étant uniques. On désigne donc une durée *delay* et chaque bracelet émettra à  $t = ID * delay$ .

Une fois que chaque bracelet a émis, c'est à dire à  $delay * nb\ de\ bracelets$ , on émet la dernière trame de la même façon que celle-ci. Cette fois, la trame sera :



*Exemple de trame émise tour à tour par les bracelets contenant les ID détectés*

- La première partie contient le préambule
- Le **troisième** octet est l'identifiant du bracelet
- Le **quatrième** est la commande 'i' pour ID
- Les **cinquième** , **sixième** et **septième** sont les ID reçus par le bracelet 8

Une fois toutes ces données transmises, les tourelles peuvent déterminer la position grâce au premier algorithme et au croisement des données de positionnement relatif. Cependant

je n'ai pas pu aller au bout de ce développement en pratique, il reste donc un certain travail à réaliser.

## 5. Travail à poursuivre

Malgré l'étude du sujet bien avancé, je n'ai malheureusement pas eu le temps de terminer son application, je vais donc maintenant aborder les différentes opérations que je n'ai pas pu réaliser.

Premièrement, les résultats des tests ne me sont pas parvenus, et il est primordial pour ce projet que ces derniers soient concluants. Il faut donc vérifier la véracité de ceux-ci.

Une fois l'algorithme confirmé, il faut l'intégrer dans le code source afin que les bracelets puissent déterminer leur position provisoire. Il faut également développer l'algorithme rectifiant la position grâce à la localisation relative.

Enfin, je n'ai pu récupérer les trames du côté des tourelles, suite à plusieurs problèmes que je n'ai pas pu expliquer, comme la réception des données de la tourelle elle-même, alors que les interruptions de réception de données sont désactivées. Néanmoins le reste du programme ainsi qu'une grande partie du raisonnement sont achevés.

## Conclusion

Même si je n'ai pu aller au bout de ce projet, je pense qu'il est possible d'obtenir, grâce à toutes les données relevées, des résultats intéressants. Cependant, le nombre d'essais qu'il m'a fallu pour réaliser des relevés exploitables me laisse penser qu'il serait difficile de réaliser ces relevés sur un grand nombre de personnes dans un endroit contenant beaucoup d'interférences.

Il m'aurait été agréable d'avoir plus de temps pour déterminer si cette solution est possible et avoir des résultats plus concrets.

Ce projet m'a beaucoup apporté au niveau personnel et m'a permis de développer de nouvelles compétences. En effet, gérer un projet seule m'a appris à être plus minutieuse et à régler chaque problème rencontré de façon autonome.

De plus, la localisation 'indoor' est un sujet très présent ces derniers temps, et j'ai apprécié de pouvoir en apprendre plus sur ce domaine, ainsi que sur l'utilisation des radiofréquences. Ce projet correspondait réellement au département informatique, microélectronique et automatique, mais surtout il répondait aux attentes que j'avais concernant ce stage de dernière année.