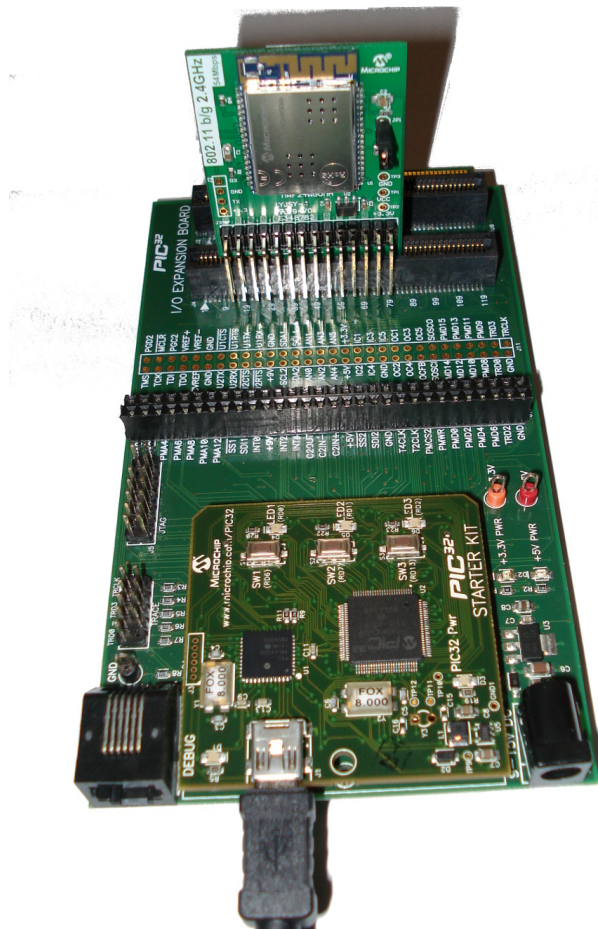


PAR 153 - DOCUMENT DE SYNTHÈSE

7 avril 2014

PAR 153 : Etude et développement d'un nœud de capteurs  
sans fils multistandard (IEEE 802.15.4 IEEE 802.11)



# Résumé

Le but du projet est de faire communiquer grâce à une carte électronique un réseau WPAN avec un réseau WLAN dans le cadre des réseaux de capteurs sans-fils (WSN). Les capteur sans fils communiquent sur des réseaux dits WPAN tandis que nos ordinateurs classiques communiquent en WLAN C'est donc une passerelle WPAN/WLAN qui doit être mise en œuvre.

Le projet consiste donc en une partie de documentation, une partie de conception du système, et une partie de développement (en langage C) et de tests.

La manière dont la passerelle doit fonctionner et interagir avec les deux réseaux a été pensée et expliquée. Des composants compatibles ont été choisis et le programme commencé. Cependant suite à des problèmes de développement et l'ambition peut-être trop élevée du projet, le programme permettant à la passerelle de fonctionner n'est pas abouti.

**Mots-clés : Microcontrôleur, Électronique, Développement C, Microchip, IEEE 802.11, WLAN, IEEE 802.15.4, WPAN, Bridge, WSN**

# Table des matières

<b>Table des matières</b>	<b>ii</b>
<b>Table des figures</b>	<b>iii</b>
<b>0 Introduction</b>	<b>1</b>
<b>I Contexte et présentation générale</b>	<b>2</b>
<b>1 Contexte</b>	<b>2</b>
1.1 Usage général et présentation globale . . . . .	2
<b>2 Inscription de la thématique dans le Laboratoire INL, auquel je suis rattaché</b>	<b>5</b>
2.1 Organisation . . . . .	5
2.2 Domaines de recherches . . . . .	5
<b>II État de l'art</b>	<b>6</b>
<b>1 Solution commerciale</b>	<b>6</b>
1.1 GainSpan GS2000 . . . . .	6
<b>2 Solutions académiques</b>	<b>7</b>
2.1 Études généralement réalisées . . . . .	7
2.2 Un intérêt pour la compatibilité physique de ces deux types de normes . . . . .	8
2.3 Développement d'un Framework du fonctionnement du réseau . . . . .	8
<b>3 Bilan</b>	<b>9</b>
<b>III Le projet : démarche et opérations</b>	<b>10</b>
<b>1 Démarche générale</b>	<b>10</b>
<b>2 Normes de réseaux</b>	<b>10</b>
2.1 Organisation de la communication . . . . .	10
2.2 Architecture des paquets . . . . .	12
2.3 Choix de fonctionnement . . . . .	13
<b>3 Choix des composants</b>	<b>13</b>
3.1 Composants MICROCHIP . . . . .	13
3.2 Modules de communication . . . . .	13
3.3 Microcontrôleur maître . . . . .	13
<b>4 Les opérations : détails et méthode de programmation</b>	<b>14</b>
4.1 Module WLAN . . . . .	14
4.2 Module WPAN . . . . .	15
4.3 Conversion et traitement des données : le liant . . . . .	15
<b>5 État actuel du projet</b>	<b>15</b>

<b>IV Description de l'avancement et problèmes rencontrés</b>	<b>17</b>
1 Prise en main, documentation et réflexions sur le projet	17
2 Toussaint - Noël : recherche d'une erreur sur le module WLAN	17
3 Janvier - Mars : mise en place de la communication SPI sur le module WPAN	18
4 Bilan : état actuel du projet	18
<b>Conclusion</b>	<b>19</b>
<b>Bibliographie</b>	<b>20</b>

## Table des figures

I-1.1	Schéma simplifié d'un réseau WSN . . . . .	3
I-1.2	Forme d'une supertrame . . . . .	3
I-1.3	Réseau WSN avec nœud central multistandard . . . . .	4
I-1.4	Restriction du projet au nœud central. . . . .	4
II-1.1	Schéma bloc du composant . . . . .	7
II-2.1	Division des modules de l'interface ZigBee. . . . .	8
II-2.2	Chemin des données . . . . .	9
III-2.1	Mécanisme du CSMA/CA avec temporisation . . . . .	11
III-2.2	Mécanisme de communication avec CAP et CFP . . . . .	12
III-2.3	Structure d'un paquet suivant la norme IEEE 802.15.4. . . . .	12
III-2.4	Structure d'un paquet suivant la norme IEEE 802.11 . . . . .	12
III-3.1	Module MRF24WG0MA : Interface WLAN . . . . .	14
III-3.2	Pic 32 Starter Kit . . . . .	14
III-3.3	Carte de développement (travail sur un module). . . . .	14
III-4.1	Diagramme schématique du programme . . . . .	16
III-5.1	Bilan de l'avancement du projet . . . . .	16

## 0 Introduction

Faire communiquer des réseaux de types différents est souvent nécessaire. On transforme par exemple le réseau filaire en un réseau Wifi dans une borne Wifi. Cela nécessite de transformer le signal d'un format à un autre, voire d'un medium à un autre.

Le projet présenté ici s'ancre dans la recherche sur les réseaux de capteurs<sup>1</sup> sans fils. En effet, ceux-ci communiquent entre-eux en suivant une norme particulière, différente de celles utilisées par nos ordinateurs portables. Pour des raisons de simplicité de recueil des données des capteurs par exemple, on peut souhaiter faire communiquer ces deux types de systèmes ensemble malgré tout. Pour cela, on peut mettre en place une passerelle ou bridge entre les deux éléments communicants (ou entre les deux réseaux de communication).

C'est cette passerelle que ce projet se propose de concevoir.

Ce projet a été réalisé dans le cadre des Projets d'Application recherche (PAr) de l'École Centrale de Lyon, sous la tutelle de M. David Navarro, enseignant-chercheur du laboratoire INL<sup>2</sup>, et de Mme Sandrine Bec, enseignant-chercheur au LTDS<sup>3</sup>. Il s'est déroulé d'octobre 2013 à avril 2014.

La partie I présente le contexte du projet et ces objectifs, la partie II résume l'état de l'art et la partie III présente la démarche suivie ainsi que les solutions apportées. Enfin le détail de l'avancement et des problèmes rencontrés se trouve dans la partie IV.

---

1. Capteur : De manière générale, instrument permettant de convertir un signal donné (température, champ électrique...) en tension. Ici on étendra cette définition aux composants liés au capteur. On appellera donc capteur : le capteur, son instrumentation, le microcontrôleur qui lui est attaché ainsi que l'alimentation et le module de communication

2. Institut des Nanotechnologies de Lyon : <http://inl.cnrs.fr>

3. Laboratoire de Tribologie et Dynamique des Systèmes : <http://ltds.ec-lyon.fr/spip/>

---

## *Première PARTIE*

# CONTEXTE ET PRÉSENTATION GÉNÉRALE

---

## 1 Contexte

### 1.1 Usage général et présentation globale

#### 1.1.1 Réseaux de capteurs sans fil

Les réseaux de capteurs sans fil (WSN : Wireless Sensors Network) sont de plus en plus utilisés pour acheminer des informations<sup>1</sup> diverses de multiples endroits vers un centre de calcul ou de stockage. Que le but soit de mesurer des données<sup>2</sup> liées à l'environnement (hydrométrie, température, pollution...), de suivre ou surveiller des biens et personnes (système anti-cambriolage, veille médicale, etc.) ou de rendre intelligentes des infrastructures (indiquer la densité de circulation ou la présence d'un accident à des systèmes de signalisation par exemple), on retrouvera toujours la même architecture du système global. Un certain nombre de capteurs, que l'on appellera nœuds<sup>3</sup>, cherchent à communiquer avec un nœud central, qui stockera les informations ou les traitera. Plutôt que de lier tous ces capteurs de manière filaire, on utilisera souvent une communication radio.

Ce type de réseau diffère des réseaux plus « classiques » par l'importance des contraintes énergétiques et dimensionnelles des nœuds.

#### 1.1.2 Choix de la norme de communication

De nombreuses normes de communication radio existent : IEEE 802.11 (Wifi), IEEE 802.15.4 (ZigBee), Bluetooth, Bluetooth LE... La plupart des réseaux de capteurs utilisent la norme IEEE 802.15.4 couplé ou non avec une surcouche comme ZigBee par exemple. Ce choix est motivé par une composante extrêmement importante des réseaux de capteurs : la gestion de l'énergie.

En effet, ces capteurs sont, dans la majorité des cas, des systèmes isolés, alimentés par batterie. On les fait donc communiquer avec le moyen le plus économe. Les normes évoquées ci-dessus diffèrent essentiellement par l'encapsulation des données. Ainsi l'entête d'un paquet Wifi contient au moins 30 octets, quand un paquet respectant la norme IEEE 802.15.4 n'en a que 11. Comme

---

1. Information : Ce que l'on veut transmettre, partie utile de la communication (ex : valeur de la température dans le cas d'un thermomètre). Différenciée de la donnée car plus générale : on peut envoyer une information grâce à plusieurs paquets...

2. Donnée : Élément du paquet que l'on veut transmettre : partie utile de la communication (ex : valeur de la température dans le cas d'un thermomètre)

3. Nœud : Dans un réseau on parle de nœud pour évoquer un point "d'arrêt" ou un carrefour pour les données : lorsqu'un paquet arrive à un nœud, il est redirigé vers une maille ou une autre du réseau, voire vers le composant jouant le rôle de nœud s'il est destinataire.

la transmission d'un octet demande une certaine énergie, d'autant plus grande que l'émission est aérienne, plus le paquet sera petit, moins il utilisera d'énergie. A cela s'ajoute d'autres aspects liés à la couche physique (manière dont le signal est modulé, annoncé...). Bien qu'envoyer des données brutes (sans entêtes, ne respectant aucune norme) serait plus économe, dans un souci de compatibilité et de connaissance de la source de la donnée, nous sommes forcés d'utiliser une certaine norme. Ceci motive le choix usuel de la norme IEEE 802.15.4.

### 1.1.3 Méthode de communication

De nombreux moyens de communication existent, dont le choix est essentiellement fait par la taille du réseau, en terme numéraire ou d'étendue géographique.

Dans le cas d'un réseau très étendu, la faible portée d'une technologie sans fil oblige à mettre en place des rebonds. Ainsi le capteur 1 envoie sa donnée au nœud central via le nœud 2 puis le nœud 3. Chacun des nœuds intermédiaires (capteur ou non), lorsqu'il reçoit le paquet, doit l'analyser afin de comprendre qu'il n'est pas la destination et le renvoyer au nœud suivant.

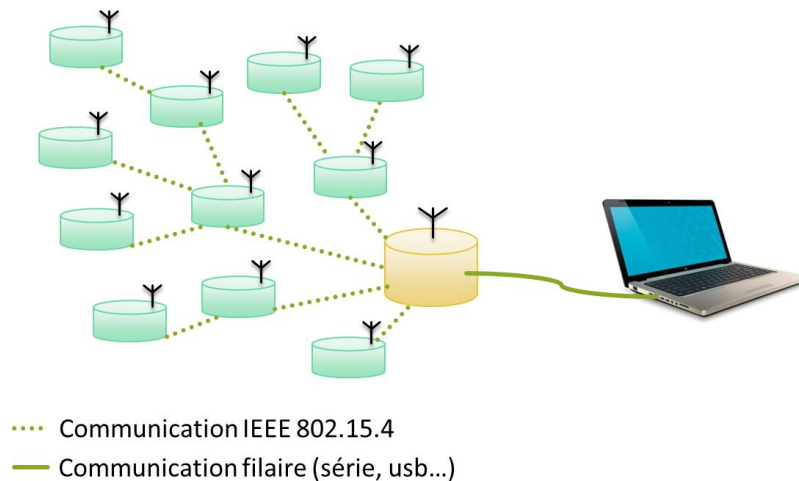


FIGURE I-1.1 – Schéma simplifié d'un réseau WSN

Dans le cas d'un réseau contenant de nombreux capteurs, il faut décider de la synchronisation des envois de données. S'il y a  $N$  capteurs, les capteurs envoient-ils un paquet tous les  $N+1$  unités de temps (la première servant à les synchroniser)? Ou bien demandent-ils la parole au nœud central qui la leur donne pour qu'ils envoient la donnée? Dans le premier cas, on aurait alors une supertrame de la forme donnée en figure I-1.2.

Signal de synchronisation (nœud central)	Paquet du nœud 1	Paquet du nœud 2	Paquet du nœud 3	...	Paquet du nœud $N-1$	Paquet du nœud $N$
$t_0$	$t_0 + \frac{1}{N+1} * \Delta T$	$t_0 + \frac{2}{N+1} * \Delta T$	$t_0 + \frac{3}{N+1} * \Delta T$	...	$t_0 + \frac{N-1}{N+1} * \Delta T$	$t_0 + \frac{N}{N+1} * \Delta T$

FIGURE I-1.2 – Forme d'une supertrame,  $\Delta T$  est la durée de la supertrame

Ces interrogations font partie des nombreuses questions auxquelles l'ingénieur créant l'architecture du réseau devra préalablement répondre afin de paramétrer les capteurs de la bonne manière.

### 1.1.4 Connexion au nœud central

En général, le nœud central et la machine stockant et effectuant les calculs ne sont pas le même système. On trouvera plutôt un nœud central qui reçoit l'ensemble des communications des nœuds

extérieurs et qui retransmettra l'information au centre de calcul. Le lien entre les deux est souvent filaire car plus simple à mettre en place (techniquement) et plus économe. Cependant la présence d'un fil n'est pas toujours compatible avec l'environnement auquel on doit faire face (milieu isolé, hostile ou dangereux) et une connexion sans fil avec le centre de calcul serait préférable. Comme celui-ci est souvent un ordinateur classique, il ne peut pas recevoir les données encapsulées selon la norme IEEE 802.15.4 : une solution simple est de transmettre l'information en Wifi (qui suit la norme IEEE 802.11).

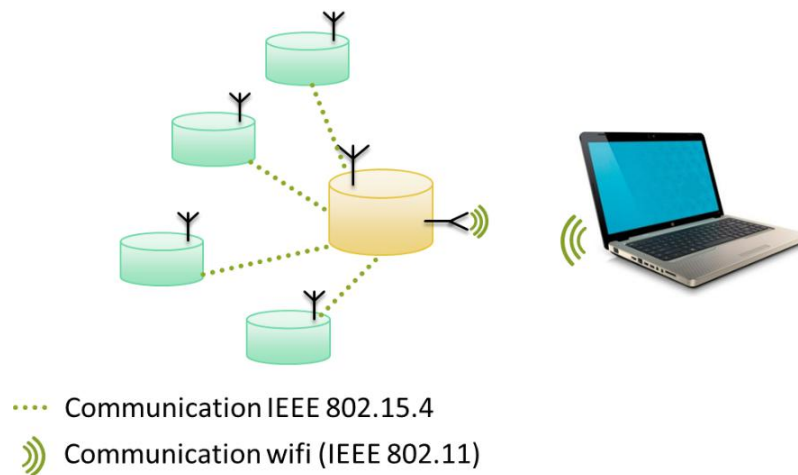


FIGURE I-1.3 – Réseau WSN avec nœud central multistandard

### 1.1.5 Objectifs du PAr

La méthode de conception des capteurs communicant suivant la norme IEEE 802.15.4 est connue et très répandue. Le but du projet est de lier ce réseau sans fil à un ordinateur classique via une carte Wifi. Il faut donc concevoir un module de communication permettant de recueillir les paquets envoyés par l'ensemble des capteurs, et de renvoyer cette information à l'ordinateur. Cela comprend donc la définition d'un moyen de synchroniser l'envoi des paquets, puis la conversion d'une norme à l'autre.

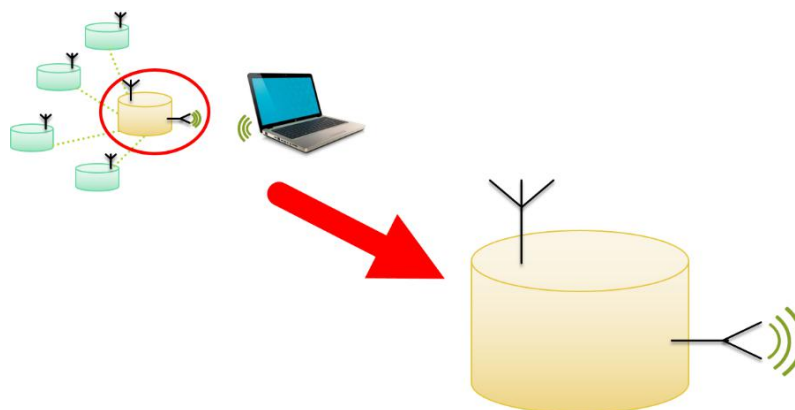


FIGURE I-1.4 – Restriction du projet au nœud central



## 2 Inscription de la thématique dans le Laboratoire INL, auquel je suis rattaché

### 2.1 Organisation

Crée en 1976, le Laboratoire d'Électronique, Automatique et Mesures Électriques (LEAME) est devenu en 1995 une unité mixte (UMR 5512) de l'École Centrale de Lyon et du CNRS – Département des Sciences pour l'Ingénieur (SPI). Le 1<sup>er</sup> janvier 1999, le LEAME a changé de nom et est devenu le LEOM : Laboratoire d'Électronique, Optoélectronique et Microsystèmes. Le 1<sup>er</sup> janvier 2007, le LEOM (ECL), le LENAC (UCBL), le LPM (INSA) et CPE ont fusionné pour créer l'Institut des Nanotechnologies de Lyon (INL), laboratoire multi-sites sur l'Ecole Centrale de Lyon et le parc scientifique de la Doua. L'INL représente environ 204 personnes dont 58 enseignants chercheurs, 18 chercheurs CNRS, 32 personnels techniques et une centaine de doctorants et visiteurs.

### 2.2 Domaines de recherches

L'INL est un laboratoire de recherche fondamentale et appliquée dans le domaine des micro- et nanotechnologies. Sa vocation est de mener des recherches, s'étendant des matériaux aux systèmes, permettant l'émergence de filières technologiques complètes pour plusieurs secteurs d'application (les semi-conducteurs et la microélectronique, les télécommunications, l'énergie, la santé, la biologie, le contrôle industriel, la défense, l'environnement).

Les programmes de recherche s'appuient pour la plupart sur les moyens de la plate-forme technologique lyonnaise Nanolyon.

Les recherches sont articulées autour de quatre grands axes thématiques :

- Matériaux
- Électronique
- Photonique & Photovoltaïque
- Biotechnologie et santé

C'est dans le département Électronique de l'INL que se déroule mon PAr.

---

## *Deuxième PARTIE*

# ÉTAT DE L'ART

---

Le problème que l'on se propose de résoudre semble relativement commun : passer d'une norme de communication à l'autre. C'est en effet une problématique à laquelle l'administrateur d'infrastructures de communications est souvent confronté. Une borne Wifi est souvent reliée par Ethernet au réseau, les paquets échangés doivent donc être convertis entre la norme IEEE 802.3 (Ethernet) et IEEE 802.11.

En cherchant sur internet et parmi les solutions commerciales, on s'aperçoit rapidement que si le concept est assez basique, presque aucune solution n'existe.

## 1 Solution commerciale

### 1.1 GainSpan GS2000

L'ensemble des données et diagrammes de cette section proviennent du document GS2000 Chip Product Brief [1].

#### 1.1.1 Présentation

GainSpan est la seule entreprise qui produit et vend des modules de conversion d'une norme à l'autre. Leur produit est une puce à basse consommation (durée de vie de plusieurs années) compatible IEEE 802.15.4 et IEEE 802.11b/g/n. Il comprend un processeur pour les programmes gérant le réseau, ainsi que de la mémoire, un contrôleur d'accès...

GainSpan vend ce circuit embarqué entièrement programmé : il contient un programme complet permettant (d'après leurs documents) de gérer de manière complète et exhaustive un réseau sécurisé respectant les deux standards présentés. On soulignera aussi que la puce est capable de tracer sa position.

Les applications ciblées par ce circuit embarqué sont les domaines évoqués plus haut, à savoir la santé, les applications de surveillance, l'automatisation...

#### 1.1.2 Caractéristiques techniques

- Dual-Core Processors 32-bits ARM (120 Mhz)
- Fréquences :
  - 802.11n (< 72.2 Mbps)

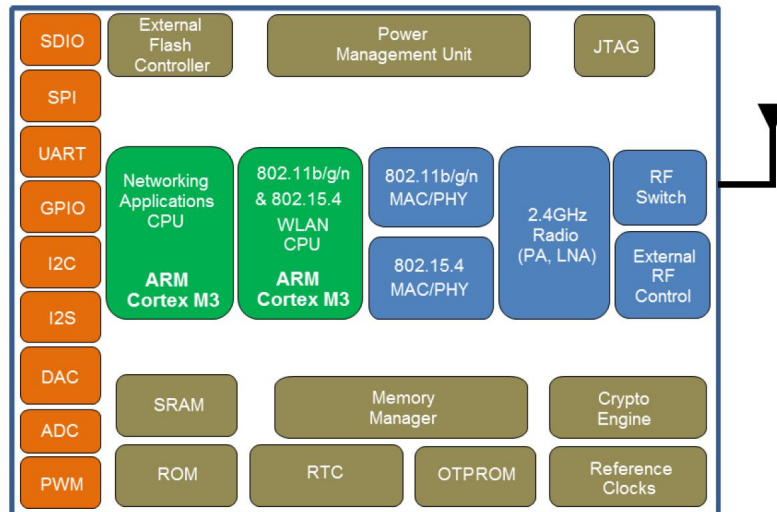


FIGURE II-1.1 – Schéma bloc du composant (d'après [1])

- 802.11g (6, 9, 12, 18, 24, 36, 48, 54 Mbps)
- 802.11b (1, 2, 5.5, 11 Mbps)
- 802.15.4 (250 Kbps)
- Sécurité : 802.11i, WPA/2-Personnel et Entreprise, Authentification EAP, 802.1x, cryptage TKIP et AES. Gère aussi les cryptages TLS, SSL, HTTPs, PKI et certificats.
- Protocoles gérés : TCP, UDP, IPv4/6, TLS (client/serveur), SNMP, DHCP (v4/6, client/serveur), DNS (client/serveur), mDNS, DNS-SD, HTTP/HTTPs (client/serveur), XML Parser.
- Températures supportées : -40 à +85°C
- Entrées/Sorties : UART, SPI, SDIO, I2C, I2S, ADC, DAC, PWM, GPIOs

### 1.1.3 Bilan

Cette solution, bien que très complète, n'est pas satisfaisante pour notre application. En effet ce produit est plutôt cher (environ 2000 euros) et présente l'inconvénient d'être verrouillé. Nous ne pourrions pas en modifier le code et l'optimiser pour nos utilisations.

## 2 Solutions académiques

### 2.1 Études généralement réalisées

La plupart des études [2] [3] s'inscrivent dans les thèmes de recherche liés à la veille médicale automatisée. L'objectif final de tels projets est de permettre à différents appareils de mesure de contrôler de manière autonome divers signaux de l'organisme afin de renvoyer un suivi de la santé du sujet en temps réel.

Il est souvent question d'une utilisation duale d'un réseau suivant la norme IEEE 802.15.6 ou IEEE 802.15.4 (deux normes très proches) et d'un autre suivant la norme IEEE 802.11. Le premier permet de lier le capteur à une base à proximité de l'individu, sans que les fréquences de communication n'interfèrent avec le matériel médical ou n'affecte l'organisme par exemple, tout en conservant une grande autonomie d'énergie. Ce type de réseau est appelé WBAN (pour Wireless Body Area Network) dans le cas du 802.15.6 et WPAN (pour Wireless Personal Area Network) dans le cas du 802.15.4. Le second réseau, de type WLAN (pour Wireless Local Area Network), relie le hub où sont centralisées les communications du WBAN/WPAN au réseau de l'hôpital (sur

lequel un serveur est prêt à recevoir les informations).

On peut trouver aussi généralement le détail du fonctionnement du beacon (moyen de gérer la communication pour éviter les chevauchements) ainsi que les priorisations dans le WBAN ou WPAN, et dans le WLAN.

L'étude de Saeed Rashwand et Jelena Masic [3] mène au choix de n'utiliser qu'une seule interface, régulièrement configurée pour émettre/recevoir selon la norme IEEE 802.15.6 puis IEEE 802.11e. Néanmoins, la mise en œuvre n'est pas abordée.

## 2.2 Un intérêt pour la compatibilité physique de ces deux types de normes

Les travaux de Wei Yuan [4] et Jelena Masic [2] [3] s'attardent longuement sur la compatibilité des réseaux WPAN ou WBAN avec les réseaux WLAN. Il semblerait en effet que ces différents types d'ondes radio interfèrent les uns avec les autres et limitent leur cohabitation. Cependant les études sont parfois contradictoire et finalement, aucune étude ne tranche définitivement sur la question.

Ceci étant un vaste sujet de traitement du signal et ne pouvant être traité dans ce projet, je l'ai seulement gardé à l'esprit comme une limite de fonctionnement de l'interface finale.

## 2.3 Développement d'un Framework du fonctionnement du réseau

Le fonctionnement des bridges WPAN/WLAN sont peu présentés dans les publications académiques.

ART-WiSe [5] (pour Architecture for Real-Time communications in Wireless Sensor Networks) est un Framework définissant le fonctionnement de réseaux WSN très adaptables pour des applications de grande envergure avec de fortes contraintes. Il permet d'obtenir une réactivité en temps réel (pour les mises à jour, le monitoring...), la fiabilité, l'adaptabilité (le nombre de nœuds, leur position et leur activité doit pouvoir changer sans affecter le réseau, en cas de congestion du réseau, il doit de lui-même pouvoir s'ajuster), l'efficacité en terme de coût et d'énergie.

Leur interface 802.15.4/802.11 est modélisée comme présentée sur la figure II-2.1.

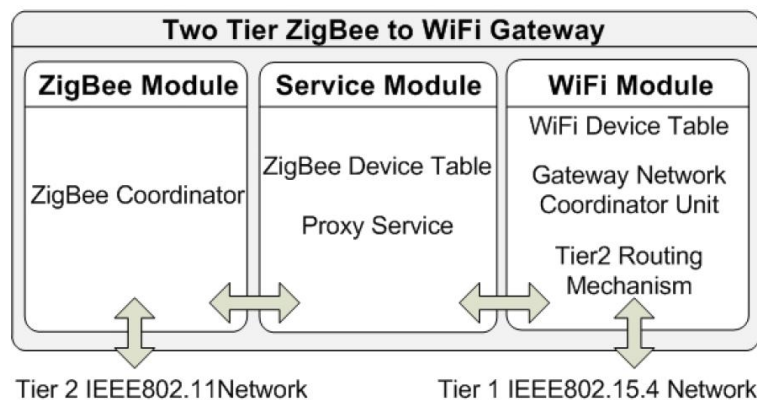


FIGURE II-2.1 – Division des modules de l'interface ZigBee (d'après [5])

Dans leur travail, l'interface sert aussi de routeur : la table ZigBee Device Table lie chaque

adresse du réseau WSN (IEEE 802.15.4) à un nœud et son état (batterie, émission,...), la table Wifi Device Table lie chaque adresse IP à une adresse MAC et permet de différencier les messages de management de ceux qui portent une donnée à transmettre.

Le cheminement logique d'une donnée est donné en figure II-2.2.

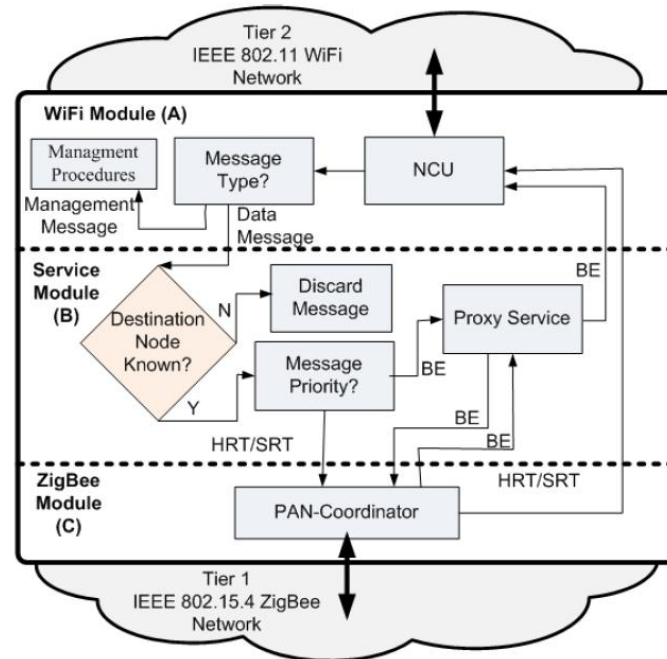


FIGURE II-2.2 – Cheminement des données (d'après [5])

La norme Wifi utilisée est la IEEE 802.11e et d'après l'article, la IEEE 802.11s devrait être plus efficace pour cette utilisation<sup>1</sup>. Les composants utilisés sont produits par l'entreprise Crossbow Technology, Inc. : Crossbow MICAz Mote et Crossbow Stargate. La programmation est réalisée sous TinyOS.

### 3 Bilan

On constate que peu de solutions sont disponibles à la vente. De plus les propriétaires de ces solutions ne laissent pas à disposition d'explications sur le moyen de conception utilisé. Les publications académiques expliquent quant à elles majoritairement la conception logique de ce type de module-interface, mais ne rentrent pas dans le détail de leur fabrication et de leur conception technique. Enfin, il semble que l'interface IEEE 802.15.4/IEEE 802.11 (ie. WPAN/WLAN) soit peu mise en place. La plupart des publications traitent en effet de l'interfaçage des normes IEEE 802.15.6/IEEE 802.11 (ie. WBAN/WLAN).

1. IEEE 802.11e : variante de IEEE 802.11 axée sur le transport d'informations vocales et vidéos

---

## *Troisième PARTIE*

# LE PROJET : DÉMARCHE ET OPÉRATIONS

---

## 1 Démarche générale

Le sujet d'étude est déjà restreint à son minimum : le nœud central servant de passerelle entre le WPAN et le WLAN.

Dans un premier temps, il faut être sûr de pouvoir créer des interfaces communicant sur des réseaux de type WPAN ou WLAN uniquement. Une fois que ce type d'interface mono-standard est développé, il est théoriquement aisé de les assembler sur un seul système.

Dans un second temps, il faut définir une procédure logique et fonctionnelle permettant de lier les deux interfaces et de faire communiquer les deux réseaux de manière effective. Cela revient à définir un protocole de communication.

## 2 Normes de réseaux

Les deux interfaces à mettre en œuvre suivent des normes IEEE. L'utilisation d'une norme a deux intérêts majeurs :

- Tout nœud du réseau, quelque soit son origine, sa marque et sa date de conception, s'il est certifié suivant la norme indiquée, pourra communiquer via notre passerelle.
- Il n'est pas nécessaire de réinventer un protocole de communication, il suffit de suivre la norme.

Néanmoins, pour certaines applications, il peut être judicieux d'adapter une norme aux besoins, voire de créer un protocole à partir de zéro.

Dans le cas de ce projet, l'utilisation de la norme IEEE 802.11 pour le côté WLAN est obligatoire pour pouvoir communiquer avec un ordinateur du marché. De même, la IEEE 802.15.4 est nécessaire pour le côté WPAN, afin que la passerelle puisse être liée à un réseau WSN classique.

La suite de ce chapitre présentera rapidement quelques éléments des deux normes et les choix effectués.

### 2.1 Organisation de la communication

Afin que chaque nœud du réseau puisse communiquer avec son interlocuteur, il faut ordonner la communication. Pour cela, différents fonctionnements ont été pensés, dont les principes sont

repris par la majorité des normes de communication. On s'intéresse ici uniquement aux normes IEEE 802.11 et IEEE 802.15.4.

### 2.1.1 Qui donne la parole ?

**Non-Beacon : la compétition ou CSMA/CA** Généralement, une sorte de compétition a lieu entre les différents nœuds souhaitant s'exprimer. Chacun, avant de s'exprimer, attend pendant une courte durée aléatoire puis vérifie que le canal de transmission est libre. Si c'est le cas, il s'exprime, empêchant alors les autres de communiquer durant le temps de la transmission. Si au contraire le canal est occupé, le nœud diffère sa communication et recommence tout le processus (voir figure III-2.1). Ce type de mécanisme est appelé CSMA/CA (pour Carrier Sense Multiple Access with Collision Avoidance).

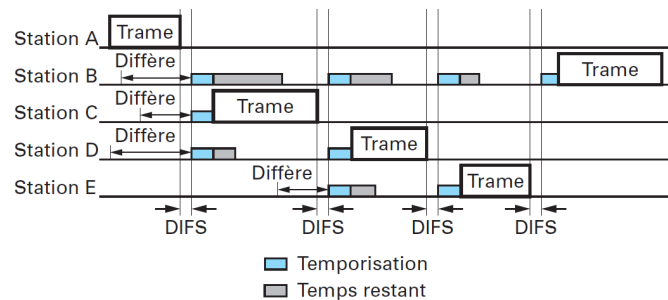


FIGURE III-2.1 – Mécanisme du CSMA/CA avec temporisation (d'après [6])

Le temps de latence aléatoire placé avant de vérifier que le canal de transmission est libre permet de limiter les chances de deux vérifications simultanées par deux nœuds différents, entraînant deux communications simultanées et donc incompréhensibles.

**Beacon** Dans des réseaux de petite taille, on détermine parfois un nœud central autour duquel les autres nœuds sont connectés, comme un réseau en étoile. Dans un tel cas, c'est souvent ce nœud central qui définit l'ordre de communication. On parle alors de PCF (pour Point Coordinated Function).

Régulièrement, le nœud central émet une balise reconnaissable par tous et permettant de les synchroniser dans le temps : le Beacon.

On peut alors imaginer plusieurs stratégies :

- On définit un ordre de parole pour chaque nœud. Cela est plus simple à mettre en œuvre, mais présente l'inconvénient qu'à chaque ajout de nœud, il faut redéfinir l'ordre de passage, augmenter la taille de la fenêtre de communication... Cela revient finalement à une supertrame telle que présentée en figure I-1.2.
- Lorsqu'un nœud veut s'exprimer, il en fait la demande sur le réseau. Le nœud central accepte alors la communication et lui alloue un emplacement temporel dit GTS (pour Guaranteed Time Slot) dans la zone CFP (pour Contention Free Period). Ce fonctionnement est synthétisé dans la figure III-2.2. Dans la zone CAP (pour Contention Access Period), les communications se font selon le modèle CSMA/CA. C'est ici que chaque nœud réserve les emplacements GTS et leur taille.

### 2.1.2 Le paquet est-il reçu ?

Les normes de communication ont prévu un moyen de vérifier si un paquet a été reçu ou non.

En effet, un *acknowledgement* peut être demandé par l'émetteur : une fois que le paquet est reçu, le destinataire retourne un petit paquet indiquant la bonne réception. Cette option est disponible dans les deux normes IEEE 802.15.4 et IEEE 802.11.

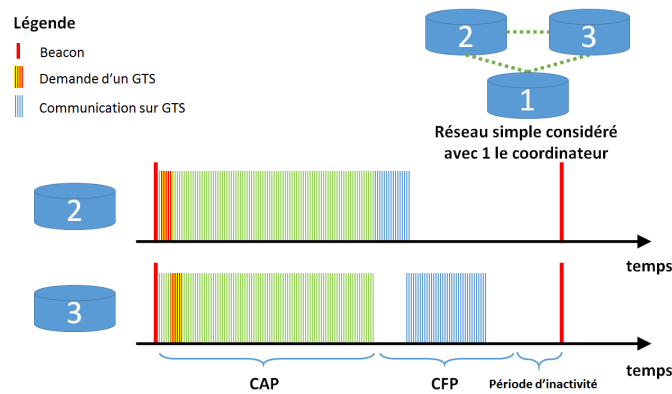


FIGURE III-2.2 – Mécanisme de communication avec CAP et CFP

## 2.2 Architecture des paquets

Les nœuds du réseau communiquent en mettant sur le canal de transmission des paquets qui suivent la norme imposée. Ces paquets ont donc une structure particulière qui permet aux nœuds du réseau de déterminer s'ils sont destinataires ou non, et s'ils le sont de lire la donnée puis d'accuser réception si l'*acknowledgment* est activé, etc.

Comme le montrent les figures III-2.4 et III-2.3, dans les deux cas le paquet est composé :

1. d'un identifiant et d'un numéro de séquence
2. d'un header, contenant de nombreux paramètres (non tous représentés ici) tels que l'adresse de destination, d'émission, la demande d'*acknowledgment*, la taille du paquet...
3. de la donnée (frame payload)
4. d'un bit de contrôle

Il faut imaginer le paquet comme une sorte de poupée gigogne. En effet, les normes de communications sont définies selon plusieurs couches (ou layers) allant de 1 à 7, suivant le modèle OSI. Les figures III-2.4 et III-2.3 représentent le paquet vu de la couche 3. Ainsi le payload n'est pas seulement la donnée à transmettre mais aussi un paquet vu de la couche 2 (MAC) structuré avec un header et un payload (contenant celui-ci un paquet vu de la couche 1 PHY).

On n'entrera cependant pas plus en détail dans le modèle OSI et dans les normes sommairement présentées ici.

Octets : 2	1	0/2	0/2/8	0/2	0/2/8	variable	2
Frame control	Sequence number	Destination PAN Id	Destination address	Source PAN Id	Source address	Frame payload	FCS
MHR						MAC payload	MFR

FIGURE III-2.3 – Structure d'un paquet suivant la norme IEEE 802.15.4

Octets : 2	2	6	6	6	2	6	variable	2
Frame control	Duration & Sequence number	Transmitter address	Receiver address	Source address	Sequence control	Destination address	Frame payload	CRC
MHR							MAC payload	MFR

FIGURE III-2.4 – Structure d'un paquet suivant la norme IEEE 802.11



## 2.3 Choix de fonctionnement

### 2.3.1 Interface WLAN

L'interface WLAN doit être compatible avec les ordinateurs du marché. Elle suivra donc les normes de ces appareils :

- Beacon
- Acknowledgement
- ...

### 2.3.2 Interface WPAN

Sur cette interface, on pourra chercher à simplifier au maximum les processus de communication, quitte à les améliorer par la suite pour augmenter la compatibilité.

Ainsi, l'utilisation d'une supertrame semble idéale : le temps est décomposé en unités temporelles définies selon le nombre de capteurs. On obtient alors des supertrames telles que présentées en figure I-1.2.

On définit donc dans le nœud central la taille (en nombre de nœuds) du réseau et les échelles de temps en découleront.

## 3 Choix des composants

### 3.1 Composants MICROCHIP

Le laboratoire travaille régulièrement avec du matériel de chez MICROCHIP. Avoir pris en main le développement sur microcontrôleurs sur des composants de cette marque était un argument de plus pour rester sur ce choix.

### 3.2 Modules de communication

Afin de pouvoir nous consacrer uniquement à l'implémentation des solutions, nous avons décidé de choisir des modules communicants chez des fournisseurs. Le design de l'antenne ou encore les processus de communication sont ainsi déjà mis en œuvre, ce qui a permis de gagner du temps (et d'être sûr de la qualité des composants utilisés).

Le laboratoire avait déjà travaillé sur le module MRF24J40 qui permet d'émettre et de recevoir des communications WPAN. Pour pouvoir utiliser ce qui avait déjà été fait, il était préférable de conserver ce module pour l'interface WPAN. Du côté WLAN, MICROCHIP propose le MRF24WB0MA et MRF24WG0MA (figure III-3.1) qui peuvent se brancher sur une carte de développement.

### 3.3 Microcontrôleur maître

Pour lier les modules de communication, il fallait choisir un type de contrôleur. Ici encore, le choix a été fait rapidement car le laboratoire disposait de *PIC 32 Starter Kit* de chez MICROCHIP (voir figure III-3.2).

Développer sur ce composant permet de tester rapidement le code. De plus, il est fait pour fonctionner avec les modules de communications choisis précédemment (au moins avec le module Wifi). De plus, les modules se branchent rapidement et il n'est pas nécessaire de concevoir une carte imprimée de A à Z pour développer (voir figure III-3.3). Enfin, si un composant est abîmé, il est aisé de le changer.



FIGURE III-3.1 – Module MRF24WG0MA : Interface WLAN

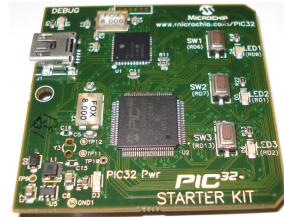


FIGURE III-3.2 – Pic 32 Starter Kit



FIGURE III-3.3 – Carte de développement (travail sur un module)

## 4 Les opérations : détails et méthode de programmation

Le projet se découpe donc en trois phases quasiment indépendantes. Il faut programmer d'une part le comportement des deux interfaces sans fil. Chacune a ses propriétés propres et demande donc un traitement particulier. D'autre il faut développer la manière dont l'information transitera d'un réseau à l'autre, c'est à dire d'une interface à l'autre. La seule contrainte commune est que ces trois éléments doivent être sur une unique et même carte et donc être développé sur le même microcontrôleur.

### 4.1 Module WLAN

Ce module (MRF24WB0MA ou MRF24WG0MA) est livré sans le détail des registres. Néanmoins des programmes complets et complexes sont à disposition des développeurs, basés sur les bibliothèques MLA (MICROCHIP Libraries for Applications). On part donc d'un programme sensé être fonctionnel que l'on simplifie afin de lui donner le fonctionnement voulu.

De plus une documentation des bibliothèques MLA existe, ce qui peut permettre un développement plus intelligent et maîtrisé.

Le programme d'exemple est extrêmement modulaire afin de pouvoir servir à la majorité des applications (serveurs web, DNS, hôte de réseau ad-hoc, etc.). Ainsi, une fois que le programme complet non modifié est fonctionnel, il ne reste plus qu'à choisir parmi les fonctionnalités ce que l'on souhaite conserver.

La documentation ne donnant pas de fonction permettant de transmettre un simple paquet suivant la norme IEEE 802.15.4, il est nécessaire de ruser et d'afficher les données reçues dans un fichier du serveur web (CVS ou HTML) par exemple.

## 4.2 Module WPAN

On gère le module WPAN en paramétrant directement les valeurs des registres STR. Cela est fastidieux mais permet de comprendre exactement ce qui est fait sur le module.

Un programme permettant de faire communiquer (émission/réception) ce type de module a été mis en place lors de la prise en main du microcontrôleur. Celui-ci était développé sur une architecture 16 bits, implémenté à l'aide du compilateur propriétaire CCS.

Il suffirait donc à priori d'adapter le code utilisé précédemment au nouveau microcontrôleur. Comme l'architecture et le compilateur ne sont pas les mêmes (le module Wifi impose une architecture 32 bits et le compilateur MICROCHIP), le travail à réalisé n'est cependant pas aussi trivial ce à quoi on pourrait s'attendre.

En effet toutes les fonctionnalités basiques (communication avec le module, paramètres des registres, interruptions, etc.) doivent être développée à nouveau. Les bibliothèques du module WLAN implémentent déjà ces fonctionnalités, malheureusement celles-ci sont optimisées pour les modules Wifi ciblés et difficilement généralisables au module WPAN. De plus la complexité du code fait qu'il est difficile de s'en inspirer.

Il est donc nécessaire de se plonger dans la documentation du microcontrôleur maître, de croiser les informations avec la documentation du module afin de comprendre les fonctions à développer, puis de chercher dans la documentation très générale du compilateur la manière de les mettre en œuvre.

Pour résumer, il est nécessaire de :

1. définir et configurer la communication SPI<sup>1</sup> entre le maître et le module
2. définir les interruptions<sup>2</sup> que le module est susceptible de générer
3. réveiller le module
4. le réinitialiser
5. le configurer afin qu'il suive le comportement voulu et que ses paramètres soient ceux souhaités

La partie concernant le module uniquement (réveil, initialisation) est assez simple en suivant la documentation et ne dépend pas du maître. Toute la difficulté est ici de permettre la communication (SPI et interruptions) entre le maître et le module.

## 4.3 Conversion et traitement des données : le liant

Une fois que l'on est sûr que les deux modules fonctionnent de manière indépendante à l'aide du même microcontrôleur maître, il ne reste plus qu'à créer les routines de conversion et traitement des données.

En parallèle des réceptions et émissions de paquets, le microcontrôleur désencapsule puis traite les données. Il gère aussi la manière dont le temps de parole est distribué. On cherche à réaliser un programme dont le fonctionnement est schématisé en figure III-4.1.

## 5 État actuel du projet

La figure III-5.1 fait état des fonctionnalités générales mises en places, en cours d'implémentation ou non développées. Toutes ces fonctionnalités ne sont pas aussi longues à mettre en place

1. Serial Peripheral Interface. Une liaison SPI est un bus de données série synchrone où le canal de transmission est bidirectionnel.

2. Une interruption permet d'interrompre littéralement le déroulement du programme global. On exécute alors la fonctionnalité qui lui est associée (exemples : traiter le paquet reçu, envoyer un paquet, etc.).

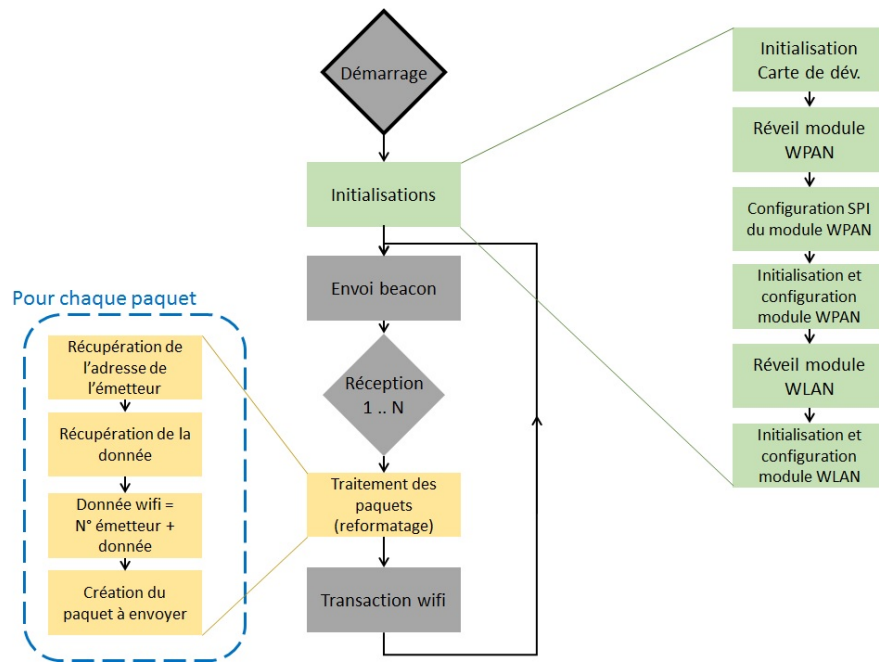


FIGURE III-4.1 – Diagramme schématique du programme

les une que les autres. Ainsi, le tri des fonctions utiles et le traitement des données est bien plus rapide à effectuer que les configurations générales des modules ou la créations fonctionnalités de communication entre le maître et le module.

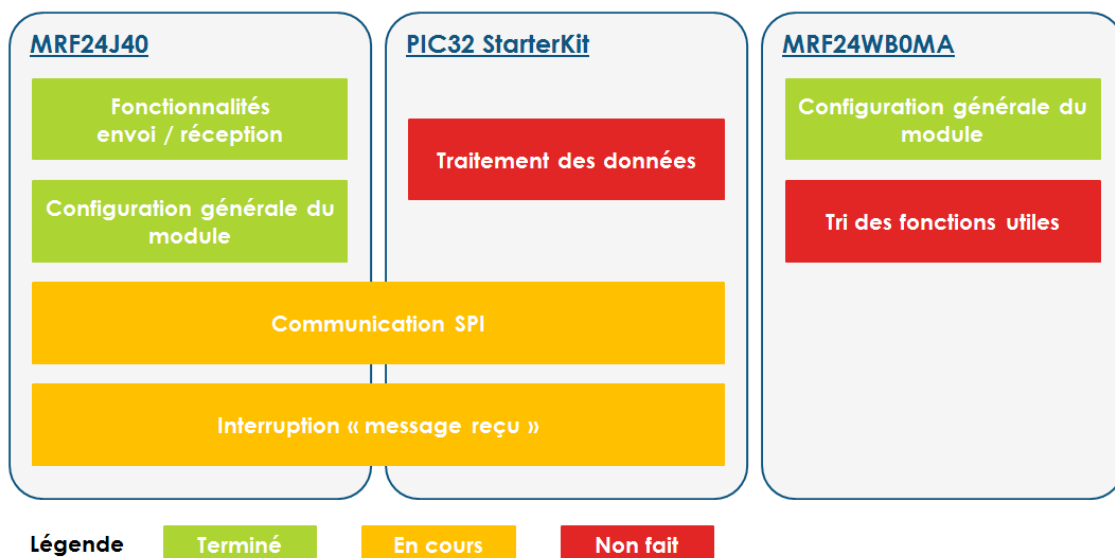


FIGURE III-5.1 – Bilan de l'avancement du projet

---

## *Quatrième PARTIE*

# DESCRIPTION DE L'AVANCEMENT ET PROBLÈMES RENCONTRÉS

---

## 1 Prise en main, documentation et réflexions sur le projet

J'ai entre-aperçu l'électronique des microcontrôleurs dans un approfondissement de S7. J'ai alors proposé à M. Navarro de travailler sur un projet dont il serait commanditaire et tuteur. Il m'a présenté ce sujet de passerelle entre réseaux sans fils ayant des normes différentes. Ayant fait beaucoup de réseau durant mon mandat de président de l'association ECLAIR, mais essentiellement à travers d'applications sous Linux, j'ai été intéressé par ce projet qui me permettait de me plonger au cœur d'un composant de réseau.

J'ai commencé par appliquer ce que j'avais appris durant le semestre sur une carte, permettant de communiquer sur un réseau WPAN, utilisée par un des doctorant.

En parallèle de cette remise à niveau en C et sur les microcontrôleurs, j'ai étudié les deux normes et réfléchi à la manière de les rendre compatibles. C'est alors que nous avons décidé avec M. Navarro de commencer par utiliser des modules faciles à connecter sur une carte de développement, avant de souder un carte complète à la fin du projet si le résultat était satisfaisant.

Le module MRF24WG0MA (voir III-4.1) m'a posé quelques problèmes dès le début lorsque j'ai constaté que sa documentation ne détaillait pas ses registres. J'ai donc téléchargé les bibliothèques d'exemple (*Microchip Libraries for Applications*). Aucune documentation de la structure de ces bibliothèques n'existe sur le site du fabricant, seule une source académique [7] (trouvée en février) a donné quelques informations.

## 2 Toussaint - Noël : recherche d'une erreur sur le module WLAN

Arrivé aux vacances de Toussaint, une grande partie de la conception était pensée et n'avait plus qu'à être développée. Avant de me lancer dans le développement de ma solution sur le module MRF24WG0MA, j'ai essayé de faire fonctionner le programme mis à disposition par MICROCHIP sans modification. Celui-ci ne fonctionnait pas sur le composant, et il m'a fallu plusieurs séances pour seulement comprendre d'où venait le problème.

Le programme de test crée un serveur grâce au module et de nombreuses connexions : il est

donc extrêmement dense. J'ai dû lire et décoder l'ensemble du code source pour enfin comprendre que l'initialisation générait une erreur. Il semble que le contrôleur maître ne réussissait pas à communiquer avec le module. Ceci compris, j'ai essayé pendant quelques séances de corriger la communication en vain. J'en ai tout de même conclu que le module était sans doute cassé et une commande du MRF24WG0MA et d'un deuxième similaire mais d'une version différente (le MRF24WB0MA) a été réalisée par M. Navarro.

Dans le même temps, quelques circuits ont grillé suite à des erreurs de manipulation, ce qui m'a fait perdre encore un peu de temps.

Après la période de Noël, nous avons reçu les nouveaux modules. Le MRF24WG0MA identique n'a pas fonctionné tandis que le MRF24WB0MA a pu traverser sans problème le programme et j'ai pu me connecter avec mon ordinateur au réseau WLAN établi. Une lettre dans la référence du composant (un G à la place d'un B) m'a donc fait perdre 2 mois, alors qu'aucune contre-indication n'était donnée sur le site de MICROCHIP.

### 3 Janvier - Mars : mise en place de la communication SPI sur le module WPAN

Lors de la commande de matériel, nous avons trouvé le module WPAN (MRF24J40MA) sous la même forme que le MRF24WB0MA, c'est-à-dire facile à connecter à la carte de développement.

J'ai alors adapté le code utilisé au début du projet afin qu'il puisse fonctionner avec le compilateur de MICROCHIP (MPLAB). Le plus difficile était de développer les communications SPI. En effet, les bibliothèques fournies pour l'autre module n'étaient pas compatibles et le compilateur utilisé lors de la prise en main du projet proposait des fonctions simplifiant le développement. Après de nombreux essais infructueux, ces fonctions ont enfin fonctionné.

Tout semblait fonctionner correctement et j'ai continué à adapter le code du projet de début d'année. Le microcontrôleur maître arrivait à communiquer avec le module, il n'a jamais pu réussir à comprendre sa réponse. En effet, à l'oscilloscope je voyais la réponse attendue, tandis que le microcontrôleur ne pouvait pas afficher cette réponse dans une variable.

### 4 Bilan : état actuel du projet

Actuellement, comme l'indique la figure III-5.1, le module WPAN est quasiment programmé. Il ne reste qu'à résoudre le problème auquel je suis confronté : faire en sorte que le microcontrôleur maître comprenne les réponses du module.

Le module WLAN est fonctionnel mais fait beaucoup trop de choses. Il faut simplifier le programme et définir la manière dont les informations sont communiquées à l'utilisateur via Wifi.

La communication entre les deux modules n'a plus qu'à être implémenté, ce qui ne peut être fait tant que les modules ne sont pas opérationnels de manière indépendante.

---

## CONCLUSION

---

Ce projet m'a permis d'approfondir mes légères connaissances en microcontrôleurs. Je n'avais avant cela suivi qu'un cours d'approfondissement sur le sujet en S7.

Il m'a fallu prendre des initiatives et tester des manipulations pour régler les problèmes auxquels j'étais confronté, au risque de griller certains composants. J'ai évidemment amélioré mes connaissances en développement C et ma culture informatique (réseaux, fonctionnement d'un processeur/contrôleur...).

Cela me donne un premier aperçu de la voie dans laquelle je suis en train de m'orienter pour la suite de mes études, à savoir l'électronique et la R&D.

Si je sais que je continuerai à un niveau ou à un autre l'électronique (à priori plutôt dans le traitement de l'information), le projet m'a montré que travailler sur des microcontrôleurs n'est pas si simple et ne me plaît pas vraiment. En effet, les entreprises les concevant n'ouvrent pas complètement l'accès à leurs outils et le fonctionnement interne des composants n'est pas normé. De plus, si on change de compilateur, une grande partie du travail doit être refaite. Ce sont des aspects qui m'ont beaucoup dérangé. Ayant l'habitude de travailler avec des solutions OpenSources, ce mercantilisme m'a semblé plutôt absurde. Le développement de ces outils, qui me semblait n'avoir que peu de valeur ajoutée face à la conception faite en amont, demande au contraire de nombreux savoirs et de se plier aux choix du constructeur, sans en comprendre toujours l'intérêt (par exemple : pourquoi la communication SPI est-elle ainsi sur ce module et pas sur celui-ci ?).

Peut-être est-ce le format du Projet d'Application : être seul, pendant un temps finalement assez court, qui me donne cette impression. Si mon bilan ne semble pas forcément positif et que le projet m'a finalement assez frustré, j'en ai tiré de nombreux enseignements, notamment pour mon orientation.

# Bibliographie

- [1] "GainSpan GS2000." <http://www.gainspan.com/gs1011soc1>.
- [2] J. Misic and V. B. Misic, "Bridging between ieee 802.15.4 and IEEE 802.11b networks for multiparameter healthcare sensing," Selected Areas in Communications, IEEE Journal on, no. 4, pp. 435–449, 2009.
- [3] S. Rashwand and J. Misic, "Bridging between IEEE 802.15.6 and IEEE 802.11e for wireless healthcare networks," Journal of Ad-Hoc Networks, Under Review.
- [4] W. Yuan, Coexistence of IEEE 802.11b/g WLANs and IEEE 802.15.4 WSNs : Modeling and Protocol Enhancements. PhD thesis, Delft University of Technology, 2011.
- [5] J. Leal, A. Cunha, M. Alves, and A. Koubaa, "On a IEEE 802.15.4/ZigBee to IEEE 802.11 gateway for the ART-WiSe architecture," in IEEE Conference on Emerging Technologies and Factory Automation, 2007. ETFA, pp. 1388–1391, 2007.
- [6] D. Trezentos, "Standard pour réseaux sans fil : IEEE 802.11," Techniques de l'Ingénieur Réseaux Locaux, no. TE-7375, 2002.
- [7] Microchip Inc., "Documentation des bibliothèques TCP/IP Stack." <http://www.egr.msu.edu/classes/ece480/capstone/fall11/group03/TCP/IP%20Stack%20Help.pdf>.