

Réseaux de capteurs : Système d'acquisition et de routage d'information

Sensor Networks : Information Acquisition and Routing System

Ahmed Habbani

Laboratoire LISIF-SYEL Université Pierre et Marie Curie 4 place Jussieu F - 75252 PARIS CEDEX 05 France, habbani_ahmed@yahoo.fr.

Olivier Romain

Laboratoire LISIF-SYE, Université Pierre et Marie Curie, 4 place Jussieu F- 75252 PARIS CEDEX 05 France.

Jamal El Abbadi

Laboratoire LEC Ecole Mohammadia d'Ingénieurs Avenue Ibsina B.P. 765 Agdal Rabat Maroc.

Patrick Garda

Laboratoire LISIF -SYEL Université Pierre et Marie Curie 4 place Jussieu F- 75252 PARIS CEDEX 05 France.

Résumé

Le présent article s'inscrit dans le cadre du développement d'un système original dans le domaine de la technologie d'information. Il introduit la conception, la spécification et la réalisation d'un nœud du réseau de capteur sans fils dans une architecture embarquée de type FPGA. Ce nœud permettra la construction d'un réseau d'unités autonomes mobiles capables de se déplacer dans des zones inconnues, inaccessibles ou hostiles à l'être humain (incendie, radiation, tremblement de terre, ...). Afin de collecter des données environnementales par différents capteurs (CCD, magnéto-résistifs, température, l'humidité...), et de les transmettre par routage à une unité de traitement distante. Pour cela nous avons développé et implémenté, dans un premier temps, des IPs d'acquisition permettant la mesure de certaines grandeurs telles que la température, l'humidité et/ou le monoxyde de carbone CO... puis nous avons intégré le module de routage basé sur le protocole proactif OLSR dans la plate forme de Co-Design. Des tests réels dans un réseau de capteurs de topologie variable ont été réalisés et ont permis de valider expérimentalement le fonctionnement de ce nœud.

Abstract

This article fits into development of an original embedded system. It introduces the design, specification, and realization of a wireless node sensor network in embedded FPGA architecture. This node will allow the construction of autonomous mobile network units which can move in unknowns, inaccessible or hostile areas for human being in order to collect environmental data by various sensors and transmits them by routing to a unit of distant process. For that, we have first developed and implemented IPs acquisitions which permits the measure of some grandeur like temperature, humidity and/or monoxide of carbon CO... Then we have integrated the routing module based on the OLSR proactive protocol in the Co-design platform. Real tests in network sensor of variable topology have been carried out and have permitted working node validation.

1. Introduction

Les capteurs sont devenus des éléments incontournables dans tous les systèmes où les informations issues de l'environnement extérieur sont nécessaires pour évaluer et agir (Pujolle, Salvatori, et al., 2004). De ce fait, avoir une connaissance précise et complète sur ce milieu exige le déploiement de plusieurs capteurs, et si possible, conjuguer toutes les informations renvoyées pour optimiser l'action à faire. Les avancées remarquables dans le domaine des télécommunications ont permis de supprimer les liaisons filaires de transmission fort encombrantes d'autant plus que les câbles d'alimentation ne sont plus indispensables, étant donné que la consommation de tels circuits n'a cessé de baisser. Notons qu'un réseau de capteurs est constitué d'un grand nombre d'unités appelées nœuds. Chaque nœud est composé principalement d'un ou plusieurs transducteurs, d'une unité de traitement et d'un module de communication, etc. Ces nœuds dialoguent entre eux selon une topologie (fixe ou mobile) du réseau afin d'acheminer les informations à une unité de commande hors de la zone de mesure. Cette configuration du réseau est généralement appelée, réseau ad hoc.

Toutes ces fonctionnalités nous permettent d'imaginer un système complexe construit autour de plusieurs capteurs sans fils, capables de s'adapter selon les évolutions rencontrées. Pour cela, nous avons conçu et réalisé un système original qui permet d'intégrer trois fonctions qui sont l'acquisition, le traitement et le routage de l'information autour d'une architecture embarquée de type FPGA. La figure 1 représente un schéma type de fonctionnement d'un tel système.



Figure 1. Architecture interne d'un nœud du réseau de capteurs sans fils

Cependant, la mise en place de ce système est confrontée à de nombreux problèmes parmi lesquels ceux liés au routage des informations vers une unité distante via les différents nœuds du réseau éventuellement mobile. En effet, l'acheminement des données nécessite la connaissance et la mise à jour de la table de routage de chaque nœud mobile afin d'optimiser la durée du transfert. Pour réaliser cette opération, nous disposons de différents algorithmes de type réactifs, proactifs ou hybrides. Certains de ses algorithmes ont été standardisés récemment par le groupe MANET.

D'autres contraintes apparaissent dans un système d'information embarquée. Elles sont généralement liées à la consommation d'énergie, ceci afin d'optimiser l'autonomie des nœuds qui constituent le réseau.

Dans le cadre de notre travail, pour répondre à ces problématiques, nous avons adopté et implémenté l'algorithme proactif OLSR dans une architecture embarquée nommée, MERITE (Mobile Embarqué Reconfigurable Intelligent et TELécommuniquant), qui constitue le nœud élémentaire d'un réseau de capteurs sans fils mobile.

L'organisation de cet article comprend 7 sections principales que nous rappelons ci-dessous.

Après une introduction générale, nous donnons dans la section 2, la problématique de cet article. La section 3 est dédiée à la présentation des différentes méthodes (proactifs, réactifs et hybride) de routage des données du groupe MANET. Dans la section 4, nous décrivons brièvement l'architecture de MERITE. La section 5 présente la méthodologie de conception que nous avons retenue. La section 6 est consacrée à la description de l'algorithme OLSR (Optimized Link State Routing Protocol), sa technique des relais multipoints et ses applications les plus connues, ainsi que son intégration dans la plate forme MERITE. La section 7 est le résultat de l'implémentation d'OLSR. Puis, nous finalisons notre article par des perspectives et une conclusion.

2. Les réseaux de capteurs : la problématique

L'évolution des technologies en électronique permettra de concevoir dans un futur très proche sur une même puce, tout un système électronique (SoC) hétérogène composé d'unités d'acquisition, de traitement et de communication. Cette révolution en matière d'intégration des systèmes électroniques permettra d'étendre le champ d'application des systèmes électroniques embarqués à celui des réseaux de capteurs de très grande envergure.

Actuellement, les réseaux de capteurs (Hac, 2003) fondés sur des architectures discrètes trouvent diverses applications dans les domaines de la robotique, de la domotique, de l'environnement, du transport et de la surveillance du territoire et des territoires ennemis, etc. Par exemple, l'université de Berkeley, Intel et DARPA (USA) ont conçu en collaboration un réseau constitué de 800 capteurs différents pour des applications de topologie. Chaque élément du réseau est constitué d'un capteur de température, un capteur de lumière, une batterie, un module RF T1000-10kb/s, un microprocesseur Risc 8bits 4MHz basse consommation et un OS appelé TinyOs (figure 2).

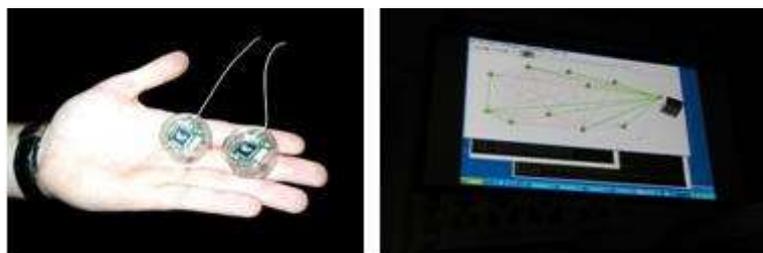


Figure 2. Réseau de capteurs sans fil pour des applications militaires

Un autre exemple est donné par l'université de Pennsylvanie qui a réalisé un réseau de capteurs pour contrôler la qualité des eaux (figure 3).

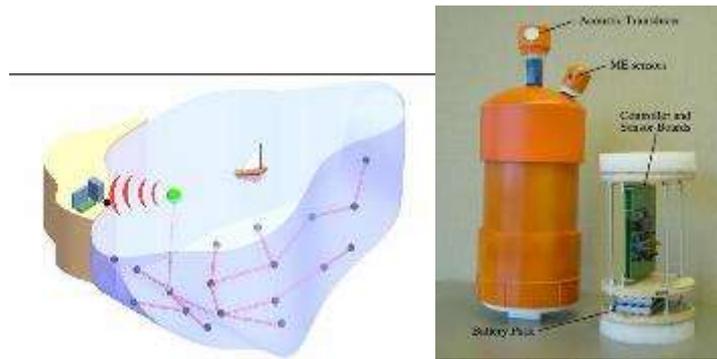


Figure 3. Système de contrôle de la qualité d'eau

Enfin, l'université de Pise en Italie, a réalisé un système composé de réseaux de capteurs pour contrôler des parcs naturels (Gaz, feux, animaux, ...), comme l'illustre la figure 4.



Figure 4. Système de contrôle des parcs naturels

Ces différentes applications permettent d'ores et déjà d'étudier et d'apporter des solutions aux différents problèmes liés à la gestion de réseaux de capteurs de grande envergure.

Les problèmes actuels sont essentiellement liés à l'autonomie par rapport au coût d'une communication lors de l'interrogation d'un capteur parmi N et des phases d'auto organisation du réseau lorsqu'une modification de la topologie survient ou initialement lors du largage du réseau. Les choix de l'architecture matérielle du capteur ainsi que l'algorithme de routage sont alors primordiaux afin de garantir une grande autonomie et l'envoi correct des données collectées en un point à un destinataire. Nous avons fait le choix de répondre au second problème en intégrant un algorithme de routage à une architecture flexible de type FPGA.

3. Le routage des données

3.1 Introduction

Dans cette partie, nous présentons un état de l'art des protocoles de routage proposés pour effectuer le routage dans les réseaux ad hoc. Notons que la plupart de ces protocoles sont en cours d'évaluation par Manet, groupe de travail spécialisé dans les environnements mobiles ad hoc.

La transmission de l'information dans un système embarqué peut se faire de deux façons comme le montre la figure 5.

L'envoi direct est possible lorsque les mobiles sont suffisamment proches les uns des autres pour que le signal reçu ne soit pas trop atténué. Chaque noeud est en lien étroit avec n'importe quel autre, et aucun intermédiaire ne peut s'interposer dans cette relation directe privilégiée.

L'envoi par routage se déroule entre des nœuds relativement éloignés, sujets à l'affaiblissement des signaux émis. Les nœuds jouent à la fois le rôle de client et de serveur, relayant de la sorte les paquets vers leur destination finale.

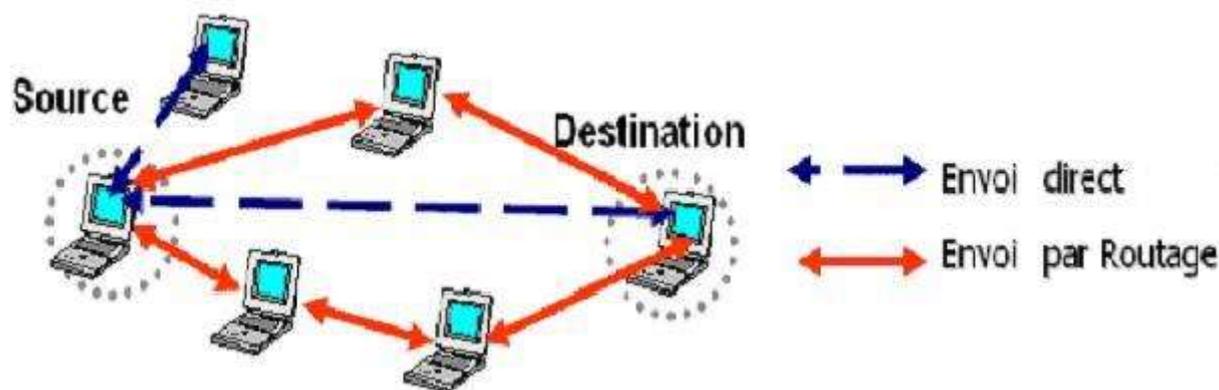


Figure 5. L'envoi direct et envoi par routage

En 1997, l'IETF, Internet Engineering Task Force, <<http://www.ietf.org/>> crée le groupe de travail MANET, Mobile Ad Hoc NETWORK, <<http://www.ietf.org/html.charters/manet-charter>> pour définir une spécification de protocole de routage pour les réseaux sans fil AD HOC. A notre connaissance, et d'après Paul Mühlethaler (Mühlethaler, 2002), c'est le seul organisme de normalisation à ce jour qui se penche sur ce problème. Cette spécification distingue entre 3 types de protocoles : réactifs, proactifs et hybrides.

3.2 Les protocoles réactifs

Les protocoles réactifs recherchent les routes (chemins) suite à la demande d'une application. Lorsque le réseau a besoin d'une route, une procédure de découverte globale des routes est lancée, dans le but d'obtenir une information spécifique, inconnue au préalable. Les routes sont détruites lorsqu'elles ne sont plus utilisées.

On dénombre essentiellement deux algorithmes :

- AODV (Ad hoc On demand Distance Vector Routing) qui est un protocole de type vecteur de distance, utilise la technique Expanding Ring qui limite le nombre de sauts des paquets de recherche de route à partir du nœud source C.E. (Perkins, Belding-Royer et al., 2002).
- DSR (Dynamic Source Routing) qui utilise la technique de routage par source et possède le même mécanisme qu'AODV pour limiter le nombre de sauts d'une recherche de route (Johnson, Maltz et al.).

3.3 Les protocoles proactifs

Les protocoles proactifs établissent les routes à l'avance et sont particulièrement utilisés dans un réseau dense (réseaux de grande taille). Ils maintiennent en permanence une vision globale de l'état du réseau ad hoc grâce à une gestion périodique des tables de routage et l'échange de trames périodique de contrôle. Ils déterminent la route grâce à des algorithmes de calcul du plus court chemin. Les principaux protocoles sont :

- OLSR, Optimized Link State Routing : le routage se fait conformément aux spécifications du RFC MANET (Clausen, Jacquet, 2003), qui utilise un routage saut par saut et une technique par état des liens pour déterminer dynamiquement les tables de routage (Clausen, Jacquet et al. 2003).
- FSR, Fisheye State Routing : protocole de type état des liens, chaque nœud diffuse son voisinage local avec une fréquence qui dépend du nombre de sauts qu'un paquet doit effectuer (Xiaoyan, Guangyu, 2001).
- TBRPF, Topology Broadcast Based on Reverse-Path Forwarding : protocole de routage destiné aux réseaux ad hoc mobiles. Il échange régulièrement des informations sur la topologie du réseau afin d'en déterminer les tables de routage <<http://tbrpf.erg.sri.com>>.

3.4 Les protocoles hybrides

Ils fonctionnent en mode proactif pour garder la connaissance locale de la topologie et le mode réactif pour les nœuds lointains. Le principal protocole est ZRP (Zone Routing Protocol) fondé sur la technique vecteur de distance (Haas, Pearlman et al. 2001).

4. La plate-forme MERITE

Nous présentons dans cette partie le projet MERITE (Garda, Romain et al., 2003) ainsi que notre plateforme expérimentale du système d'acquisition et de routage que nous avons utilisé pour tester l'algorithme de routage que nous avons retenu.

4.1 Présentation de MERITE

MERITE (Mobile Embarqué Reconfigurable Intelligent et Télécommunicant) est une plate-forme de prototypage de capteurs intelligents sans fils élaborée au sein du groupe Système Electronique (SYEL) du laboratoire Instruments Système Ile de France (LISIF) pour des applications topologiques de réseaux d'objets communicants. La composition de cette plateforme repose sur l'utilisation conjointe de différentes sortes de capteurs (CCD, magnéto résistifs, ...), d'une unité de traitement et de routage, d'un module de radiocommunication

sans fil utilisant la norme bluetooth et d'une unité de traitement et de routage basée sur un cœur de microprocesseur (IP logicielle).

4.2 Applications de MERITE

L'une des principales applications est la construction d'un réseau d'unités autonomes mobiles capables de se déplacer dans des zones inconnues, inaccessibles, hostiles à l'être humain ou dans des zones à risques (incendie, radiation, tremblement de terre, ...) en vue par exemple d'optimiser l'assistance humaine. On arrive ainsi à fournir des informations sur le terrain afin d'établir une stratégie d'évolution selon le but souhaité. On peut par exemple localiser des victimes lors des opérations de sauvetage. Ceci est possible grâce à de petits mobiles capables de s'infiltrer à travers les décombres ou d'autres, plus étanches pour explorer les fonds aquatiques. Une autre application, et non la moindre, est l'exploitation militaire. Dans ce contexte, l'emploi des réseaux de capteurs peut aller des surveillances de routine des périmètres, jusqu'à assister des attaques aériennes ou terrestres et conduire des opérations d'espionnage. Pour cela, il faut qu'aucun élément ne soit indispensable pour le fonctionnement du réseau. Une telle architecture, dite ad hoc, permet de maintenir le réseau en action suite à la perte d'un ou de plusieurs éléments et nécessite un module de routage.

4.3 L'architecture fonctionnelle de MERITE

L'architecture fonctionnelle de MERITE (figure 6) est composée des unités suivantes (Pinna, Alexandre et al., 2002), (Dumontier, Luthon, 1999), (Lohier, Garda et al., 2000), (Lohier, Lacassagne et al., 2000), (Faura, Garda, 2003) :

- unité de capteur d'acquisition ;
- unité de détection de mouvement ;
- unité de réseau de neurone ;
- unité de compression ;
- unité d'interface radio sans fil.

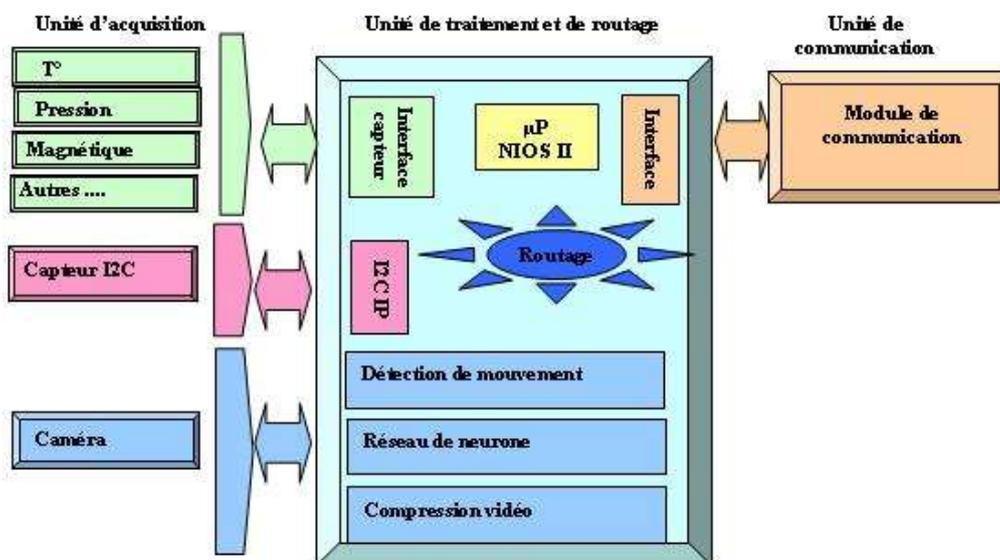


Figure 6. L'architecture de MERITE

4.4 Système d'acquisition et de routage de MERITE

La plateforme expérimentale du système d'acquisition et de routage de MERITE (figure 7), est conçue autour du kit de développement ALTERA <<http://www.altera.com/>> et comprend essentiellement 3 unités à savoir une unité d'acquisition, une unité de routage et une unité d'interface radio.

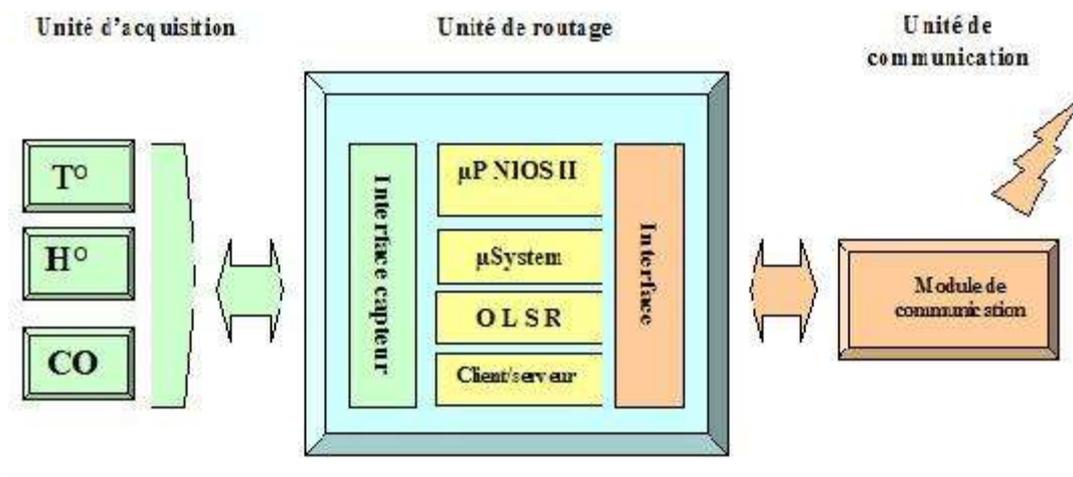


Figure 7. Architecture interne de la balise de la plate forme expérimentale du système d'acquisition et de routage

4.4.1 Unité d'acquisition

Le rôle de l'unité d'acquisition est d'obtenir des mesures numériques des paramètres environnementaux. Pour cela, l'unité d'acquisition est basée dans un premier temps sur l'utilisation de trois capteurs analogiques :

- un capteur de température SMT16030,
- un capteur d'humidité de type résistive,
- un capteur de monoxyde de carbone (CO), HS134.

Pour qu'elles soient traitées, les mesures obtenues par les capteurs sont numérisées par un unique convertisseur analogique numérique de type AD7810.

4.4.2 Unité de routage

Le rôle de l'unité de routage est d'acquérir les informations en provenance de l'unité d'acquisition et de les envoyer à l'unité de radiocommunication afin que l'unité de contrôle puisse les réceptionner correctement par l'intermédiaire d'autres systèmes.

4.4.3 Unité de radiocommunication

Pour réaliser notre réseau de capteurs sans fil, deux normes peuvent être utilisées, soit la norme bluetooth <<http://www.rfsolutions.co.uk>> (IEEE802.15) soit la norme WIFI (IEEE802.11b). Le choix entre ces deux normes dépend essentiellement de plusieurs caractéristiques comme la consommation, la possibilité de réaliser des réseaux ad hoc, la simplicité d'interfaçage avec la plateforme ALTERA.

Par rapport à l'application envisagée, nous avons choisi la norme bluetooth (figure 8), au moyen de module de radiocommunication de type BRM01 de la société RFSOLUTION du fait des avantages qu'il confère : module radio de classe 1, 100mW, 460.8 kb/s, interface de type série, etc. De plus, ces modules peuvent adresser jusqu'à 32 esclaves. Ainsi, des topologies de réseaux comme des « PICONET » ou des « SCATTERNET » pourront être réalisées. Nous pouvons voir sur la figure ci-dessous le module bluetooth utilisé à la fois par le système et par l'unité de contrôle.

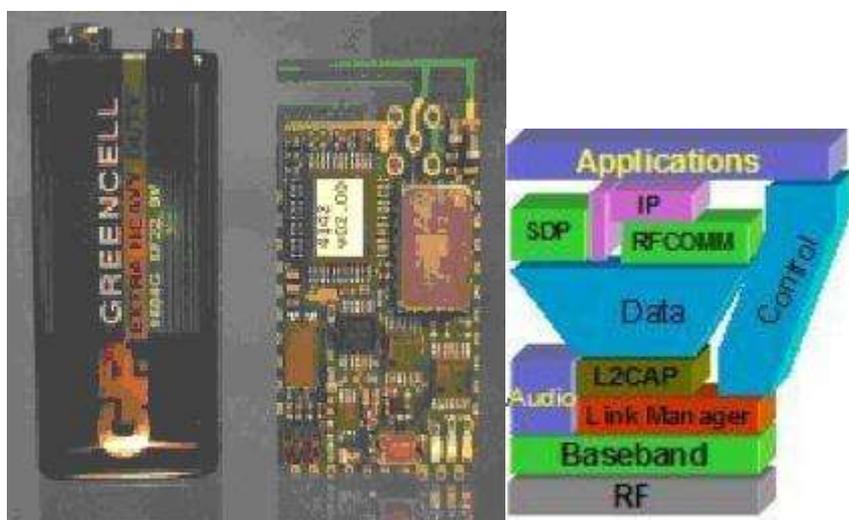


Figure 8. Module de radiocommunication de type BLUETOOTH et sa pile de protocoles

Pour tester l'unité de radiocommunication, nous avons utilisé une unité de contrôle basée sur l'utilisation d'un PC intégrant une interface de radiocommunication identique à celle utilisée dans les balises. La liaison utilisée entre le PC et le module de radiocommunication est de type série (RS232) comme le montre la figure 9.

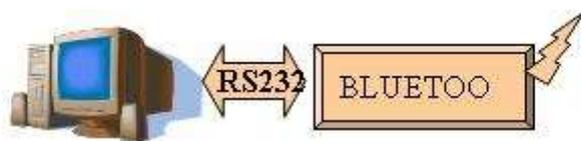


Figure 9. Architecture de l'unité de contrôle

5. Méthodologie de conception

Nous présentons dans cette section la méthodologie de conception retenue. La méthodologie s'appuie sur un partitionnement matériel et logiciel des spécifications de la balise suivant une approche hiérarchique classique de type Top-Down. Pour plus de flexibilité, l'architecture de la balise étant décrite sous forme de blocs fonctionnels, une méthodologie à base d'IP (Intellectual Property) a par conséquent été utilisée.

L'ensemble IPs qui permet de gérer les différentes entrées/sorties de la balise (l'interface capteur de température, l'interface de gestion des convertisseurs analogiques numériques ainsi que les interfaces pour la communication avec le module bluetooth) sont entièrement décrites en VHDL synthétisable et leur gestion est réalisée en C. Une IP softcore de processeur Nios II permet de gérer leur comportement. Un système d'exploitation temps réel de type μ Clinux a été utilisé. La figure 10 résume le flot de conception utilisé.

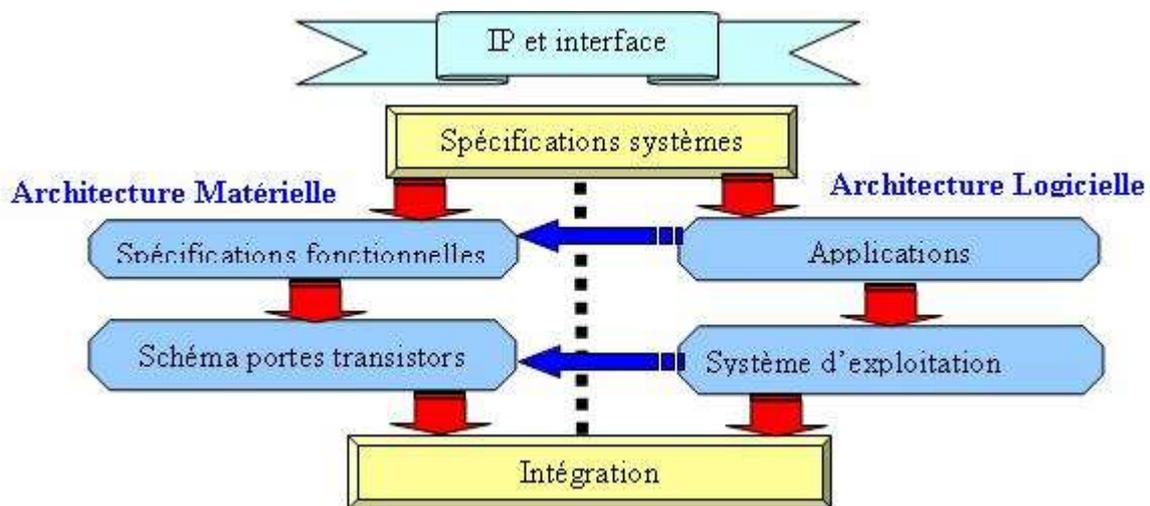


Figure 10. Méthodologie de la conception de l'IP capteur

Pour la conception et la réalisation matérielle et logicielle, nous avons utilisé les outils suivants :

- Quartus,
- SOPC Builder,
- IDE NIOS II,
- Microtronix μ Clinux 2.6.9.

6. Implémentation

Nous avons choisi le protocole proactif l'OLSR. Ses avantages par rapport à d'autres protocoles sont nombreux : la facilité de déployer un réseau en un temps très court et l'encombrement du réseau plus faible. Sa technique de diffusion par des relais multipoints, lui permet d'atteindre toutes les unités dans le réseau avec un nombre réduit de répétition, ceci implique une optimisation, lors des diffusions des messages de contrôle dans le réseau et la consommation de la bande passante. Chaque unité du réseau échange périodiquement des

informations sur la topologie du réseau avec ses voisins et sélectionne de façon optimale ses relais qui seront alors responsables du routage des données. Les réseaux OLSR MERITE sont de ce fait auto-configurables.

6.1 Spécification OLSR

Comme son nom l'indique, OLSR est un protocole à états de liens optimisés ; il détermine aussi des routes de plus court chemin. Alors que dans un protocole à état de lien, chaque noeud déclare ses liens directs avec ses voisins à tout le réseau, dans le cas d'OLSR, les noeuds ne déclarent qu'une sous-partie de leur voisinage grâce à la technique des relais multipoints.

Relais multipoints. Ils consistent essentiellement, en un noeud donné, à ignorer un ensemble de liens et de voisins directs, qui sont redondants pour le calcul des routes de plus court chemin. Plus précisément, dans l'ensemble des voisins d'un noeud, seul un sous-ensemble de ses voisins est considéré comme pertinent. Il est choisi de façon à pouvoir atteindre tout le voisinage à deux sauts (tous les voisins des voisins). Cet ensemble est appelé l'ensemble des relais multipoints. Un algorithme de calcul de relais multipoints est donné dans (Qayyum, Laouiti et al., 2002). Ces relais multipoints sont utilisés pour diminuer le trafic dû à la diffusion des messages de contrôle dans le réseau, et aussi pour diminuer le sous-ensemble des liens diffusés à tout le réseau puisque les routes sont construites à base des relais multipoints.

Diffusion par relais multipoints. La diffusion d'un message, à tout le réseau, par répétition, peut se faire par l'inondation classique utilisant la règle un noeud retransmet un message si et seulement si il ne l'a pas déjà reçu. La diffusion par relais multipoints (Qayyum, Laouiti et al. 2002) diminue le nombre de retransmissions en utilisant la règle suivante. Un noeud retransmet un message si et seulement si :

- il ne l'avait pas déjà reçu,
- il vient de le recevoir d'un noeud dont il est un relais multipoint.

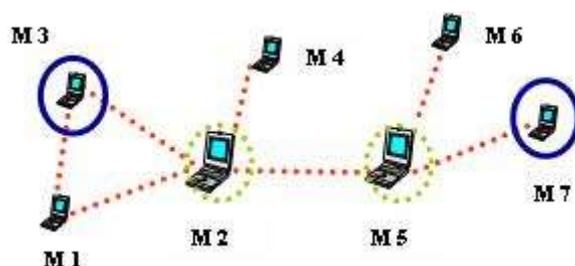


Figure 11. Routage d'un paquet de diffusion en passant par les MPRs

Cette figure illustre un exemple de MPR (Multi-Protocol Router). Le Relais Multipoint M2 est choisi pour être le MPR de M1, M3, M4, et M5. Le Relais Multipoint M5 est lui-même le MPR de M6, M7 et M2. Si M3 souhaite envoyer un paquet de diffusion vers M7, il transfère les paquets à M2, lequel se charge de les envoyer à M5, qui, en dernier ressort, les dépose en M7.

Protocole OLSR : Pour maintenir à jour toutes les informations nécessaires au choix des relais multipoints et le calcul de la table de routage, les noeuds OLSR ont besoin de s'échanger des informations périodiquement. Pour s'informer du proche voisinage, les noeuds OLSR envoient périodiquement des messages dits HELLO contenant la liste de leurs voisins. Ces messages permettent à chacun de choisir son ensemble de relais multipoints. Le deuxième type de messages d'OLSR sont les messages TC (Topology Control).

Un nœud envoie dans un paquet TC l'ensemble des liens l'unissant à ses MPR. Ce paquet TC est envoyé en broadcast et relayé par ses MPR. Ces informations offrent une carte de réseau contenant tous les noeuds et un ensemble partiel des liens suffisant pour la construction de la table de routage.

6.2 Etat de l'art sur ses applications

Le protocole OLSR est un grand succès mondial, il est utilisé dans plusieurs applications. Nous en citons dans cette section quelques unes.

- Défense : OLSR est le protocole de référence DARPA pour les réseaux tactiques, <<http://www.darpa.mil>>
- Automobile: OLSR est considéré pour l'Internet Car (ITS) au Japon (Okada, Wakikawa et al, 2004), <<http://www.sfc.wide.ad.jp/InternetCAR/> >, <<http://www.internetits.org/en/top.html> >
- Réseaux de capteurs : OLSR est utilisé par BAE SYSTEMS Advanced Technology Centre, UK (Dearlove, 2004).
- Réseaux citoyens : OLSR pratiqué dans les réseaux citoyens (Paris, Bruxelles, Lille sans fil, Toulouse, etc.)

6.3 Intégration à MERITE

6.3.1 Introduction

L'implémentation de l'algorithme OLSR à MERITE, fondée sur une plateforme ALTERA Cyclone (System On Programmable Chip), peut se faire de deux manières différentes (figure 12) : directement en logicielle ou mixte (partie logicielle en C et une accélération matérielle en VHDL).

Nous avons choisi après une première étude menée sur des architectures de type PC, d'implémenter une version d'OLSR uniquement logicielle afin de vérifier dans un premier temps la conformité des résultats, puis de comprendre dans un second temps les optimisations à faire afin de répondre au critère de la consommation et la rapidité d'exécution (figure 12).

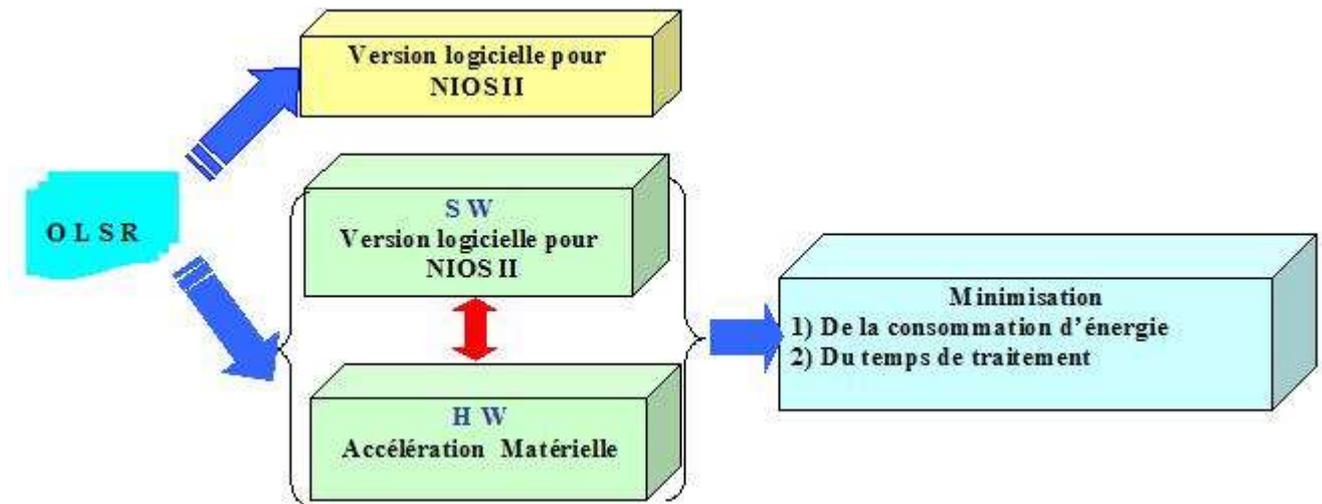


Figure 12. L'implémentation de l'algorithme d'OLSR sur une plate forme de Co-design.

6.3.2 Méthodologie de développement

La méthodologie suivie pour la mise en place de ces différents blocs est décrite par la figure 13.

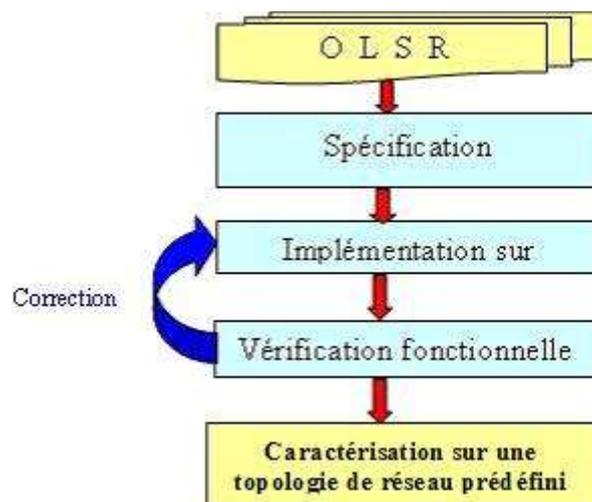


Figure 13. Méthodologie du développement

6.3.3 Conception Matérielle (HW)

OLSR a été porté sur un kit de développement basé sur l'utilisation d'un FPGA de type Cyclone intégrant un processeur logiciel (IP) 32 bits de type Nios II, figure 14.

Ce kit possède l'avantage d'intégrer une interface Ethernet munie d'une API de type TCP/IP, comme l'illustre la figure 14 et la figure 15. Ainsi, un réseau câblé de carte toute pourvue d'OLSR peut être aisément réalisé.

Avant d'intégrer l'OLSR on a veillé à évaluer la taille mémoire consommée (RAM, FLASH) et nous avons pris en compte la faible empreinte de la mémoire.

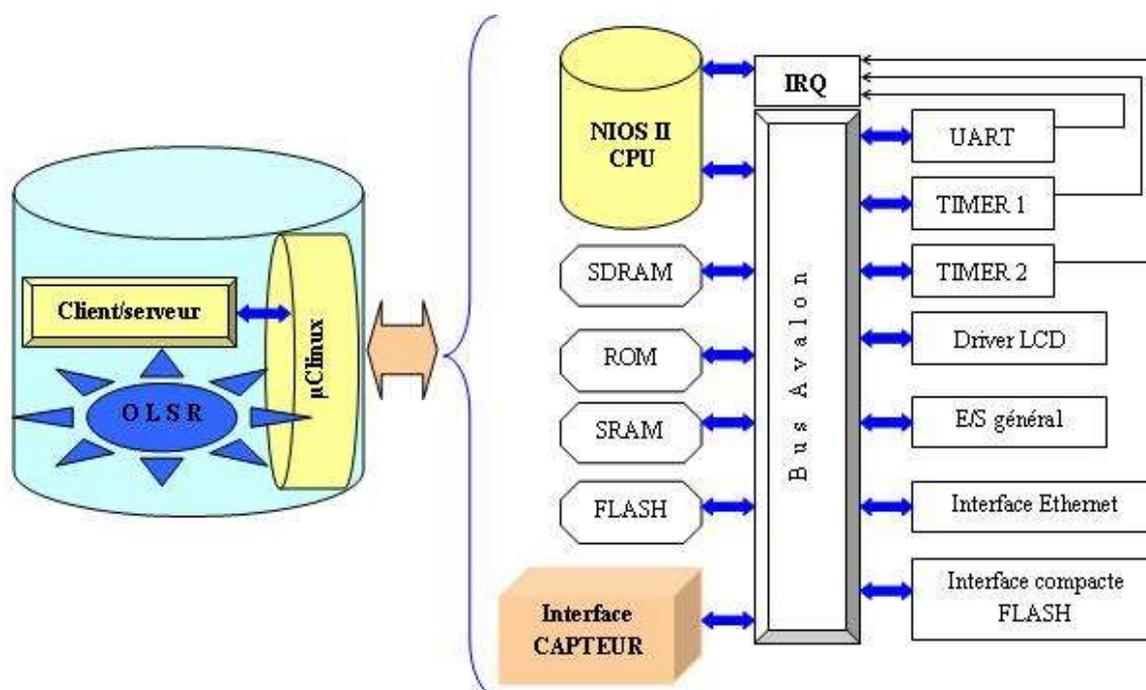


Figure 14. Synoptique du Système SoPC NIOS II pour l'acquisition et le routage de la température

6.3.4 Conception logicielle (SW)

Nous avons utilisé une plateforme NIOS II/Cyclone avec l'environnement Quartus/SOPC builder sur station d'ordinateur muni du système windows. Le logiciel Quartus (ALTERA) nous a permis à la fois de représenter une architecture matérielle et le transfert vers un SOPC.

Le SOPC-Builder intégré dans quartus nous a permis de configurer l'architecture du processeur NIOS II et définir les bibliothèques de programmation nécessaires aux pilotages de ses interfaces.

Le protocole OLSR ne fait pas de routage des données lui-même. Il se contente uniquement de mettre à jour la table de routage de la pile IP d'un noyau du système d'exploitation.

Il existe d'ores et déjà un grand nombre de distributions embarquées basées sur Linux dont la majorité est constituée de projets open source : PeeWee Linux, RTLinux, RTAI, TUXIA, Red Hat Embedded Linux, µClinux, Embedix (Kadionik, 2002), (Kadionik, 2002a), (Lombardo, 2001), (Hollabaugh, 2002).

L'intégration d'OLSR dans la plate forme est finalisée et a été rendue possible en utilisant de surcroît un système d'exploitation de type µClinux. Le gros avantage de µClinux <<http://www.uclinux.org>>; <<http://www.uclinux.com>>, <http://www.microtronix.com/product_linux.htm> par rapport à ses concurrents est la compatibilité de l'API de programmation avec les systèmes Linux standard. Il dispose également de toutes les fonctionnalités réseau TCP/IP, disponibles sur le noyau Linux et supportées par la carte Altéra, figure 15. De plus il est très raisonnable au niveau de l'empreinte mémoire.

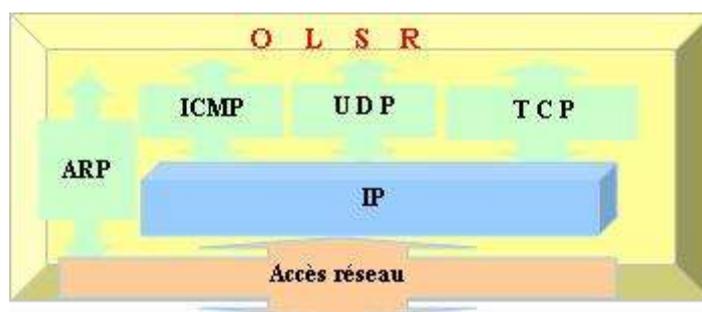


Figure 15. Les fonctionnalités réseau TCP/IP de la carte Altéra utilisées par l'OLSR

7. Résultats expérimentaux

7.1 Architecture d'implémentation

L'architecture adoptée pour l'implémentation est une architecture modulaire. Un premier module gère toutes les tables et s'occupe de l'analyse du contenu des messages reçus ainsi que des différents calculs de routage et d'élection des relais multipoints. Le deuxième module est un module dépendant du système. Cette deuxième partie s'occupe de la communication réseau, de la gestion des interfaces et de l'interaction avec le noyau. La figure 16 représente les modules de routage et de communication ainsi que les sous modules.

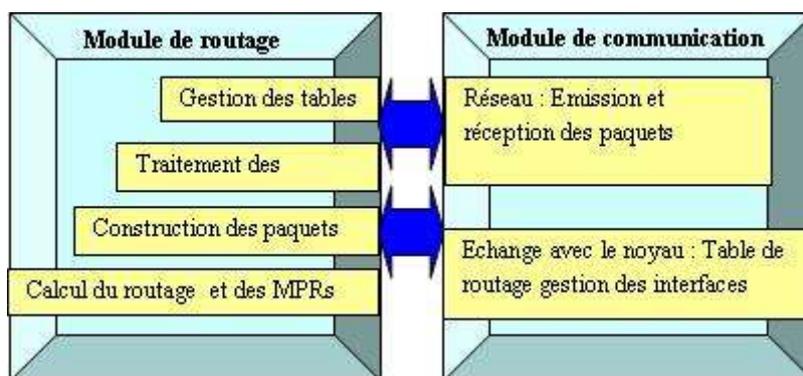


Figure 16. Architecture de l'implémentation

7.2 Topologie de test

Pour vérifier la possibilité de router les données collectées par différents capteurs, nous avons choisi dans un premier temps une topologie réseau de 5 nœuds (contenant l'OLSR), figure 17.

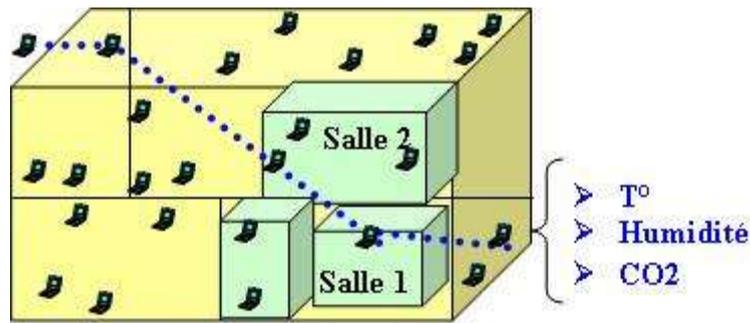


Figure 17. Ttopologie de test

Le protocole a été lancé sur les 2 unités se trouvant aux extrémités de notre réseau. Ces machines n'ont pas pu s'identifier, comme l'illustre la figure 18, car elles n'ont pas de MPR. « mpr is 0 » signifie qu'aucun voisin ni de premier saut, ni de second saut n'est détecté. La table de routage ne contient que l'adresse de notre réseau et de la boucle locale.

```
[root@localhost root]# olsrd -d
mpr is 0
mpr is 0
mpr is 0

[root@localhost root]# route
Table de routage IP du noyau
Destination      Passerelle      Genmask          Indic Metric Ref      Use Iface
192.168.0.0      *               255.255.255.0   U      0      0       0 eth1
127.0.0.0        *               255.0.0.0       U      0      0       0 lo
[root@localhost root]#
```

Figure 18. Résultats de l'exécution de l'OLSR sur 2 unités lointaines

L'exécution de l'OLSR sur les unités qui se trouvent entre ces deux noeuds, les différents MPR, et les voisins du premier et du second saut sont mentionnés. Ces informations, comme le montre la figure 19, changent dès la détection d'un changement dans le réseau.

```
mpr is 4
192.168.0.5
status 3
192.168.0.1=====>Deja un voisin second niveau
192.168.0.3=====>Deja un voisin second niveau
ADDRESS          STATUS
-----
192.168.0.5          3
TWO HOP NEIGHBORS ADDRESS

[root@localhost root]# route
Table de routage IP du noyau
Destination      Passerelle      Genmask          Indic Metric Ref      Use Iface
192.168.0.1      *               255.255.255.255 UH    1     0       0 eth1
192.168.0.3      192.168.0.5    255.255.255.255 UGH   2     0       0 eth1
192.168.0.5      *               255.255.255.255 UH    1     0       0 eth1
192.168.0.0      *               255.255.255.0   U     0     0       0 eth1
127.0.0.0        *               255.0.0.0       U     0     0       0 lo
[root@localhost root]#
```

Figure 19. Résultats de l'exécution de l'OLSR sur les unités se trouvant entre les 2 unités lointaines

8. Conclusion et perspectives

Le problème traité dans cet article est la conception, la spécification et la réalisation d'un système original d'acquisition et de routage des données collectées par des capteurs mobiles. Ce système est basé sur une architecture de type FPGA et doté d'un microprocesseur NIOS II. Nous l'avons intégré dans la plateforme expérimentale MERITE.

Nous avons pu tester cet algorithme dans un réseau de capteurs de topologie variable. Les résultats obtenus sont très satisfaisants et montrent qu'on peut transmettre des informations à des unités lointaines en détournant tous les obstacles possibles, afin de router un message d'un point (capteur intelligent) vers un autre (centrale de traitement).

Toutefois des travaux d'amélioration sont encore nécessaires afin de consolider le présent système. Ils concernent en particulier la minimisation de la consommation d'énergie et la vitesse de traitement et de transmission des données ainsi que la transportabilité du système vers d'autres plateformes.

L'intérêt d'un tel travail est son grand impact pour les applications liées aux réseaux de capteurs sans fils mobiles notamment celles dédiées au domaine militaire.

Bibliographie

Pujolle G., Salvatori O. et Nozick J. (2004). Les réseaux, éditions 2005. Eyrolles.

Anna Hac (2003). Wireless Sensor Network Designs. John Wiley & Sons, ISBN: 0-470-86736-1.

Mühlethaler P. (2002). 802.11 et les réseaux sans fil. Eyrolles.

Perkins C.E., Belding-Rover E.M., Das S. R. (2002). Draft-ietf-manet-aodv-13. <http://www.ietf.org/internet-drafts/draft-ietf-manet-aodv-13.txt>

Johnson D.B., Maltz D.A., Aon N., Hu Y.C., Jetcheva J.G. (). Draft-ietf-manet-dsr-09. <http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-09.txt>

Clausen T., Jacquet P. (2003). Project Hipercom. INRIA Rocquencourt, France. <http://www.ietf.org/internet-drafts/draft-ietf-manet-olsr-11.txt>

Clausen T., Jacquet P., Laouiti A., Minet P., Muhlethale P., Qayyum A., Viennot L. (2003). OLSR RFC3626. experimental edition.

Xiaoyan H., Guangyu P. (2001). Fisheye State Routing Protocol (FSR) for Ad Hoc Networks. Rockwell Scientific Company. http://www.geocities.com/ray_gng/draft-ietf-manet-fsr-02.txt

Haas Z. J., Pearlman M. R., Samar P., (2001). Zone Routing Protocol (ZRP). Internet Draft, Draft-ietf-manet-zrp-04.txt, Travail en cours.

Garda P., Romain O., Granado B., Pinna A., Faura D., Hachicha K. (2003) Architecture of an intelligent BEACON for wireless sensor network. Proceedings IEEE NNSP2033, Toulouse : 17-19 Septembre.

Pinna A., Alexandre A., Belhaire E., Garda P., Granado B. (2002) Design of an Integrated Silicon Connectionist retina, Proceedings of the 15th IEEE ASIC/SOC2002 Conference, Rochester - New York : 25-28 Septembre.

Dumontier, Luthon F., (1999). Real-time DSP Implementation for MRF-Based Video Motion Detection, IEEE Transactions on Image Processing, Vol. 8, Num. 10, Octobre. 1999, 1341-1347.

Lohier F., Garda P., Lacassagne L., (2000). Procédé et dispositif de traitement de séquences d'images avec masquage, National Patent N° FR 62060 L, France : 3 February 2000, International extension pending.

Lohier F., Lacassagne L., Garda P., (2000). Masked-Motion-JPEG2000 : a new reduced-complexity video sequence compression scheme based on a MRF-motion detection algorithm towards inter-frame masking, ICSPAT'2000, Int. Conf. On Signal Processing Applications and Technology, Dallas : 16-19 Octobre.

Faura D., Garda P., (2003). Segmentation d'images couleur pour la compression de séquences vidéo par l'algorithme Mask Motion JPEG2000, journées CORESA'03, Lyon : Janvier 2003.

Qayyum A., Laouiti A., Viennot L., (2002). Multipoint relaying technique for flooding broadcast messages in mobile wireless networks, Proceedings Hawaii International Conference on System Sciences HICSS, Hawaii : Janvier.

Okada K., Wakikawa R., Uehara K., Murai J. (2004). OLSR for InternetCar System", OLSR Interop and Workshop 2004, San Diego : 6-7 Août.

Dearlove C. (2004). OLSR Simulation, Implementation and Ad Hoc Sensor Network Application, OLSR Interop & Workshop, San Diego.

Kadionik P. (2002). Le projet µClinux. Linux Magazine. Février.

Kadionik P. (2002a). Administrez facilement votre réseau. Linux Magazine. Octobre.

Lombardo J. (2001). Embedded Linux. Editions New Riders.

Hollabaugh C. (2002). Embedded Linux. Editions Addison Wesley.

Pour citer cet article

Ahmed Habbani, Olivier Romain, Jamal El Abbadi et Patrick Garda. «Réseaux de capteurs : Système d'acquisition et de routage d'information». e-TI, Numéro 2 / Issue 2, 15 avril 2006, <http://www.revue-eti.net/document.php?id=828>.