



---

**TELESURVEILLANCE DES PANNEAUX D’AFFICHAGE A TRAVERS UN  
RESEAU WIFI.**

**RONAL HUMBERTO NOCUA CIFUENTES**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER  
ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA Y  
TELECOMUNICACIONES  
BUCARAMANGA  
2006**



ÉCOLE SUPÉRIEURE CHIMIE PHYSIQUE  
ELECTRONIQUE DE LYON



Projet de fin d'études

**TELESURVEILLANCE DES PANNEAUX D’AFFICHAGE A TRAVERS UN  
RESEAU WIFI.  
ETI3**

**Présenté par : M. Ronald NOCUA.**

**Directeur de projet : M. Richard DUTRULLE.**

**Tuteur Ecole : M. Fabrice JUMEL.**



**Université Industriel de Santander  
Colombie**

## TABLE DES MATIERES

<b>1</b>	<b><u>CONTEXTE DU PROJET</u></b>	<b>12</b>
1.1	<b>LA SOCIETE EMBEDIA™ ET SES OBJECTIFS.</b>	<b>12</b>
1.1.1	DOMAINE DE L'ENTREPRISE	13
1.1.2	PARTENAIRES DU PROJET	13
1.1.3	MARCHE	14
1.2	<b>PROBLEMATIQUES A RESOUDRE</b>	<b>15</b>
1.3	<b>LE SYSTEME EMBARQUE EMBEDIA</b>	<b>16</b>
1.3.1	COMMUNICATION LOCALE	17
1.3.2	COMMUNICATION A DISTANCE	17
1.3.3	GESTION TELEMANTENANCE	18
1.4	<b>ETAT DU DEVELOPPEMENT</b>	<b>18</b>
1.4.1	COMMUNICATION LOCALE	18
1.4.2	COMMUNICATION A DISTANCE	19
1.4.3	GESTION TELEMANTENANCE	19
1.5	<b>CAHIER DES CHARGES</b>	<b>19</b>
1.5.1	INTERACTIVITE	19
1.5.2	GESTION DE TELEMANTENANCE	19
<b>2</b>	<b><u>DEVELOPPEMENT DU PROJET</u></b>	<b>24</b>
2.1	<b>PROCEDURE DE TRAVAIL</b>	<b>24</b>
2.1.1	CYCLE EN V	24
2.1.2	CYCLE EN V A EMBEDIA	25
2.1.3	PARTAGE DE RESSOURCES	26
2.2	<b>TRAVAUX REALISES</b>	<b>26</b>

---

2.2.1	CONTRAINTE DU DEVELOPPEMENT .....	27
2.2.2	PORTAGE DE CODE SUR LA CARTE ADVANTECH.....	28
2.2.3	GESTION DE TELEMANTENANCE.....	29
2.2.4	TEST DE VALIDATION DE L'APPLICATION D'INTERACTIVITE .....	47
2.2.5	INDUSTRIALISATION DU PRODUIT .....	50
<b>2.3</b>	<b>ÉTAT DU PRODUIT .....</b>	<b>50</b>
2.3.1	COMMUNICATION LOCALE.....	51
2.3.2	COMMUNICATION A DISTANCE .....	51
2.3.3	GESTION DE TELEMANTENANCE .....	52
<b>3</b>	<b><u>SYNTHESE ET CONCLUSIONS .....</u></b>	<b><u>53</u></b>
	<b><u>ANNEXES .....</u></b>	<b><u>56</u></b>
	<b><u>ANNEXE I.....</u></b>	<b><u>57</u></b>
	<b><u>ANNEXE II.....</u></b>	<b><u>59</u></b>

## TABLE DES FIGURES

FIGURE 1: SYSTEME EMBEDIA	17
FIGURE 2: SOLUTION EMBEDIA	21
FIGURE 3 : CYCLE EN V	25
FIGURE 4: TRAVAUX REALISES	27
FIGURE 5 : GESTION ALARMES	33
<i>FIGURE 6 : GESTION SCRUTATION</i>	37
<i>FIGURE 7 : GESTION « SCRUTATION- CONFIGURATION »</i>	40
FIGURE 8: GESTION DES ALARMES	45
FIGURE 9 : SCRUTATION	46
FIGURE 10: SCRUTATION-CONFIGURATION	47



## REMERCIEMENTS

J'aimerais remercier les personnes et organismes qui ont contribué, de près ou de loin, au travail que je vous présente aujourd'hui :

À Richard Dutrulle, pour son encadrement, l'intérêt qu'il m'a apporté ainsi que sa sagesse et les très bons moments partagés.

A Frederic Chazelle et Atmin Ait-Hamouda pour leur collaboration constante pendant mon séjour à l'entreprise.

Aux institutions qui ont rendu possible ce projet : la Société Embedia, l'école d'ingénieurs CPE de Lyon et à l'Université Industriel de Santander en Colombie.

Aux collègues de Embedia, Youssef, Mingh et Damien, pour leur gentillesse et leur appui dans le développement du projet.

Merci à tous ceux qui ont apporté un grain de sable à la réalisation de ce projet et à la construction d'une ambiance de travail très agréable

## RESUMEN

**TITULO:**

GESTION DE LA SUPERVISION DE PANELES PUBLICITARIOS A TRAVES DE UNA RED WIFI\*

**AUTOR**

RONALD HUMBERTO NOCUA CIFUENTES

**PALABRAS CLAVES**

Sistema embebido, tiempo real, thread

**DESCRIPCIÓN**

El proyecto desarrollado presenta una solución que permite la gestión de la supervisión a distancia de los paneles publicitarios a través de una red Wifi. Cada panel ha sido dotado de un sistema embebido, el cual es el encargado de albergar la aplicación realizada. Este sistema realiza asimismo el establecimiento de la red ad-hoc Wifi, sobre la cual va a ser enviada la información hasta el terminal supervisor.

A partir de los requerimientos impuestos por la compañía explotadora de los paneles publicitarios, se realizó la fase de especificación, en la cual se definieron los objetivos generales así como los resultados que se esperaban obtener con la aplicación. A continuación se procedió a la concepción de la aplicación. Esta concepción definió las tres subaplicaciones principales, las cuales son: la *gestión de alarmas*, que permite verificar el funcionamiento de las principales variables susceptibles de fallar; la *escrutación*, la cual brinda al terminal supervisor la posibilidad de conectarse al módulo (sistema embebido) para determinar su funcionamiento; y la *configuración de parámetros*, que permite al puesto remoto cambiar los parámetros de funcionamiento del autómata que administra la actividad del panel.

Tanto el sistema embebido como las aplicaciones fueron desarrolladas sobre el entorno Win CE. Este entorno permite la construcción del sistema embebido a bajo nivel (controladores y base de registros), así como a nivel aplicativo a través de "Embedded Visual C++", sobre el cual se llevó a cabo la mayor parte del desarrollo.

El proceso general desarrollado fue testado sobre un panel real, permitiendo la verificación de los parámetros de funcionamiento del mismo en tiempo real, con el envío de la información hasta el terminal supervisor. Las otras dos subaplicaciones también fueron testadas con el fin de asegurar el correcto desempeño de la aplicación.

---

\*Trabajo de Grado

---

## ABSTRACT

### TITLE:

MANAGEMENT OF THE ADVERTISING-PANELS SUPERVISION THROUGH A WIFI NETWORK\*

### AUTHOR

RONAL HUMBERTO NOCUA CIFUENTES

### KEYWORDS

Embedded system, real time, thread

### DESCRIPTION OR CONTENTS

In this project, a solution to allow the remote management of the advertising-panels supervision through a Wifi network is presented. Each panel has been provided with an embedded system, in which the realized application is lodged. This system also effects the establishment of the ad-hoc Wifi network, on which the information will be send to the supervisor terminal.

Based on the requirements imposed by the advertising-panels exploiting company, the specification phase was realized, in which the general objectives, as well as the expected results, were defined. Then, the application was conceived. This conception defined the three main sub-applications: *alarms management*, which allows the verification of the functioning of the principal variables susceptible of failing; *the polling*, which gives the terminal supervisor the possibility of connecting to the module (embedded system) in order to determine its functioning; and *the parameters configuration*, which lets the remote terminal to change the functioning parameters of the automate who controls the panel activity.

The embedded system, as well as the applications, were developed on the Win CE environment. This environment allows the construction of the embedded system at low level (drivers and registers database) and also the applicative level through the "Embedded Visual C++", on which the most of the development was carried on.

The general developed process was tested on a real panel, letting the verification of its functioning parameters in real time, with the sending of the information to the supervisor terminal. The others two applications were also tested with the purpose of assuring the correct performance of the application.

---

\*Trabajo de Grado

---

## RESUME

### **TITULO :**

TELESURVEILLANCE DES PANNEAUX D’AFFICHAGE A TRAVERS UN RESEAU WIFI<sup>3</sup>

### **AUTOR**

RONALD HUMBERTO NOCUA CIFUENTES

### **PALABRAS CLAVES**

Systeme embarqué, temps réel, thread, multi-taches.

### **DESCRIPCION**

Le projet développé présent une solution qui permet la gestion de la surveillance à distance des panneaux d’affichage sur un réseau Wifi. Chaque panneau a été équipé d’un système embarqué, lequel héberge l’application réalisée. Le système effectue aussi l’établissement du réseau ad-hoc Wifi, sur lequel va être envoyée l’information jusqu’au terminal superviseur.

En partant des besoins imposés pour la société qui exploite les panneaux d’affichage, on a réalisé la phase de spécification, dans laquelle les objectifs principaux et les résultats attendus pour l’application ont été définis. En suite, on a effectué la conception de l’application. Cette conception a défini les trois sub-applications principales, lesquelles sont: la gestion des alarmes, qui permet vérifier le fonctionnement de l’activité du panneau ; la scrutation, laquelle fournit au terminal superviseur la possibilité de se connecter au module (système embarqué) de façon de déterminer le fonctionnement ; et la configuration des paramètres, qui permet au poste distante changer les paramètres de fonctionnement de l’automate qui gère l’activité du panneau.

Le système embarqué et les applications ont été développées sur l’environnement Win CE. Cet environnement permet la construction du système embarqué à bas niveau (pilotes et base de registres), ainsi que niveau applicatif à travers de “Embedded Visual C++”, sur lequel a été développé la plupart du projet.

L’application générale développée a été testée sur un panneau réel, en permettant la vérification des paramètres du fonctionnement du panneau en temps réel, avec l’envoi de l’information jusqu’au terminal superviseur. Les autres deux sub-applications ont été testées en assurant le bon fonctionnement de l’application.

---

<sup>3</sup> Project de fin d’études

## Introduction

J'ai réalisé mon stage dans une entreprise qui travaille dans le domaine des systèmes embarqués. La société propose une solution innovante qui transforme toute source d'information, particulièrement les panneaux d'affichage publicitaire, en objet communicant. Le but est d'envoyer des informations complémentaires à celles visualisées par le public, relatives à un produit afin de les rendre plus attractives et augmenter leur impact auprès des consommateurs. Un module électronique embarqué sur chaque panneau permet l'envoi des informations mais fournit aussi la possibilité de relier l'ensemble des panneaux d'affichage dans un réseau étendu hybride (Wifi, GPRS..) pour permettre la télésurveillance et la supervision à distance.

Dans la première partie du rapport, je décris un peu plus l'entreprise et le produit qui est développé, ainsi que le cahier des charges proposé pour la gestion de la télémaintenance.

La deuxième partie décrit la procédure de travail suivie pour développer les fonctionnalités et les diverses tâches réalisées au cours de mon stage. Dans cette partie sont contenues l'analyse et la conception relatives au cahier des charges pour réaliser la gestion de la télémaintenance. A la fin de ce chapitre sont décrites les évolutions du projet compte tenu du fait que j'ai travaillé sur plusieurs domaines.

La dernière partie est dédiée à la synthèse et à la conclusion sur mon séjour au sein de l'entreprise ainsi qu'aux travaux réalisés.

## 1 Contexte du projet

### 1.1 La société EMBEDIA™ et ses objectifs.

Les technologies issues du monde « télécom » offrent de nombreuses possibilités pour améliorer la communication vers le public et leur utilisation conduit à l'émergence de nouveaux secteurs. Les créateurs d'EMBEDIA ont décidé de combiner leurs ressources pour initialiser un nouveau projet : EMBEDIA™, "Média Embarqué" (Embedded Media). Pour eux, la communication interactive avec le public est l'avenir de l'information : impact temps réel, information instantanée, valeur ajoutée au message initial, etc. EMBEDIA™ est une SAS créée au 1<sup>er</sup> Avril 2005 au capital de 50 000 Euros.

Dans ce contexte, leur attention se porte rapidement sur l'affichage publicitaire car il présente des analogies fortes avec les réseaux de publiphones bien connus par les fondateurs : situés de façon stratégique au cœur des villes et des lieux de passage, les panneaux d'affichages s'adressent au public en réseaux. Les problématiques des Afficheurs sont proches de celles du domaine des télécommunications : avoir des moyens de traitement et de communication embarqués pour optimiser l'exploitation des panneaux –télémaintenance, statistiques de fonctionnement, nouveaux services pour l'utilisateur, etc.

Un exploitant et un constructeur de panneaux ont déjà fait confiance à EMBEDIA™ qui propose ses systèmes à d'autres secteurs : Grande Distribution, Distribution Spécialisée, Organisation d'événementiel, Universités, Stades de sports, Contrôle d'accès, Maintenance Industrielle.

L'objectif du projet EMBEDIA est de « rendre interactifs et communicants » tous les supports d'information situés stratégiquement à proximité du public. Ceci leur permettrait de communiquer des informations supplémentaires ou

complémentaires par l'intermédiaire des appareils mobiles personnels usuels (téléphones mobiles, PDAs).

### **1.1.1 Domaine de l'entreprise**

Le domaine de l'entreprise est orienté vers le développement des solutions pour des systèmes embarqués. Le produit principal développé par la société comprend les principales technologies de communication sans fil de courte et moyenne distance (Bluetooth et Wifi).

### **1.1.2 Partenaires du projet**

Parmi les partenaires technologiques et industriels, EMBEDIA™ compte sur le support de : l'Incubateur Créalys, le laboratoire du Citi (Centre d'Innovation de Télécom et Intégration de services), l'Insavalor, l'Inria et Adeneo (fusion entre Adetel et Adeset).

Les partenaires commerciaux incluent un exploitant et un constructeur de panneaux, lesquels sont impliqués dans les premiers tests de l'innovation sur le marché de l'affichage extérieur. EMBEDIA™ a livré ses premiers systèmes interactifs en Août 2005.

EMBEDIA™ compte aussi sur l'accompagnement juridique, financier et institutionnel de plusieurs entités régionales.

### 1.1.3 Marché

En 2002, le montant total des dépenses de publicité en France s'élevait à 30 milliards d'euros : un tiers pour les médias de masse (TV, presse, affichage etc.) et le reste en « hors-média » (marketing direct, mailing, événement, PLV etc.). Hormis internet et le télé-achat, le public n'a pas les moyens d'interagir avec ces médias. En utilisant la technologie Bluetooth (communication sans fil de proximité entre deux équipements / « Machine To Machine »), EMBEDIA exploite la possibilité de transformer toute source d'information de masse en source d'informations interactives.

Le marché "Machine To Machine" est en pleine explosion. Selon le groupement d'intérêts Bluetooth (SIG), trois millions d'équipements Bluetooth sont vendus dans le monde chaque semaine ([10],[11]) En 2008, le seul marché européen comptera près de 286 millions d'équipements Bluetooth (téléphones mobiles, PDAs, ordinateurs portables) (Forrester 2004) et selon, 60% des téléphones portables vendus en Europe en 2006 intégreront en standard cette technologie qui équipe aujourd'hui 13% du parc mobile européen ([12]). Equiper les supports d'information publicitaires (tels que les panneaux d'affichage, écrans plasma, arrêts de bus, vitrines de magasin etc.) avec la technologie embarquée d'EMBEDIA fera émerger un nouveau marché. Ces nouvelles sources d'informations interactives communiqueront sur demande avec des équipements Bluetooth (et autres standards) situés à proximité, permettant ainsi aux utilisateurs d'interagir avec les informations et de les télécharger directement sur leurs appareils mobiles.

Avec son système interactif innovant, EMBEDIA cible les segments de marché :

- Publicité extérieure, indoor, transports,

- Promotion sur lieux de ventes,
- Événementiel.

En France, le parc cible de l'affichage extérieur est estimé à 20 000 *panneaux* existants. L'affichage publicitaire extérieur recherche activement de nouvelles solutions pour réagir à la réduction constante du nombre de sites d'affichage (lobbies écologiques, pressions municipales) en Europe de l'Ouest. Sur ce seul marché, le nombre estimé de panneaux d'affichage extérieurs représente 100 000 unités.

### **1.2 Problématiques à résoudre**

- **Les individus** (utilisateurs, consommateurs) aimeraient pouvoir exploiter les informations qui leur sont délivrées dans leurs déplacements quotidiens depuis les supports d'affichage extérieurs. Parfois ils n'ont pas le temps ou les moyens de les noter et les oublient. Ils ne peuvent pas non plus interagir avec celles-ci (consulter, réserver, participer) dans l'instant, pendant les moments d'attente, de transit (shopping, transport, domicile-travail), etc.
- **Les entreprises, les annonceurs** (industries, distributeurs, collectivités locales...) qui communiquent sur leurs images, leurs produits, leurs promotions via les campagnes publicitaires ou d'informations (comme par exemple sur les panneaux d'affichage) recherchent de plus en plus à personnaliser leurs messages, préserver et renforcer leurs présences, et étendre le dialogue avec leurs clients.
- **Les Opérateurs d'espaces publicitaires** cherchent à améliorer et moderniser leurs médias pour diffuser les campagnes publicitaires en créant

une nouvelle valeur ajoutée à leurs clients annonceurs (personnalisation), au public (meilleure information, besoin impulsif d'aller plus loin) et à eux-mêmes (différenciation, exploitation, services).

- **Entreprises Annonceurs** souhaitent réaliser la télémaintenance et le télépilotage à distance, puisque jusqu'à maintenant ces opérations sont réalisées de façon locale.

### **1.3 Le système embarqué Embedia**

EMBEDIA propose une solution embarquée communicante et interactive (brevet déposé) qui transforme toute source d'information traditionnelle (panneaux d'affichage, écrans électroniques, vitrines, panneaux d'information/signalisation, arrêts de bus etc.) en un média interactif (Fig.1) capable de distribuer, à la demande du public, des informations complémentaires (texte, image, son, vidéo), vers les équipements mobiles (téléphone mobile, PDA) et de communiquer avec les autres panneaux en réseau .

Dans le cadre de son application au secteur de l’Affichage Publicitaire Extérieur, la solution d’EMBEDIA présente de nombreux avantages pour :

- **Le public utilisateur** : Connexion sur demande et gratuite aux supports d'information ; capture et mémorisation instantanées de l'information (aide-mémoire mobile).
- **Entreprises Annonceurs** : Visibilité augmentée (répétition) ; impact personnalisé ; analyse d'audience ; augmentation de l'incitation à l'achat.
- **Afficheurs / Opérateurs d'espaces** : Intelligence embarquée dans le support d'information ; gestion à distance via un réseau adhoc ; nouveaux services et aides à l'exploitation.



Figure 1: Système Embedia

### 1.3.1 Communication locale

La communication locale permet l'envoi d'information vers les téléphones portables depuis le module qui utilise la technologie Bluetooth.

### 1.3.2 Communication à distance

Le réseau Wifi/GPRS permet de relier les panneaux d'affichage dans un réseau étendu pour surveiller leur fonctionnement.

### 1.3.3 Gestion Télémaintenance

La gestion de télémaintenance va permettre de réaliser l'enquête sur l'état de fonctionnement de l'automate qui gère l'activité du panneau.

## 1.4 *Etat du développement*

### 1.4.1 Communication locale

Au début de mon stage, le développement du produit était dans sa phase initiale. Une première version de l'application d'interactivité avait été développée sur une carte d'évaluation de base, **ADENEO DK**, contenant un processeur ARM9. Cette carte a permis de réaliser la première version du produit concernant l'interactivité et d'effectuer les premiers essais.

L'environnement de programmation de l'application est Windows CE avec Embedded Visual C++. Cet environnement fournit la possibilité de construire la plateforme du système embarqué, à l'aide de l'outil Platform Builder, en ajoutant les divers composants qui font partie de la carte ( Périphériques, mémoires, ..). Win CE réalise la construction au niveau des pilotes pour fournir une base qui servira au développement de l'application.

La première version du produit établit la connexion Bluetooth avec un terminal mobile et après la cible effectuée l'envoi d'un fichier vers le téléphone portable utilisant cette technologie.

## **1.4.2 Communication à distance**

Le développement des fonctions de base de cette partie a été réalisé principalement par le CITI. Les premières études ont été orientées sur le choix des antennes qui vont permettre la communication sur Wifi et sur le portage de l'application « réseau maillé hybride » sur Windows CE. Les premiers tests sur la porteuse ont été effectués pour réaliser le choix des antennes.

## **1.4.3 Gestion Télémaintenance**

Le développement de la gestion de la télémaintenance n'avait pas commencé. Les premières démarches sur ce sujet ont commencé avec ma participation spécifique.

# **1.5 Cahier des charges**

## **1.5.1 Interactivité**

En ce qui concerne l'interactivité, il faut effectuer le changement de la carte d'évaluation par une carte industrielle du marché. Le développement doit être effectué sur la carte qui va être commercialisée afin d'assurer un possible processus d'industrialisation.

## **1.5.2 Gestion de télémaintenance**

### **1.5.2.1 Présentation de l'existante**

La société partenaire qui fabrique les panneaux de publicité extérieure travaille avec un automate qui réalise le contrôle des capteurs permettant le fonctionnement des panneaux déroulants. Un module GSM est ajouté pour effectuer la communication vers le superviseur central et l'envoi d'un SMS pour la remontée des alarmes.

Lorsqu'une alarme est déclenchée, l'automate envoie cette information vers le module GSM. Ce module attend la notification et envoie un SMS vers la personne compétente selon le défaut (Annexe I).

#### **1.5.2.2 Solution Embedia télémaintenance**

Embedia propose la création d'un module qui réalisera la gestion et supervision des panneaux d'affichage. Le module Embedia remplacera le module GSM et la remontée des alarmes se fera à travers le réseau Wifi qui sera établi entre les modules et le superviseur.

Dans la fig.2, la solution proposée pour établir la surveillance à distance est décrite.

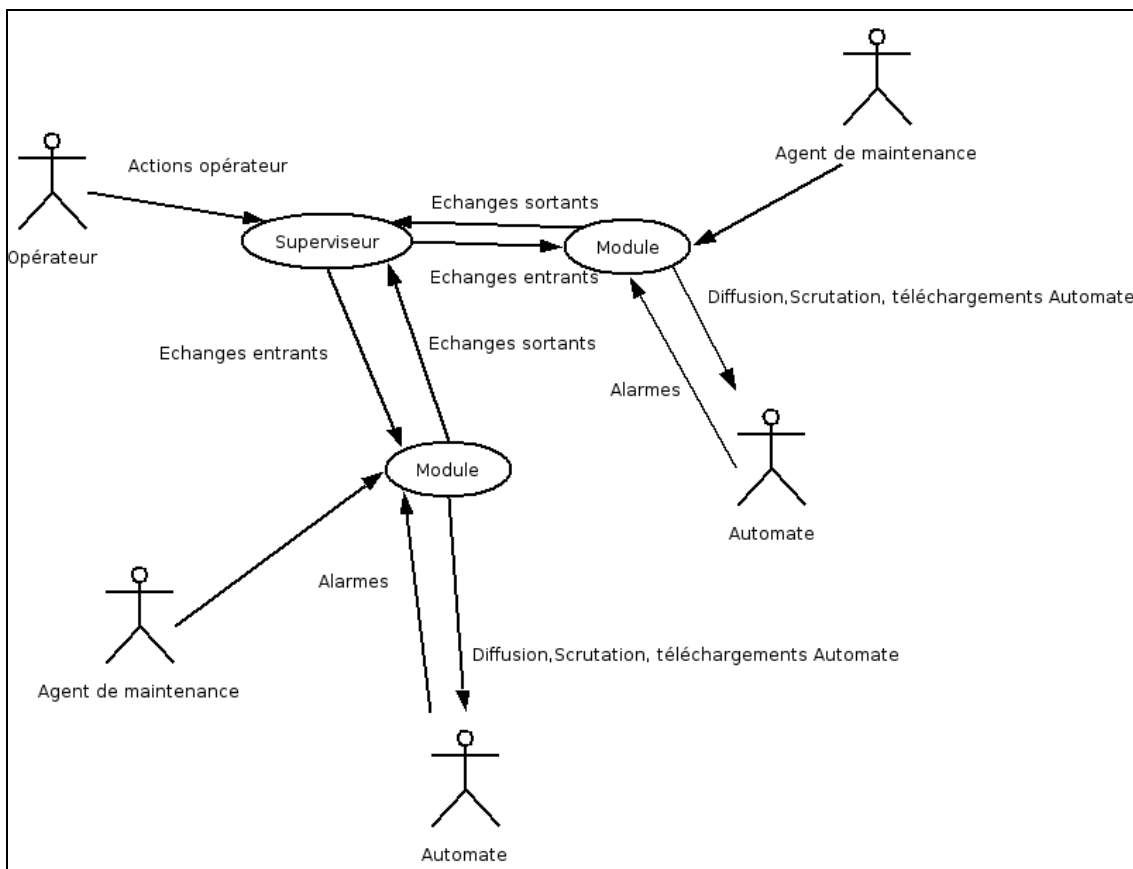


Figure 2: Solution Embedia

### 1.5.2.2.1 Gestion des Alarmes

Le processus qui va être développé doit permettre de déterminer l'état des alarmes et d'envoyer cette information au superviseur. Le tableau 1 décrit les alarmes qui peuvent se présenter sur le panneau. En régime permanent, une alarme indique un dysfonctionnement.

Nom	Description
Alimentation	Absence d'alimentation électrique
Moteur	Le moteur est bloqué
Porte	La porte est ouverte
Chauffage	Le chauffage ne fonctionne pas
Éclairage	L'éclairage ne fonctionne pas

Tableau 1: Alarmes de l'automate

Lorsqu'une alarme est détectée, le processus doit remonter le plus rapidement possible l'alarme au superviseur via une connexion TCP/IP.

Pour effectuer l'envoi des alarmes au superviseur, le module réalisera l'établissement de la connexion au niveau TCP/IP sur un « user port » qui sera accordé entre les deux entités de façon à sécuriser les données échangées.

Il faudra aussi établir le protocole de communication qui sera utilisé pour l'envoi de l'information. L'entête du paquet, la longueur et les codes qui vont permettre un premier niveau de cryptage de l'information.

La gestion des alarmes aura sans doute priorité sur les autres processus, l'envoi sera fait instantanément sans aucun délai.

#### **1.5.2.2.2 Scrutation**

La scrutation est un processus qui permet au superviseur d'atteindre le module via

une connexion TCP/ IP et de connaître l'état de fonctionnement de l'automate ainsi que des paramètres d'exploitation du module.

La scrutation sera définie comme un deuxième type de connexion qui sera établie entre le superviseur et le module. Il faudra utiliser un autre port pour la communication descendante (du superviseur vers module).

Le protocole défini doit permettre de différencier les connexions entrantes et sortantes de façon à sécuriser la connexion entre les deux parties.

#### ***1.5.2.2.3 Configuration des paramètres de l'automate***

L'automate qui contrôle le processus sur chaque panneau permet de fixer les différents paramètres à partir d'une interface série. Depuis le superviseur l'opérateur doit être capable de réaliser le changement des paramètres sur chaque panneau à distance.

Lorsque le superviseur souhaite connaître les paramètres qui sont fixés sur un panneau spécifique, il doit effectuer une demande de « scrutation - configuration » en établissant une connexion TCP/IP vers le module avec l'entête de données spécifique.

Le code qui identifie la demande d'une scrutation - configuration doit être défini entre les parties de la même façon que précédemment.

Une fois que le module a identifié le type de scrutation - configuration, il interroge l'automate sur l'état actuel de sa configuration. Cette information est envoyée vers le superviseur après avoir effectué la construction du paquet avec les entêtes définies pour les connexions sortantes.

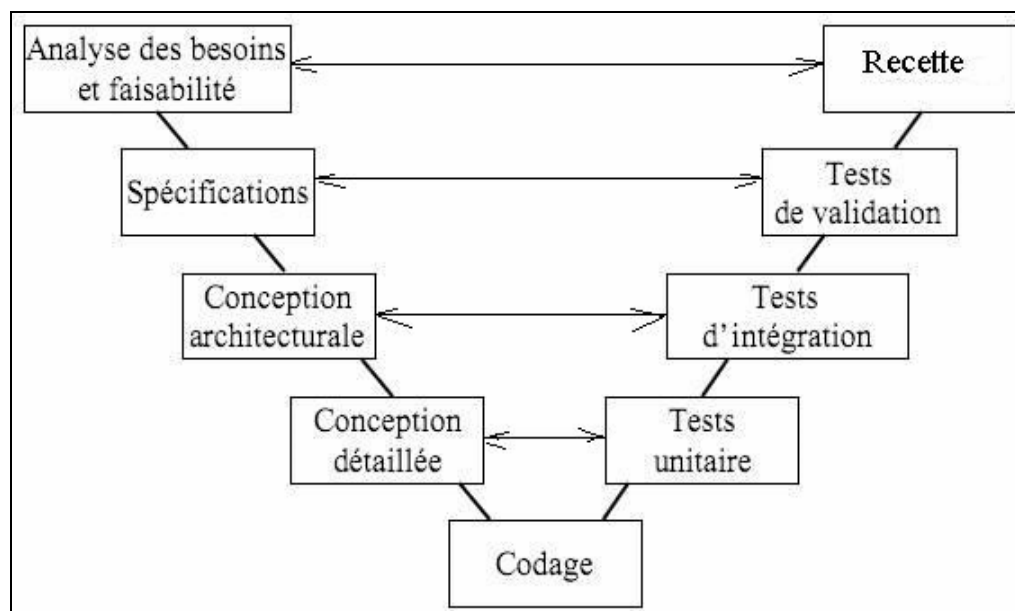
## **2 Développement du projet**

### ***2.1 Procédure de travail***

La procédure de travail au sein de la société suit le standard de développement du cycle en V.

#### **2.1.1 Cycle en V**

Le cycle en V est un standard de l'industrie du logiciel qui permet en cas d'anomalie, de limiter un retour aux étapes précédentes. Le cycle en V consiste à associer à chaque étape de conception une autre étape de test. Les phases de la partie montante, doivent renvoyer de l'information sur les phases en vis-à-vis lorsque des défauts sont détectés afin d'améliorer le logiciel.



**Figure 3 : Cycle en V**

### 2.1.2 Cycle en V à Embedia

Dans le développement réalisé par l'équipe de la société, on a essayé de reprendre la théorie du cycle en V et de l'appliquer lors de la réalisation des travaux. Lors du développement des applications, l'équipe a défini le cycle en V de façon particulière.

La première phase du développement, la phase de spécification, définit les besoins qui veulent être résolus pour la application de façon détaillée.

La deuxième partie, la phase de conception, dans laquelle on décrit comment on va résoudre le problème, permet de prendre en compte le comment.

La phase suivant, la phase du développement, dans laquelle on réalise le codage de l'application. Lorsque cette phase est terminée, on procède à la réalisation des

tests unitaires sur l'application développée. Ces tests permettent de déterminer dans un premier temps si les fonctions implémentées réalisent les tâches pour lesquelles ont été créées.

Les tests unitaires permettent déterminer la concordance entre l'application qui a été développée et la conception réalisée.

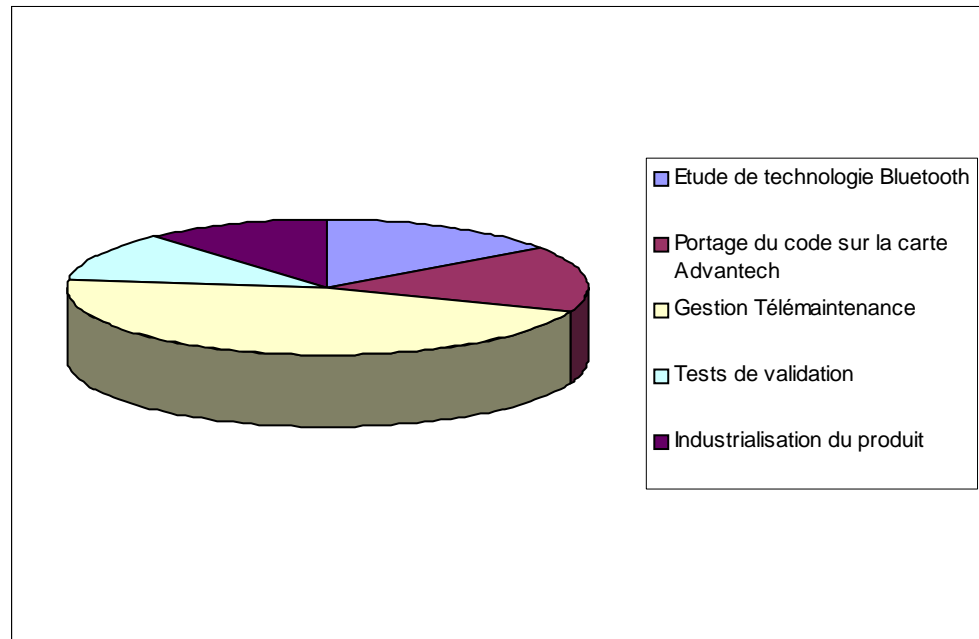
Les tests de validation permettent de déterminer si l'application en général correspond à la spécification réalisée.

### **2.1.3 Partage de ressources**

L'équipe a utilisé un serveur commun pour stocker et partager les informations du projet. Chaque membre de l'équipe peut accéder à la documentation technique ou peut stocker les documents de spécification et conception et les diverses versions du code réalisé.

## **2.2 Travaux Réalisés**

Dans les six mois de la durée de mon stage j'ai travaillé sur plusieurs étapes du développement et de conception finale du produit ; la Fig. 4 montre de manière graphique l'ensemble des travaux réalisés.



**Figure 4: Travaux réalisés**

### 2.2.1 Contrainte du développement

Embedia travaille depuis l'année 2004 en partenariat avec des sociétés impliquées dans le domaine de la publicité extérieure.

Embedia a proposé aux deux sociétés de réaliser un produit spécifique par rapport aux besoins de chacun. Alors le développement (spécification, conception et codage) n'a pas été réalisé de façon statique : chaque fois que le produit a été présenté chez le client, les suggestions données ont été prises en compte afin d'améliorer l'application finale.

Cette démarche de développement permet tant à Embedia comme aux clients d'arriver à un produit performant, puisque les besoins spécifiques et l'ergonomie du produit ont été adaptés de façon consécutive.

La contrainte imposée par cette méthode de travail est, évidemment, le ralentissement du développement. A chaque réforme, il a fallu changer la spécification et la conception, pour pouvoir réaliser le codage des applications.

## **2.2.2 Portage de code sur la carte Finale**

Comme il est mentionné dans le cahier des charges, pour passer à la deuxième phase du produit, il a fallu migrer depuis la carte d'évaluation vers une carte industrielle disponible sur le marché.

Sur les cartes disponibles du marché, on a choisi la carte PCME 7520 d'un fabricant reconnue dans les systèmes embarqués, dans un premier temps au regard des spécifications techniques nécessaires à l'implémentation de l'application interactive Bluetooth.

Cette carte a nécessité de travaux d'adaptation particuliers de façon à lui permettre d'assurer le fonctionnement du protocole Bluetooth sous Win CE. Au début, on a essayé d'utiliser le port USB pour permettre la transmission d'information sur la technologie Bluetooth. On a trouvé des problèmes sur le pilote USB du port puisqu'il n'y avait pas de compatibilité avec la clef Bluetooth USB.

Après réflexions, le choix s'est porté sur une carte Bluetooth PCMCIA et Compact Flash. La mise en place de la carte n'a pas été simple car il a fallu trouver les bons pilotes de la carte.

Une fois les pilotes mis en place, il a fallu configurer la base des registres afin que la carte soit correctement prise en compte par la plateforme.

### **2.2.3 Gestion de Télémaintenance**

Dans ce chapitre sont décrits l'analyse et la conception de la télémaintenance. Les diverses tâches sont abordées spécifiquement avec les sous-processus principaux pour chaque application programmée.

#### **2.2.3.1 Analyse et conception.**

Le module doit communiquer avec l'automate qui est responsable de la gestion des états des capteurs. Afin de permettre au module de déterminer les états des alarmes, il faut réaliser une interface électronique pour adapter les niveaux de tension entre les deux dispositifs (cible et automate).

Pour effectuer les changements des paramètres, le module établit une communication série RS-232 avec l'automate.

##### **2.2.3.1.1 Conception du Hardware**

La carte CPU sur laquelle on développe l'application de télémaintenance permet la lecture des signaux externes à travers un module entrées-sorties présent sur la carte.

Pour effectuer la remontée des alarmes la cible doit communiquer avec l'automate pour s'informer sur l'état de fonctionnement du panneau et, selon le cas, envoyer l'information vers le superviseur qui la traitera.

L'automate gère les signaux entre 0 et 24 volts et le niveau de tension accepté pour la carte est TTL standard, il faut donc réaliser un étage d'adaptation entre les deux dispositifs.

Les sorties de l'automate sont prises entre 0-24 volts, ces sorties indiquent les états des relais qui surveillent le fonctionnement du panneau. Par exemple: défaut d'alimentation, défaut d'éclairage et porte ouverte.

Avec l'étage d'adaptation, on a mis en œuvre un étage d'opto-isolement. Cet étage permet « d'isoler » les différents étapes d'un système, afin de diminuer la propagation d'une surtension, et de protéger les entrées de la cible des possibles fautes en provenance de l'automate.

Après l'étage d'opto-couplage, l'interface rend un signal clair, sans bruit, sur le port de la cible; pour diminuer le risque d'envoyer une fausse alarme, un étage « Schimit Trigger » a été implémenté.

L'Annexe II présente le schéma de l'interface. Au début, l'opto-couplage a été effectué en utilisant le IC **4N27** pour réaliser l'isolement opto-électrique entre les deux parties.

L'opto-coupleur fournit un isolement optique des signaux entre l'automate et la cible sans aucun risque car la tension de rupture pour ce dispositif est de l'ordre de 3 KV. L'opto-coupleur permet également de fixer le niveau de sortie.

### **2.2.3.1.2 Conception Software**

L'application doit permettre la gestion des alarmes, la gestion de la scrutation et la possibilité de changer les paramètres de configuration de l'automate à distance.

L'application est alors divisée en trois modules différents qui seront exécutés simultanément sur la cible. Ces trois parties sont décrites ci-dessous :

- *Gestion des alarmes* : Les alarmes proviennent des entrées de la carte du module. Les entrées sont liées directement aux sorties de l'automate qui vont jusqu'au module GSM.
- *Scrutation* : ce sous-processus permet au superviseur de connaître l'état du module.
- *Scrutation - Configuration* : ce processus permet de changer les paramètres de l'automate qui gère l'activité du panneau.

#### **2.2.3.1.2.1 Gestion des alarmes**

La gestion des alarmes est un processus qui permet la remontée des alarmes vers le superviseur. Le processus est divisé en deux sous-processus réalisant le traitement des événements de façon ordonnée ( Cf Figure 5 sur la page suivante)

Le premier sous-processus T\_LireAlarmes vérifie l'état des entrées à période régulière toutes les 5 secondes. Lorsque une alarme est déclenchée, il effectue l'envoi d'un message d'alarme vers le processus principal qui doit le rediriger vers le superviseur.

Le deuxième sous-processus T\_GesComAlrmSup réalise l'établissement de la connexion TCP/IP entre le module et le superviseur. Il attend un message de notification des alarmes qui contient l'information sur l'état des entrées. Ce message est envoyé par la tâche T\_LireAlarmes. Lorsque T\_GesComAlrmSup a reçu la notification, il ouvre la connexion et construit le paquet de données des alarmes.

La construction du paquet est effectuée après l'établissement de la connexion entre les deux parties. Le protocole permet un premier niveau de sécurité puisqu'il fixe une valeur qui identifie le type de connexion. Il permet la vérification des données reçues car il ajoute l'information de la longueur de l'article dans l'entête du paquet.

Le Tableau 2 décrit l'entête du paquet qui contient la spécification du protocole utilisé et le Tableau 3 montre la définition de l'article Alarmes.

<b>Nom</b>	<b>Code</b>	<b>Description</b>	<b>Taille</b>
Type	??	Dialogue Module vers Superviseur	??
Longueur	??	Longueur en octets du message	??
Donnée	??	Article Alarme	??

*Tableau 2: Entête du paquet*

<b>Nom</b>	<b>Code</b>	<b>Description</b>	<b>Taille</b>
------------	-------------	--------------------	---------------

<b>Nom</b>	<b>Code</b>	<b>Description</b>	<b>Taille</b>
Type	??	Article alarme	??
Longueur	??	Longueur en octets du message	??
Donnée	??	Etat des alarmes	??

Tableau 3: Définition de l'article Alarmes

La figure 5 présente un schéma de la gestion des alarmes ; on montre aussi le dialogue entre les deux sous-processus et le superviseur pour la gestion des alarmes.

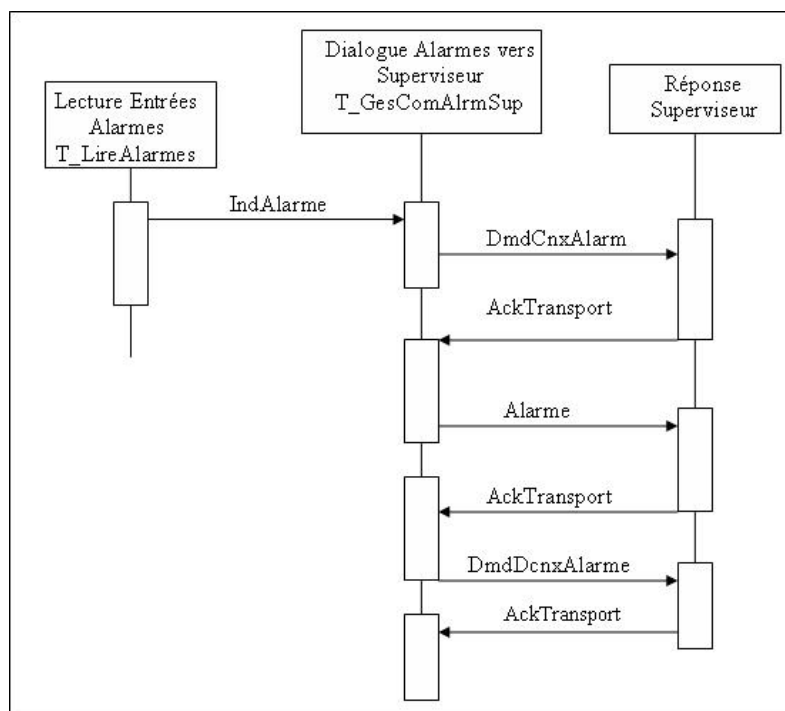


Figure 5 : Gestion Alarmes

Le Tableau 4 présente une description des messages échangés entre les entités qui permettent la gestion des alarmes.

<b>Nom</b>	<b>Description</b>	<b>Source</b>	<b>Destination</b>
IndAlarme	Indication d'une alarme: contient l'état des alarmes lors d'un changement d'état.	T_LireAlarmes	T_GesComAlrmSup
DmdCnxAlar me	Demande de connexion au niveau transport, au superviseur sur le port associé aux alarmes.	T_GesComAl rmSup	Superviseur
Alarme	Article décrivant l'état des alarmes du module.	T_GesComAl rmSup	Superviseur
DmdDcnxAlar me	Demande de déconnexion au niveau transport du service de traitement des alarmes.	T_GesComAl rmSup	Superviseur
AckTransport	Acquittement de la couche transport	Superviseur	T_GesComAlr mSup

**Tableau 4: Description du dialogue Gestion Alarmes.**

#### **2.2.3.1.2.2 Gestion Scrutation**

Ce processus attend une demande de connexion TCP/IP réalisée pour le superviseur. Le protocole défini permet un premier niveau de sécurité de la connexion. L'entête du paquet est décrit dans le tableau ci-dessous, le premier argument permet d'identifier le type de connexion, la longueur est utilisée pour

assurer la bonne réception des données et le dernier paramètre permet de savoir le type de scrutation.

<b>Nom</b>	<b>Code</b>	<b>Description</b>	<b>Taille</b>
Type	??	Dialogue Superviseur vers Module	??
Longueur	??	Longueur en octets du message	??
Donnée	??	Scrutation normale	??

*Tableau 5:Entête de la connexion*

L'application doit être capable d'identifier le type de connexion entrante et de réaliser la scrutation des paramètres comme l'état des alarmes ou le nom du fichier qui est utilisé à ce moment par l'application d'interactivité.

A la fin ce processus doit construire un paquet qui spécifie l'information souhaitée avec une entête qui identifie le type de la connexion, la longueur du paquet et les données comme il est montré dans le tableau suivant :

<b>Nom</b>	<b>Code</b>	<b>Description</b>	<b>Taille</b>
Type	??	Dialogue Module Vers Superviseur	??
Longueur	??	Longueur en octets du message	??
Donnée	??	Article état alarmes	??
	??	Longueur	??
	??	Etat alarmes	??

	??	Article nombre de connexions	??
	??	Longueur	??
	??	Nombre de connexions : hexa	??
	??	Article nom du fichier interactivité	??
	??	Longueur	??
	????	Données : chaîne ASCII cadrée à gauche, terminée à droite avec un zéro.	??

*Tableau 6: Description article Scrutation*

Le processus de scrutation est divisé en deux sous processus :

**T\_ServeurScrutation** : c'est un processus qui permet de répondre à une demande de connexion du superviseur. L'attente est effectuée sur un port utilisateur. La demande est étudiée, si elle correspond à une demande reconnue, alors un message est envoyé à la tâche de traitement afin de fournir les informations demandées. Dans le cas contraire, une déconnexion est appliquée immédiatement.

**T\_GesScrutation** : c'est un processus qui permet de fournir les informations de l'état du module. Ce processus est en attente d'un message ; lorsqu'il reçoit le message, les données sont préparées et envoyées vers le processus *T\_ServeurScrutation*.

La Fig. 6 décrit le dialogue entre les deux sous-processus principaux. Les principales données échangées sont montrées dans le tableau 7.

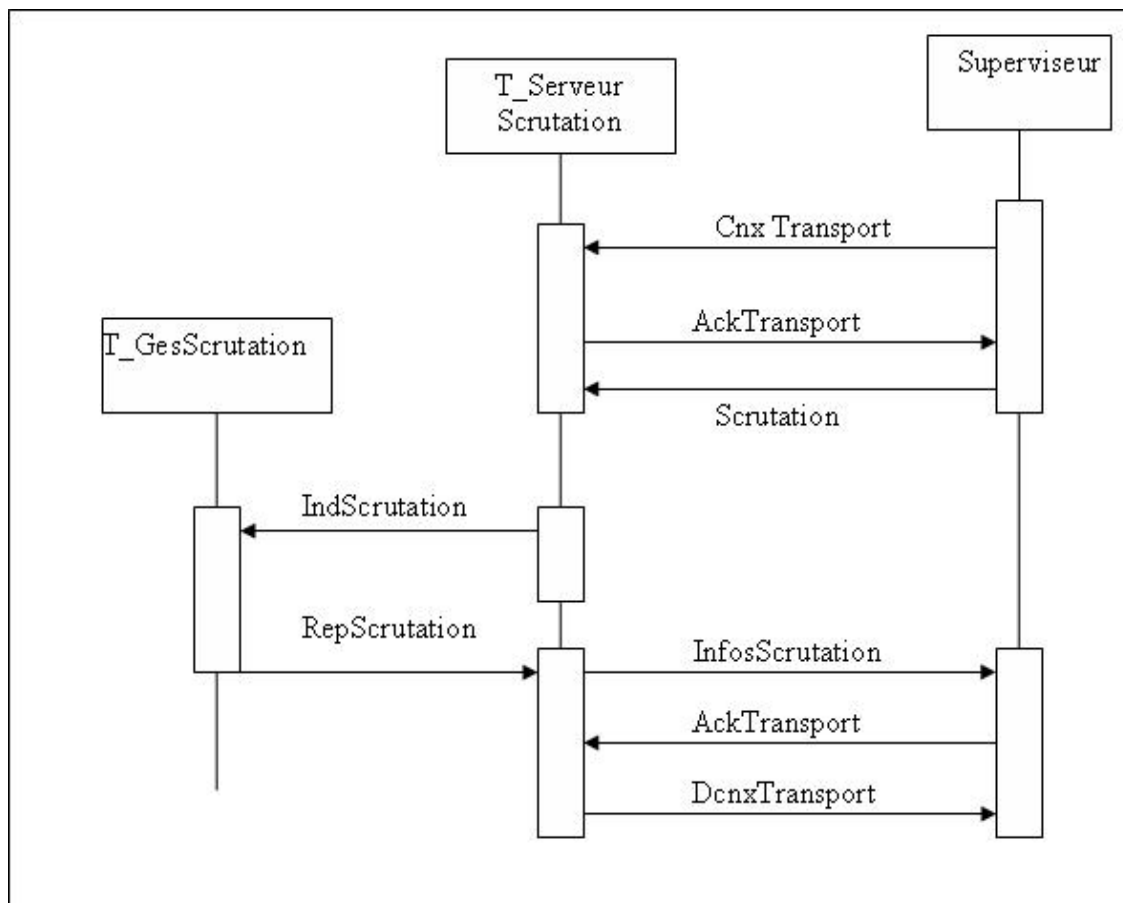


Figure 6 : Gestion Scrutation

Nom	Description	Source	Destination
CnxTransport	Indication d'une demande de connexion transport émanant du superviseur pour le service de scrutation	Superviseur	T_ServeurScrutation

<b>Nom</b>	<b>Description</b>	<b>Source</b>	<b>Destination</b>
Scrutation	Indication d'une demande bien formée de scrutation	Superviseur	T_ServeurScrutation
IndScrutation	Indication des informations de scrutation.	T_ServeurScrutation	T_GesScrutation
RepScrutation	Ce message contient les informations de scrutation	T_GesScrutation	T_ServeurScrutation
InfosScrutation	Les informations de scrutation sont envoyées au superviseur par la couche transport.	T_ServeurScrutation	Superviseur

*Tableau 7: Description des états Scrutation*

### 2.2.3.1.2.3 Gestion scrutation configuration

La gestion du processus « scrutation-configuration » est effectuée en utilisant quatre sous- processus qui permettent l'établissement de la communication série avec l'automate et la transmission de données entre le deux entités. Ces sous processus sont :

1. T\_Cons : il attend la notification de la tâche principale à l'arrivée d'une scrutation -configuration, lorsque cet événement est arrivé, il procède à la configuration de la connexion. La notification est envoyée vers T\_Gestion pour l'informer que la configuration de la connexion est faite.

2. T\_Gestion : c'est la tâche principale, s'il a eu une demande de « scrutation configuration », cette tâche envoie un message vers T\_Cons, elle reste en attente de la notification de réalisation de la configuration. En même temps cette tâche réalise l'administration de la transmission et de la réception des données avec l'automate, et l'envoi de l'information vers le superviseur.
3. T\_Trasmission : cette tâche réalise l'écriture sur le port des données qui sont envoyées pour T\_Gestion.
4. T\_Reception : cette tâche effectue la lecture du port et réalise l'envoi de l'information vers la tâche de gestion.

Pour envoyer les données depuis le module vers le superviseur, la construction du paquet est faite de la même façon que dans le cas de la scrutation normale. Le code des articles doit changer pour spécifier le paramètre qui veut être modifié dans l'automate.

La description de l'entête de connexion est montrée dans le tableau 8.

<b>Nom</b>	<b>Code</b>	<b>Description</b>	<b>Taille</b>
Type	??	Dialogue Superviseur vers Module	??
Longueur	??	Longueur en octets du message	??
Donnée	??	Scrutation Configuration	??

*Tableau 8: Entête de la connexion scrutation configuration*

La fig. 7 présente un schéma du dialogue entre les tâches principales du processus de scrutation – configuration.

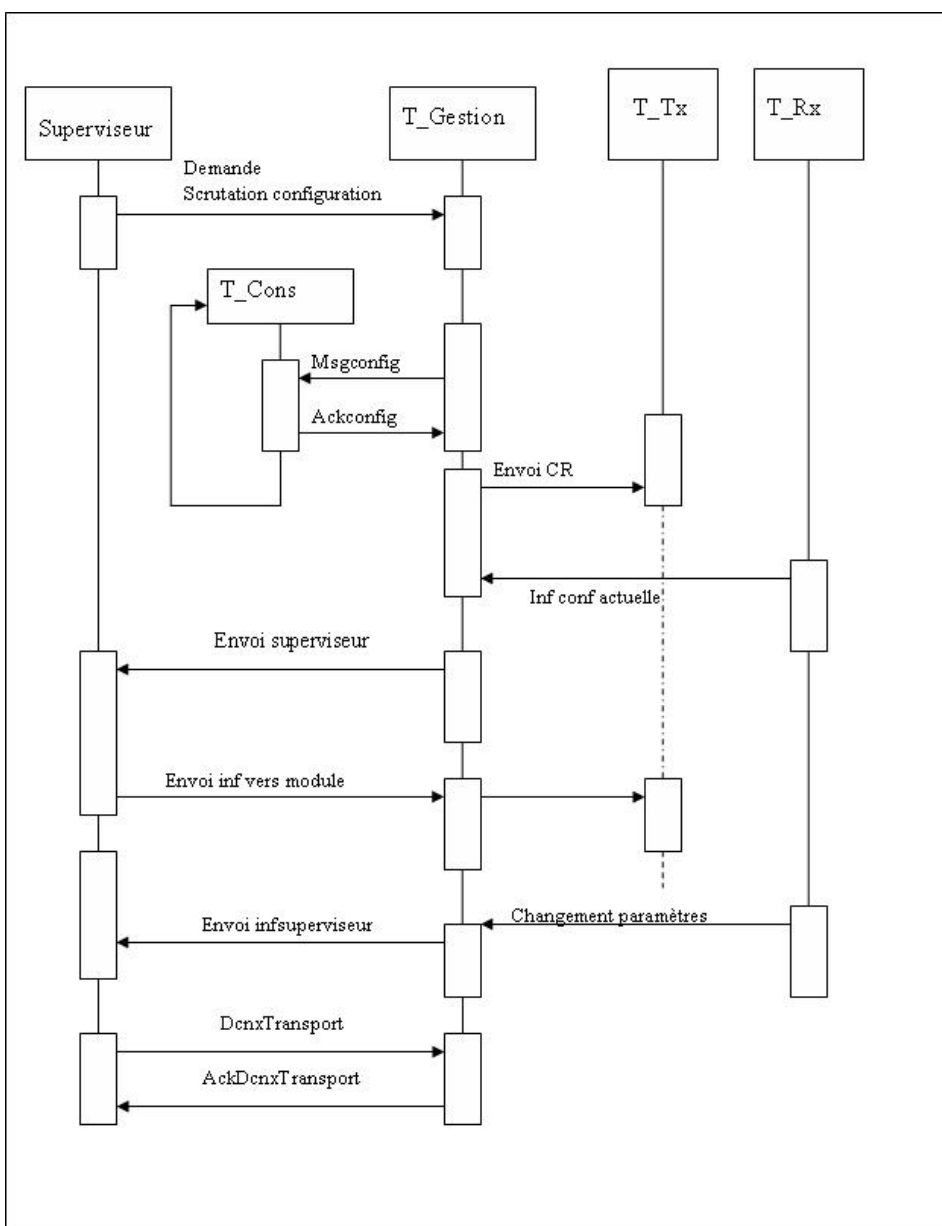


Figure 7 : Gestion « Scrutation- configuration »

Le tableau 9 décrit les informations échangées entre les tâches.

<b>Nom</b>	<b>Description</b>	<b>Source</b>	<b>Destination</b>
Demande de Scrutation -configuration	Une connexion entrante arrive sur la tâche principale, en demandant une scrutation configuration.	Superviseur	T_Gestion
MsgConfig	Demande de configuration de la connexion Série entre la cible et l'automate.	T_Gestion	T_Cons
Ackconfig	Acquittement de la réalisation de configuration.	T_Cons	T_Gestion
Envoi CR	Envoi du caractère nécessaire vers la tâche de transmission pour avoir l'état de configuration de l'automate.	T_Gestion	T_Tx
Inf configuration actuel	La tâche de réception envoie les données reçues vers la tâche principale avec l'état de configuration.	T_Rx	T_Gestion
Envoi superviseur	Envoi de données vers le superviseur d'information relative à l'automate.	T_Gestion	Superviseur.

Envoi vers module	Information envoyée pour le superviseur vers le module pour changer les paramètres de configuration de l'automate.	Superviseur	T_Gestion
Changement des paramètres	Réception de l'information qui envoie l'automate pour configurer son fonctionnement.	T_Rx	T_Gestion.
DcnxTransport	Demande de déconnexion	Superviseur	T_Gestion
AckDcnxConnexion	Acquittement de déconnexion	T_Gestion	Superviseur.

Tableau 9: Description des états "Scrutation Configuration"

## **2.2.3.2 Développement**

### **2.2.3.2.1 Environnement de programmation**

Le développement du module a été effectué dans l'environnement Windows CE. Cet environnement est conçu spécialement pour des systèmes embarqués car il donne la possibilité de construire une plateforme adaptée au niveau logiciel, au matériel spécifique sur lequel les applications seront développées.

L'environnement possède deux scénarios de développement. Le premier scénario est « Platform Builder » sur lequel on effectue la construction de la plateforme. Ce premier scénario permet d'ajouter les pilotes nécessaires pour chaque caractéristique de la plateforme et de réaliser les modifications sur la base de registres de l'environnement, ainsi que de générer l'image qui sera chargée dans la mémoire du système embarqué.

Le deuxième scénario s'appelle « Embedded Visual C++ » ; c'est l'environnement sur lequel on réalise le développement des applications. Après la génération, les applications doivent être chargées sur la plateforme pour exécution.

### **2.2.3.2.2 Développement des applications**

Comme précisé, les applications ont été développées sur « Embedded visual C++ », qui est un langage de programmation orienté à objets. Les différentes tâches sont présentées ci-après.

### 2.2.3.2.2.1 Gestion des alarmes

Le premier thread `T_LireAlarmes` réalise la lecture des états de fonctionnement du panneau. Le thread une fois initialisé, effectue une lecture des sorties automate chaque 5 sec. Si un changement d'état est détecté, `T_LireAlarmes` envoie cette information vers le thread principal.

Le deuxième thread `T_GesComAlrmSup` une fois initialisé effectue la configuration du socket TCP. Le thread reste suspendu en attendant la notification d'un changement des alarmes (`IndAlarmes`). Lorsque cette notification arrive, le processus réalise l'établissement de la connexion vers le superviseur pour l'envoi de l'information. La structure générale du processus est décrite à la **Figure 8**.

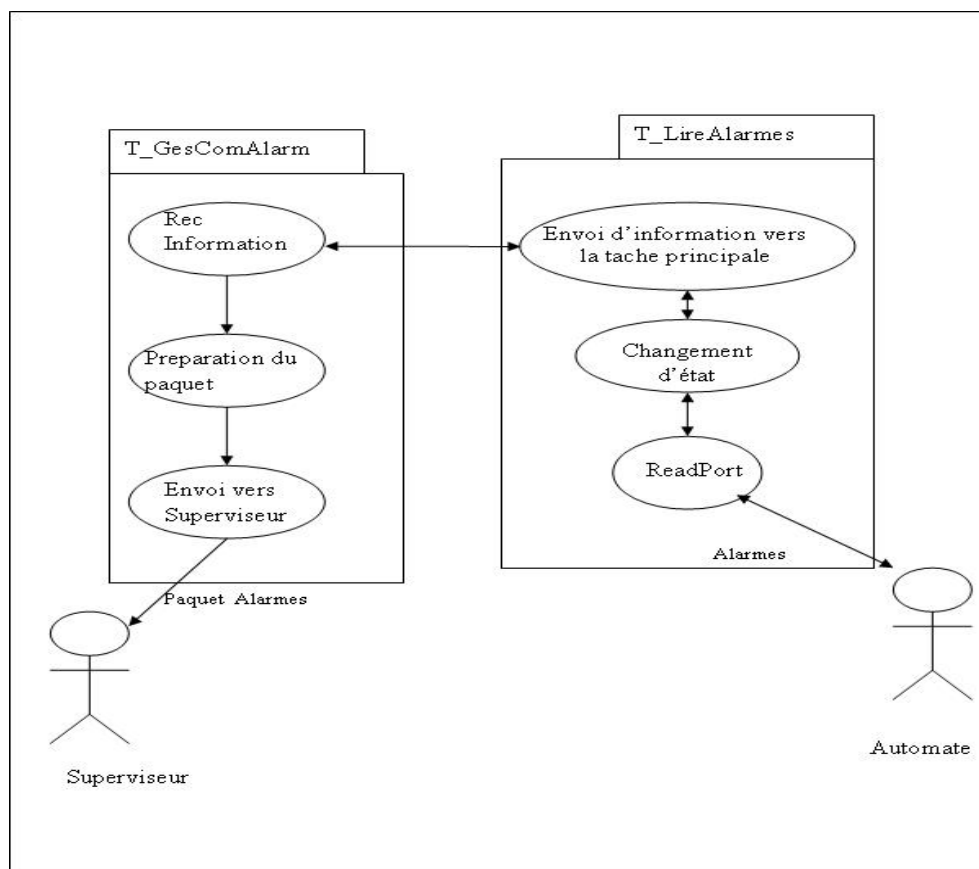


Figure 8: Gestion des Alarmes

### 2.2.3.2.2 Gestion de la scrutation

Une scrutation normale déclenchera l'envoi d'un message sur la tâche de scrutation des alarmes. Ensuite, la tâche principale envoie l'information et ferme la connexion en libérant le port. Le processus est décrit à la **Figure 9**.

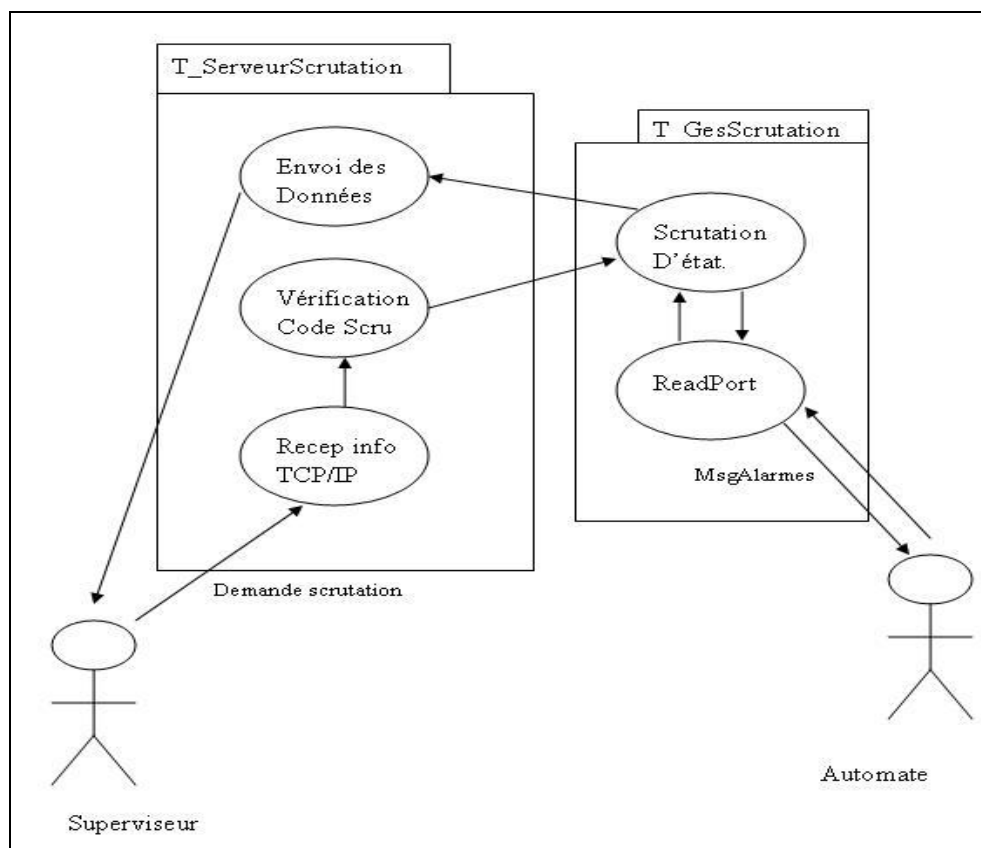


Figure 9 : Scrutation

### 2.2.3.2.2.3 Gestion Scrutation - Configuration

Le processus a été créé à partir des threads définis lors de la phase de conception. Une fois créés, ils restent suspendus jusqu'à ce que la tâche principale notifie de l'arrivée d'une nouvelle scrutation du type configuration.

Lorsque la notification de « scrutation configuration » arrive au thread principal, celui-ci est réveillé et produit le déclenchement de l'activité qui permet la connexion avec l'automate.

A la fin, la tâche principale attend une demande de déconnexion afin de libérer la connexion avec l'automate et libérer le port de la connexion établie avec le superviseur. La **Figure 10** montre les échanges réalisés entre les divers états.

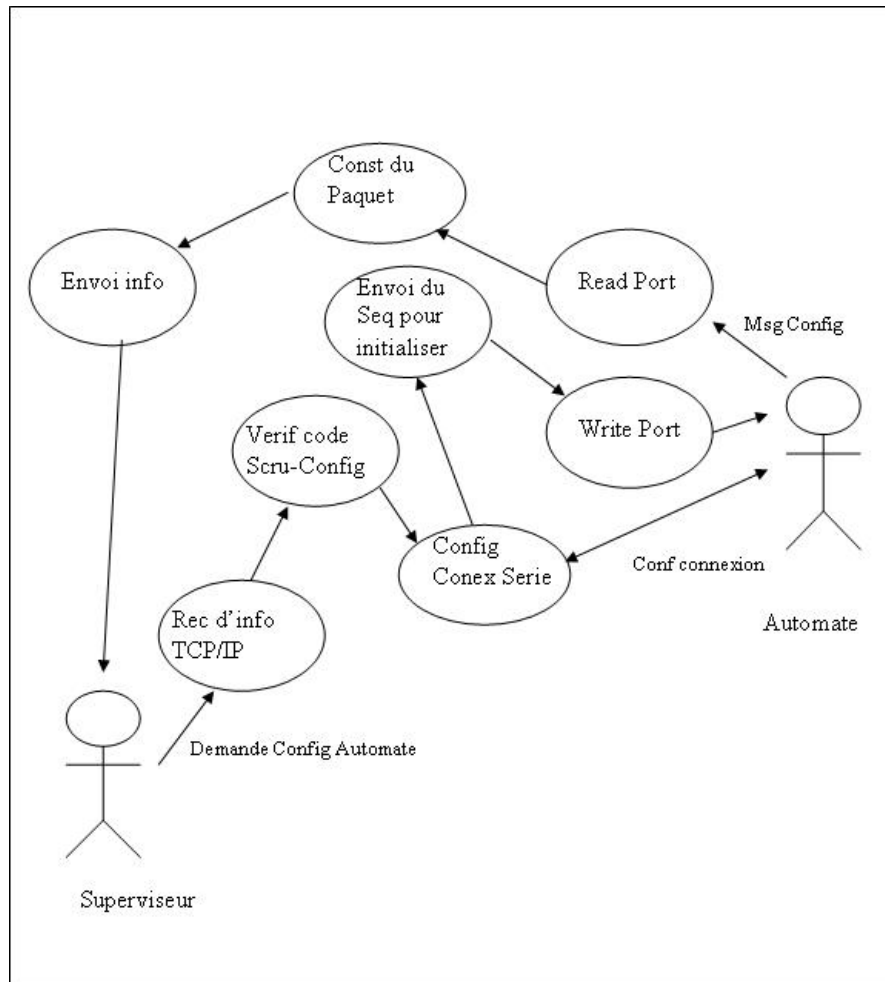


Figure 10: Scrutation-Configuration

### 2.2.4 Test de validation de l'application d'interactivité

La troisième partie de mon stage a consisté à valider les différentes versions de l'application d'interactivité sur Bluetooth.

La réalisation des tests de validation a permis de vérifier le fonctionnement correct de l'application sur le terrain. Dans un premier temps, le développement et la validation étaient réalisés par la même personne, ce qui ne permettait pas de mettre en évidence tous les dysfonctionnements. De ce fait, ces deux étapes ont été désormais réalisées par deux personnes différentes. L'étape de validation a été soumise à ma responsabilité.

#### **2.2.4.1 Test de contact entre la cible et une oreillette**

Lorsque la cible effectue la découverte de périphériques, elle réalise la demande de services sur chaque dispositif. Si le dispositif en face ne possède pas le service de réception de fichier, la cible ignore ce dispositif. Chaque périphérique est identifié par une information COD (class of device) qui permet de déterminer ses caractéristiques lors de sa découverte.

#### **2.2.4.2 Test de contact entre la cible et un ordinateur**

On a réalisé le même test de contact entre la cible et un ordinateur avec le filtrage sur le COD. L'ordinateur n'a pas été contacté, comme il était désiré. Cependant, un problème a été découvert : l'ordinateur est capable de voir le module (la cible) car le mécanisme pour se masquer n'est pas implémenté sur celui-ci. On a essayé sans succès de masquer la cible; ceci est dû à la limitation présentée par l'environnement de programmation WinCE 4.2.

#### **2.2.4.3 Test sur les divers téléphones par rapport à l'ergonomie**

Divers tests d'observation du comportement par rapport à l'ergonomie de l'application ont été menés.

Chaque téléphone possède une ergonomie différente. Pour les premiers téléphones portables qui sont sortis sur le marché, l'envoi d'information est réalisée en mode PUSH. Le mode PUSH change la réalisation du contact effectué par la cible, puisque le jumelage n'est pas obligatoire.

Les téléphones récents, permettent l'envoi d'information à travers le mode FTP, lequel change, à tout point de vue, l'ergonomie de contact de l'application car le jumelage est obligatoire.

Les tests ont été effectués sur les deux catégories de téléphones. Lorsqu'un problème est révélé, il faut remplir une fiche d'anomalie. Cette fiche permet de décrire le problème, de le reproduire si besoin est, puis d'effectuer sa correction du point de vue logiciel.

De cette façon on a réussi, de manière itérative, à réaliser une version performante de la version d'interactivité ce qui est le but principal de la société.

#### **2.2.4.4 Test du temps de premier contact. (Time of scan)**

Le temps de scan est une variable très importante pour que l'application fasse l'envoi du fichier de façon optimale. L'application utilise une fonction spécifique pour réaliser le scan de périphériques. Cette fonction permet de gérer le temps de scan. Pour déterminer les paramètres optimaux, on a réalisé le test sur les fonctions isolées qui permettent la recherche de périphériques.

Comme on ne disposait pas d'information relative à l'implémentation de la norme Bluetooth, l'estimation des paramètres a été effectuée par une méthode itérative en faisant varier chacun des paramètres. L'application réalisée permettait de récupérer le nombre de dispositifs contactés à chaque fois. Avec les informations récupérées, les études statiques sur chaque paramètre; un jeu de paramètres optimal a été extrait : celui qui a permis de minimiser le temps du scan et de maximiser le nombre de dispositifs repérés.

### **2.2.5 Industrialisation du produit**

Les négociations avec la société qui exploite les panneaux d'affichage ont abouties. La première commande du module Embedia a été passée par le client; alors il a fallu préparer les modules avec l'application d'interactivité.

C'est donc la dernière partie de mon stage, pour laquelle toute l'équipe a travaillé dans le processus d'industrialisation. Ce processus a consisté à la mise en place de la carte électronique et des éléments à l'intérieur d'un coffret.

Les indicateurs sont de LEDs qui sont mises sur la façade du module ; à chaque LED correspond un état spécifique. La première LED indique la présence de l'alimentation de la carte ; la deuxième informe sur la bonne réalisation de la mise à jour du contenu et la troisième permet de visualiser les activités Bluetooth générées par la carte.

### **2.3 Etat du produit**

L'état du produit dans les trois axes a bien avancé. Les études continuent sur la communication locale pour bien maîtriser le protocole Bluetooth. Le développement sur le réseau Wifi est en cours, afin de relier les modules avec le protocole de routage OLSR. En ce qui concerne la partie de télémaintenance, les applications ont été développées et les démarches pour la conception de la nouvelle carte ont débutées.

### **2.3.1 Communication locale**

L'application de l'interactivité a été bien développée ; la dernière version de l'application que l'équipe a élaboré a été incluse dans la première commande du produit. Cette application permet l'envoi de l'information depuis la cible vers les téléphones portables proches. L'application réalise la différenciation entre les deux ergonomies (Modes Push et FTP) ; elle réalise également le filtrage selon le type de périphérique et, avec les tests réalisés, on a réussi à déterminer les paramètres optimaux pour effectuer la recherche des périphériques. L'interactivité a été testée sur le terrain et a atteint un bon niveau de performance.

### **2.3.2 Communication à distance**

Pour atteindre ce but, nous avons dû :

- Mettre en place un réseau ad-hoc et réaliser la connexion des cibles à ce réseau
  - Installer le pilote de la carte WiFi
  - Configurer le réseau
  - Développer les utilitaires nécessaires pour la connexion automatique

- Assurer le bon fonctionnement de ce réseau :
  - Utiliser un protocole de routage (OLSR en l'occurrence)
  - Développer les applications pour contrôler l'état actuel de chaque cible et aussi du réseau entier (via le mécanisme des alarmes).

En ce moment, toutes ces tâches sont bien effectuées ; nous avons obtenu un réseau de type ad-hoc qui répond à notre besoin. Celui-ci permet aux modules de communiquer entre eux et avec le superviseur lors des opérations comme la diffusion, la scrutation et l'envoi des alarmes.

### **2.3.3 Gestion de télémaintenance**

La gestion de la télémaintenance a bien avancé. Les applications développées permettent la gestion des alarmes et le changement des paramètres de l'automate. Lorsque mon stage a commencé, il n'y avait aucun développement sur ce sujet.

Les applications ont été testées sur un panneau réel pour vérifier la remontée des alarmes vers le poste du superviseur. La société prévoit l'établissement du réseau Wifi pour Janvier 2006 et de cette façon il sera possible de réaliser la présentation complète du produit à la société constructrice de panneaux.

En ce moment, il y a une autre société qui développe la nouvelle carte Embedia qui permettra l'intégration d'un module pour adapter les signaux entre l'automate et la carte. Dans l'avenir Embedia veut développer les applications nécessaires pour que la cible remplace l'automate et, de cette façon, fournir une solution plus complète pour le client.

### 3 Synthèse et conclusions

L'expérience obtenue m'a permis d'être témoin de la naissance d'une nouvelle entreprise. J'ai participé dès le début du développement du produit jusqu'à la première commande, ce qui m'a permis de connaître les divers processus qui doivent avoir lieu pour concevoir un produit. Le processus du développement qu'on a mis en place (Cycle en V) nous a permis d'apprendre les étapes nécessaires pour réaliser le développement de logiciel de façon industrielle.

La contrainte du développement a été la modification constante des besoins du client. Cette façon de développer a été imposée par Embedia et a permis sans doute d'arriver à la constitution d'un produit performant, puisque le client a lui-même choisi l'ergonomie d'exploitation.

Les différentes présentations qui ont eu lieu chez le client ont été des opportunités réelles d'obtenir le premier contrat pour la société. Ce contexte de travail a imposé une rigueur aux tâches quotidiennes qui m'a permis d'enrichir mes capacités professionnelles.

Tout au long de mon projet, j'ai eu l'occasion de participer de manière active dans plusieurs domaines du projet global. Cette expérience de travailler dans un contexte de multi-projet m'a permis de renforcer ma formation d'ingénieur, de mieux intégrer, l'environnement et le fonctionnement d'une entreprise en création.

En ce qui concerne la part technique, j'ai pu apprendre sur les technologies de communication plus récentes. Le parcours réalisé sur Bluetooth pour valider

l'application de l'interactivité m'a permis de connaître le protocole à tous les niveaux, complétant ainsi les connaissances que j'avais sur cette technologie.

L'étude réalisée dans la première partie de mon stage sur l'ensemble de la technologie Bluetooth m'a permis d'acquérir les connaissances nécessaires pour effectuer les divers tests de l'application d'interactivité. Il a fallu étudier la norme Bluetooth pour diriger ces tests, en particulier pour déterminer le temps de scan de l'application.

La gestion de la télémaintenance a été le domaine sur lequel j'ai notamment travaillé. Pour réaliser la programmation des applications j'ai dû apprendre à utiliser l'environnement Win CE, cet environnement permet le développement complet des systèmes embarqués, lequel est un outil important pour les futures démarches dans ce domaine.

La programmation des applications dans l'environnement Win CE, m'a permis de consolider mes connaissances sur la programmation orientée objets ( Embedded Visual C++); l'utilisation des « threads » et des « messages queue » pour réaliser la communication entre les divers sous-processus est une façon intéressante d'aborder le développement des applications en temps réel et multi-Tâches , dû au contrôle qu'on peut avoir pour la transmission de données entre sous- processus.

Il faudra être capable de gérer la priorité entre la gestion de télémaintenance et l'application d'interactivité, parce que si une alarme est produite l'application d'interactivité doit être suspendue pendant que le problème généré est résolu. Il se peut que la meilleure façon soit le développement d'une application générale qui gère les applications spécifiques comme l'interactivité et la télémaintenance.

## Bibliographie

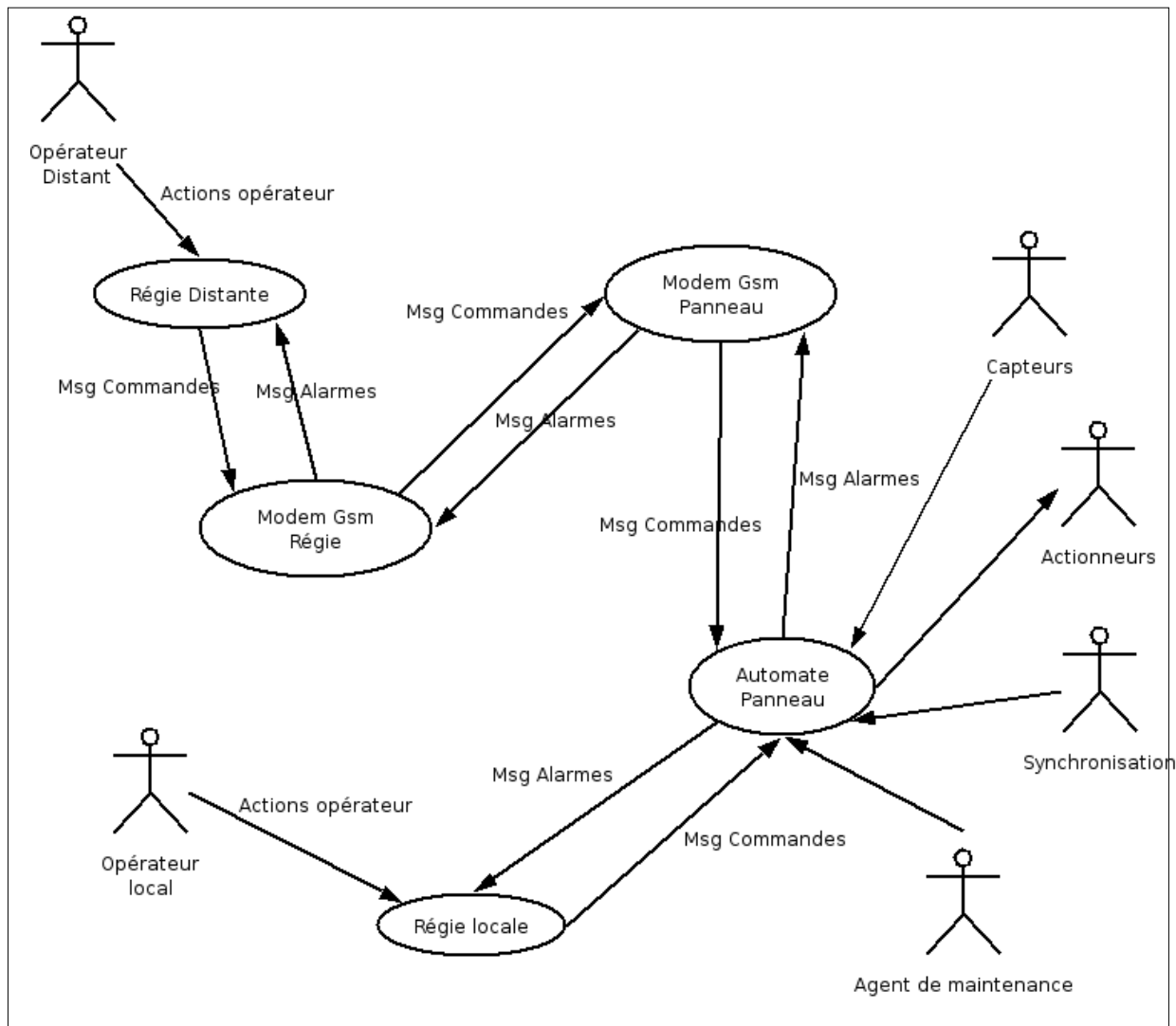
- [1] Boling, Douglas. Titre: Windows CE .NET third edition. Editor Microsoft Press.
- [2] Site Web de référence sur la technologie Bluetooth.  
[www.bluetooth.org/](http://www.bluetooth.org/).
- [3] MSDN Library.  
[msdn.microsoft.com](http://msdn.microsoft.com)
- [4] Manuel C++.  
[www.cplusplus.com/doc/tutorial/](http://www.cplusplus.com/doc/tutorial/)
- [5] [www.vunet.fr](http://www.vunet.fr) – 9 Septembre 2004 /
- [6] [www.neteconomie.com](http://www.neteconomie.com)
- [7] [www.imsresearch.com](http://www.imsresearch.com)
- [8] <http://www.orcad.com/>



---

## ANNEXES

## Annexe I



<b>Nom</b>	<b>Description</b>
Opérateur distant	Individu qui utilise le logiciel de Régie depuis un poste informatique à distance par rapport au panneau.
Opérateur local	Individu qui utilise le logiciel de Régie depuis un poste informatique à proximité du panneau.
Capteurs	Eléments qui permettent de convertir un signal de terrain en une information destinée à l'automate.
Actionneurs	Eléments qui transforment une information automate en une action.
Synchronisation	Elément d'information issu d'un automate et à destination d'un ensemble d'automates afin d'assurer synchronisation des affichages.
Agent de maintenance	Individu chargé de la maintenance du panneau. Il opère localement par rapport au panneau.

## Annexe II

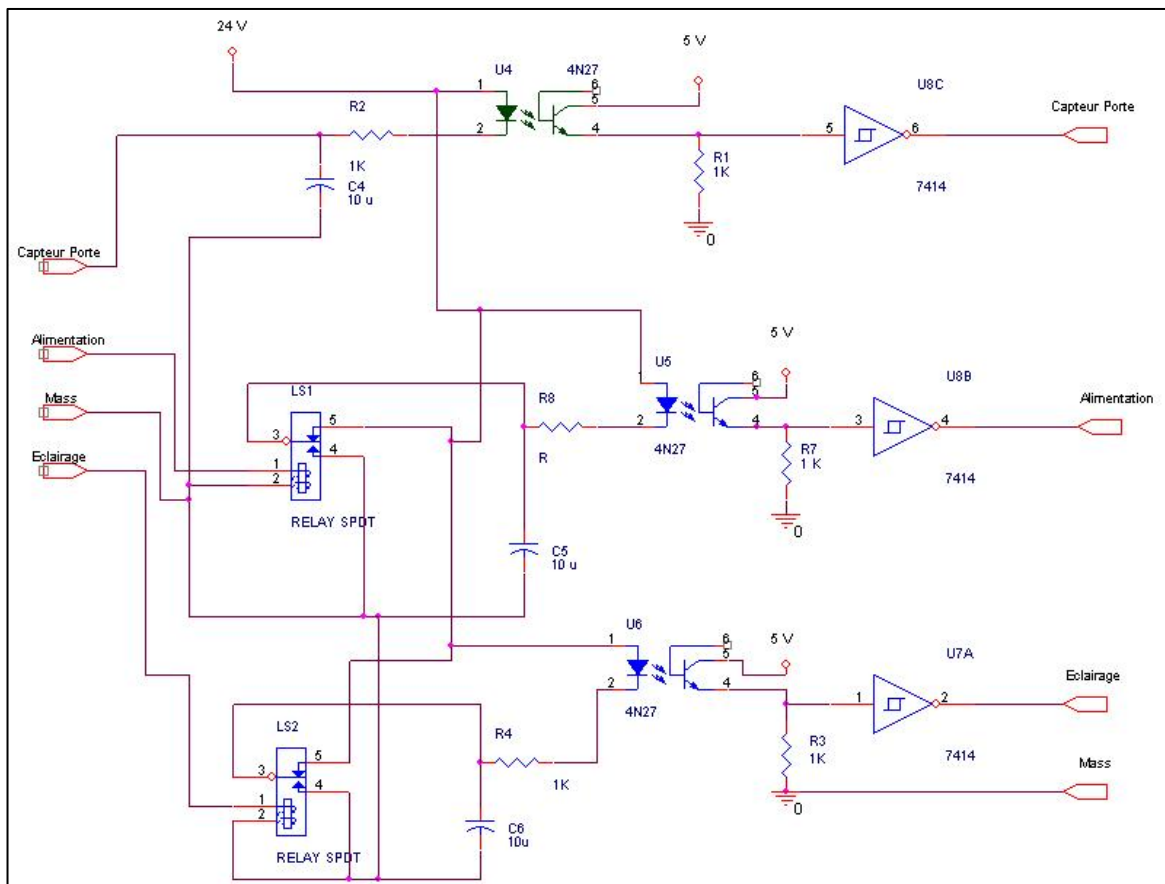


Diagramme circuit électronique