

وزارة التعليم العالي والبحث العلمي

BADJI MOKHTAR- ANNABA UNIVERSITY
UNIVERSITE BADJI MOKHTAR ANNABA



جامعة باجي مختار - عنابة

Année : 2019

Faculté: Sciences de l'Ingéniorat
Département: Electronique

MEMOIRE

Intitulé :

**Nœud de capteur d'images sans fil à faible coût
énergétique**

Présenté en vue de l'obtention du diplôme de : MASTER

Domaine : Sciences et Technologie

Filière : Télécommunications

Spécialité: Réseaux et Télécommunications

Par :

BOUTIOUTA Mohammed Islam

DEVANT Le JURY

Président : M. MOKHNACHE Azzouz

Docteur UBM Annaba

Directeur de mémoire : M. DOGHMANE Nouredine

Professeur UBM Annaba

Examineurs1: M. HAFS Toufik

Docteur UBM Annaba

Examineurs2 M. NASRI Seif A.M

Docteur UBM Annaba

Dédicaces

Je dédie ce modeste travail à

ceux qui nous ont donné la vie, le symbole de la tendresse, qui ont fait tant de

Sacrifices pour notre bonheur et notre succès, pour ma mère et mon père.

À nos parents, notre école d'enfance, dans laquelle nous étions des ombres

Toutes les années d'études où ils n'ont cessé de nous surveiller et tout au long de notre vie

Pour nous encourager, nous aider et nous protéger.

Qu'Allah les protège

À nos frères et sœurs

Pour nos professeurs

Pour nos amis d'études

À nos amis de toujours

Et tous nos amis que nous avons rencontrés le long de notre vie.

Afin d'être reconnaissant envers ceux qui nous ont soutenus et encouragés à

Effectuer ce travail de recherche.

Remerciements

Merci notre Dieu (Allah) de nous avoir donné la capacité d'écrire

et de réfléchir, la force d'y croire, la patience d'aller

jusqu'au bout de nos rêves et le bonheur de lever nos

Mains vers le ciel et prier notre Dieu pour la réconciliation

Pour commencer,

Je tiens à exprimer mes sincères remerciements

les plus sincères à mon Encadreur de mémoire

le professeur

DOGHMANE Noureddine

pour sa grande disponibilité et ses encouragements tout au long de la réalisation

de ce mémoire, de sa contribution dans l'élaboration de ce

mémoire ainsi qu'à la réussite de cette formidable année universitaire.

Je tiens à remercier également tous nos enseignants qui nous ont tant donné et formé.

Enfin Je remercie tous nos proches

Imad, Nadjmou, Mohamed, Saif

et Ahmed qui ils nous ont toujours soutenu et encouragé

avec une bonne humeur à toute épreuve.

ملخص

الهدف الرئيسي لمشروع درجة الماجستير هذا هو اقتراح حل للأجهزة في شكل اختبار (أو مقعد اختبار) يمثل عقدة الاستشعار البصرية اللاسلكية التي سيتم تنفيذها كحل عالمي وسيتم تحسينها وتقييمها. الغرض من هذا الإنجاز هو تطبيق طريقة بسيطة وفعالة لمراقبة بعض الطيور المهتدة في منطقة عنابة. في هذا الصدد ، يجب أن يفي تحقيقنا بالعديد من المتطلبات مثل بساطة التنفيذ وانخفاض استهلاك الطاقة والمرونة وبالطبع السعر الغير مكلف . لذلك اخترنا معدات بأسعار معقولة ومتاحة وبسيطة نسبياً للاستخدام. تم تجميع الحل المقترح واختباره وتقييم النتائج التي تم الحصول عليها وفقاً لمعايير مختلفة عامة الاستخدام .

Résumé

L'objectif principal de ce projet de fin d'études master est de proposer une solution matérielle sous forme d'un "testbed" (ou banc d'essai) représentant un nœud capteur visuel sans fil sur lequel une solution globale sera implémentée, optimisée et évaluée. La finalité de cette réalisation est de mettre en œuvre un moyen simple et efficace permettant de surveiller certains oiseaux menacés dans la région d'Annaba. A ce propos notre réalisation doit répondre à plusieurs exigences comme par exemple la simplicité de la mise en œuvre, la faible consommation énergétique, la flexibilité et bien sûr le fiable coût. Nous avons donc opté pour des équipements abordables, disponibles et relativement simples quant à l'utilisation. La solution proposée a été montée, testée et les résultats obtenus sont évalués sous différents critères généralement utilisés.

Abstract

The aim of this master thesis is to propose a hardware solution in the form of a testbed representing a wireless visual sensor node on which a global solution will be implemented, optimized and evaluated. The purpose of this achievement is to implement a simple and effective way for the monitoring some threatened birds in the Annaba region. In this regard our realization must meet several requirements such as simplicity of implementation, low energy consumption, flexibility and of course an inexpensive price. We have therefore opted for affordable, available and relatively simple equipment for use. The proposed solution has been assembled, tested and the obtained results are evaluated under different criteria generally used.

LISTE DES FIGURES

Figure	page
Figure 1 : Module Arduino Due	10
Figure 2 : Caméra OV7670 & brochage	11
Figure 3 : L'interfaçage de la caméra OV7670 avec un module arduino Due	12
Figure 4 : Photographie de la caméra OV7670 interfacée avec un module Arduino Due	12
Figure 5 : Module XBee PRO S1 et brochage	13
Figure 6 : Modèle en couche du fonctionnement de XBee	14
Figure 7 : Exemple d'interfaçage en un arduino et un XBee	15
Figure 8: Exemple d'interfaçage en un PC et un XBee	15
Figure 9 : Schéma de principe d'un codec JPEG	20
Figure 10: Le graphe de flux de signal (SFG) de la transformation "Pruned RDCT"	24
Figure 11 : Résultat de la compression et transmission Image_Exemple1	27
Figure 12: Courbe PSNR/ bit Rate bpp pour Image_Exemple1	28
Figure 13 : Courbe Temps de codage et transmission / bit Rate bpp pour Image_Exemple1	28
Figure 14 : Résultat de la compression et transmission Image_Exemple2	29
Figure 15: Courbe PSNR/ bit Rate bpp pour Image_Exemple2	30
Figure 16 : Courbe Temps de codage et transmission / bit Rate bpp pour Image_Exemple2	30
Figure 17 : Résultat de la compression et transmission Image_Exemple3	31
Figure 18: Courbe PSNR/ bit Rate bpp pour Image_Exemple3	32
Figure 19 : Courbe Temps de codage et transmission / bit Rate bpp pour Image_Exemple3	32
Figure 20 : Schéma de principe utilisé pour la mesure expérimentale du courant consommé	35
Figure 21 : Interface graphique du logiciel de configuration et contrôle de l'oscilloscope	36
Figure 22 : Courant consommé instantanément par le module XBee lors de la transmission Essai 1	37
Figure 23 : Courant consommé instantanément par le module XBee lors de la transmission Essai 2	37

LISTE DES TABLEAUX

Tableau	page
Tableau 1 : PSNR vs Bitrate et temps de calcul pour Image_Exemple1	27
Tableau 2 : PSNR vs Bitrate et temps de calcul pour Image_Exemple2	29
Tableau 3 : PSNR vs Bitrate et temps de calcul pour Image_Exemple3	31

LISTE DES ABREVIATIONS

ADC : Analog to Digital Converter
AGC : Automatic Gain Control
CPU : Central Processing Unit
CSI : Camera Serial Interface
DCT : Discrete Cosine Transform
DSSS : Direct Sequence Spread Spectrum
FFT : Fast Fourier Transform
EIRP : Effective Isotropically Radiated Power
FPGA : Field-Programmable Gate Array
GPS : Global Position System
HD : Haute Définition
HDMI : High-Definition Multimedia Interface
HEVC : High Efficiency Video Coding
HF : Haute Fréquence
I2C : Inter-Integrated Circuit
IoT : Internet of Things
IP : Internet Protocol
ISM : Industrielles, scientifiques et médicales
IUCN: International Union for Conservation of Nature
JPEG : Joint Photographic Expert Group
QoS : Quality of Service
QoE : Quality of Experience
LoRa : Long Range
MAC : Media Access Control
MPEG : Moving Pictures Experts Group
NiCd : Nickel-cadmium
NiMH : Nickel-hydrure métallique
NiZn : Nickel-zinc
NOx : Oxydes d'azote
OTAP : Over the Air Programming
PC : Personal Computer
PHY : Physical
PSNR : Peak Signal to Noise Ratio
RAM : Random Access Memory
RCSF : Réseaux de capteurs sans fil

RF : Radio-fréquence

ROI : Region of Interest

SCCB : Serial Camera Control Bus

SFG : Signal Flow Graph

SRAM : Static Random Access Memory

UART: Universal asynchronous receiver-transmitter

UICN : Union internationale pour la conservation de la nature

USB: Universal Serial Bus

VGA : Video Graphics Array

WiFi : Wireless Fidelity

WMSN : Wireless Multimedia Sensor Network

WSN : Wireless Sensor Network

WVSN: Wireless Visual Sensor Network

Sommaire

Titre	page
Introduction générale	01
Chapitre 1 : Position du problème	02
1. Introduction	02
2. Etat de l'art	03
3. Réseaux de capteurs sans fil	05
4. Conclusion	07
Chapitre 2 : Solution adoptée	08
1. Introduction	08
2. RCSF comme solution	08
3. Prototype adopté	09
3.1 Module Arduino Due	09
3.2 Module caméra OV7670	10
3.3 Module de communication XBee	13
4. Conclusion	16
Chapitre 3 : Algorithmes de traitement proposés	17
1. Introduction	17
2. Capture d'images	17
3. Compression d'images	19
4. Transmission d'images	24
5. Réception des images au niveau du PC	24
6. Conclusion	25
Chapitre 4 : Tests et évaluation	26
1. Introduction	26
2. Exemples d'expérimentations réalisées	26
2.1 PSNR et temps de compression/transmission en fonction du bitrate	26
2.2 Consommation énergétique due à la compression	33
2.2.1 Algorithme de compression à faible coût calculatoire	33
2.2.2 Consommation énergétique due à la transmission	34
3. Conclusion	38
Conclusion générale	39
Références bibliographiques	41

Introduction générale

Le présent projet de fin d'études master réseaux et télécommunications, que nous avons réalisé, vient dans l'optique de proposer une solution technologique à base de RCSF visuels et éventuellement scalaires (sons), pour faire en premier lieu la surveillance de certains oiseaux dans leur habitat naturel à partir de l'acquisition et la transmission de photographies. Dans un second lieu, cette solution a aussi pour but de collecter des données sur de longues durées pour contribuer à agrandir la base des connaissances par rapport à une espèce aviaire donnée. En effet, moyennant des algorithmes de traitement d'images/audio avancés effectués au niveau du centre distant de traitements, le système proposé sera capable de distinguer les différentes espèces captées et suivies. Ainsi, ce système permettra d'accompagner les autorités compétentes dans les recensements d'espèces présentes dans un espace forestier et/ou humide donné.

L'objectif global du de ce travail est de proposer des solutions technologiques basées sur des réseaux de capteurs sans fil (RCSF) pour le monitoring de la faune et même la flore. Plusieurs défis et challenges doivent être levés. Parmi ces défis nous pouvons citer l'économie énergétique des nœuds capteurs et aussi protéger les données transmises car les canaux sans fil sont réputés pour être hostiles (ajout de bruits, atténuation du signal, perte de paquets et présence de différents types d'interférences).

Une des sources de consommation les plus gourmandes concerne l'énergie de transmission entre le capteur et la passerelle. Pour cela des attentions particulières doivent être mises en œuvre afin de réduire au maximum la taille des données à transmettre (efficacité de la compression) mais aussi un choix judicieux du moyen de transmission s'impose. Nous nous sommes donc intéressés à la mise en œuvre d'un nœud capteur visuel sans fil (WVSN) à partir d'équipements simples, non onéreux, flexibles et précis. Le choix est vite fait et nous avons donc opté pour un module Arduino Due muni d'une caméra OV7670 et un module de transmission XBee PRO.

Le manuscrit est donc divisé en quatre chapitres. Le premier chapitre pose la problématique liée à ce projet et d'une manière succincte présente un état de l'art sur la surveillance des animaux dans la vie sauvage à partir de réseaux de capteurs sans fil (visuels et/ou scalaires). Le second et troisième chapitres sont consacrés respectivement aux solutions matérielle et logicielle proposées. Quant au dernier chapitre il présente les différents tests menés, les résultats obtenus et une discussion. Enfin, pour terminer une conclusion générale résume les tout ce qui a été effectué durant ce mémoire de fin d'études et propose certaines perspectives jugées pertinentes.

Position du problème

1. Introduction

Dans le monde entier, il existe plus de 10 000 espèces d'oiseaux connues [1]. Si certaines espèces aviaires commencent à décliner, cela peut mettre en lumière de graves problèmes à la fois dans la chaîne alimentaire, dans la sécurité environnementale et dans le réchauffement climatique. À cette fin, la surveillance générale de la faune et de la flore et des oiseaux en particulier peut fournir de nombreuses informations précieuses sur l'environnement, la santé et la productivité des terres, ...etc. La surveillance des oiseaux est pertinente pour identifier l'espèce et évaluer les progrès accomplis dans les efforts de sauvegarde de la biodiversité. Cela a notamment permis de définir des indicateurs, tels que la liste rouge de l'IUCN (International Union for Conservation of Nature) [1, 2, 3, 4]. Cette liste d'espèces menacées, créée en 1964, est largement utilisée comme inventaire de l'état de conservation des espèces biologiques et permet de suivre l'évolution de ces espèces et de surveiller leur habitat naturel. Selon la Liste rouge pour les oiseaux de *BirdLife International*, ces dernières années, le nombre d'espèces d'oiseaux répertoriées dans la catégorie "en danger critique d'extinction" a atteint un niveau sans précédent. En effet, 13% des espèces d'oiseaux du monde sont menacées [2, 3, 4], ce qui en fait la 3ème espèce animale la plus menacée après les amphibiens et les mammifères.

L'Algérie compte plusieurs centaines d'espèces d'oiseaux différentes [4]. Parmi lesquels des oiseaux sédentaires et d'autres migrateurs, dont plus d'une centaine d'espèces d'oiseaux d'eau et autant d'espèces d'oiseaux forestiers. La surveillance de ces oiseaux dans leurs habitats naturels présente plusieurs intérêts. D'une part, disposer d'un indicateur fiable pour évaluer régulièrement l'évolution du nombre d'oiseaux en danger dans une région. D'autre part, les oiseaux peuvent être considérés comme outil de surveillance de l'environnement. Cette surveillance, qui consiste en des observations et des mesures régulières sur ces oiseaux, devrait permettre de prendre des mesures de protection si des modifications défavorables étaient observées. La dégradation de l'habitat, l'activité humaine, les facteurs environnementaux, le changement climatique et les incendies de forêt seraient les menaces les plus importantes qui pèsent sur les espèces d'oiseaux d'une manière générale.

Sur cette optique, le projet que nous proposons tente de fournir un outil technologique, basé sur les réseaux de capteurs sans fil visuels, pour la surveillance de la vie sauvage dans son milieu naturel et les paramètres climatiques et environnementales les plus pertinents qui l'influencent.

2. Etat de l'art

Les premières méthodes de surveillance des populations d'une espèce animale ont été sous forme de collectes manuelles d'informations sur le comportement d'individus par des observateurs sur le terrain [4]. Les objectifs étaient souvent limités à l'étude de l'évolution du nombre de ces oiseaux et leur distribution spatiale sur le site, en observant les individus de manière discontinue (en moyenne 3 fois par mois). Cependant, dans certains cas, cette technique est sujette au problème de la différenciation des individus de la même espèce. Il devient difficile de garantir que deux individus observés à des moments différents sont identiques ou non.

La recherche manuelle d'informations sur le comportement d'individus par le biais d'observateurs de terrain a été la première méthode de collecte de données sur les populations animales. Cette technique est encore largement utilisée. Cependant, dans le cas des oiseaux, l'identification visuelle d'un individu est difficile pour différentes raisons :

- La végétation dense, caractéristique de leur habitat, rend difficile l'accès au site et à la localisation des oiseaux.
- La présence humaine peut perturber les animaux et modifier leur comportement, biaisant ainsi les informations collectées.
- La durée des campagnes d'observation est limitée et dépend de la disponibilité des équipes. Cependant, il est important de disposer de suffisamment de données réparties sur de longues périodes.

Les récentes améliorations apportées aux micro-technologies sont à l'origine de nouveaux types de dispositifs capables de fournir des données spatio-temporelles de manière continue et en temps réel. À partir du milieu des années 2000, la miniaturisation croissante de ces systèmes a permis leur utilisation dans la collecte automatique d'informations sur la nature. En effet, des projets de recherche ont exploité les nouvelles capacités proposées par ce que l'on appelle maintenant "les capteurs portables. Par exemple, la technologie GPS est utilisée depuis plusieurs années pour suivre et identifier la position des animaux. Le problème d'identification des individus ne se pose pas puisque le capteur est porté par un individu et par un seul. Ce dispositif pourrait être utilisé dans des situations où une observation à grande échelle était impossible. Cependant, bien que le système GPS installé sur les animaux soit actuellement la méthode la plus largement utilisée, cela n'était pas faisable dans des contextes tels que celui de certains oiseaux pour les raisons suivantes:

- Il ne fonctionne efficacement que dans les zones ouvertes et ne contient aucun objet pouvant gêner les champs de vision du spectateur. Une opération sous un feuillage dense, qui constitue l'environnement de la sauvagine, n'est pas possible.
- Il est particulièrement coûteux car il est nécessaire de capturer et d'équiper tous les individus.
- Cela consomme beaucoup d'énergie
 - Il a parfois un poids non négligeable, ce qui peut handicaper les mouvements des petits oiseaux.

Ces inconvénients majeurs pour l'étude des oiseaux forestiers et/ou d'eau ont conduit à d'autres dispositifs de collecte qui sont des capteurs fixes [5 – 14]. Des capteurs fixes, tels que des nœuds de capteurs sans fil, ont déjà été utilisés dans diverses applications aussi différentes les unes que les autres. En fonction des périphériques associés aux capteurs, ce type de dispositif permet de: détecter un nombre varié d'interactions à condition qu'elles puissent être identifiées. Contrairement au GPS, ces appareils sont fixes et ne doivent pas être installés sur les individus étudiés. Ils sont généralement installés au sol ou sur des équipements fixés au sol.

Cependant, l'utilisation de ce type de dispositifs nécessite la capacité d'identifier les individus et les interactions avec précision. Typiquement, des algorithmes de traitement du signal et/ou d'analyse de séquence d'images doivent être mis en place pour permettre la reconnaissance des individus ainsi que le type d'interaction. De plus, les données collectées sur les individus ne sont pas régulières dans le temps. Les individus sont détectés seulement quand ils sont dans le champ d'action des capteurs. Par exemple, il est nécessaire de placer un grand nombre de capteurs afin de détecter le maximum de sons et de vocalisations ou encore d'images d'oiseaux et d'animaux à surveiller. Ces nœuds de capteurs sans fil doivent également pouvoir identifier, filtrer et communiquer efficacement les informations qu'ils enregistrent aux systèmes centralisés permettant de collecter et traiter d'une manière plus avancées les données.

Dans le contexte de la collecte d'informations et de la reconnaissance des individus d'une espèce animale, une méthode de collecte idéale doit pouvoir garantir certaines contraintes fonctionnelles majeures telles que le respect du comportement de l'individu et de son habitat et donc la capacité de fonctionner sous un feuillage dense. Il doit également pouvoir couvrir une zone de taille relativement grande, de l'ordre de plusieurs centaines de mètres carrés à plusieurs hectares, et pouvoir détecter efficacement les individus et leurs interactions. Une comparaison des trois familles de techniques (méthodes manuelles, équipements mobiles et capteurs fixes) par rapport à un certain nombre de critères fonctionnels définis pour l'étude. Parmi ces critères nous pouvons citer : durée de l'étude,

détérioration de l'habitat, obligation de capture du spécimen, coût financier, changement de comportement, fonction sous un feuillage dense ...etc. Ces critères, incontournables pour un tel projet, excluent l'utilisation des appareils mobiles. En effet, bien que les appareils mobiles permettent d'effectuer la collecte sur une longue période et ne contribuent pas à la détérioration de l'habitat, ils nécessitent la capture et l'équipement de ces animaux, ce qui peut entraîner des coûts de mise en œuvre élevés et handicaper également les animaux dans leurs déplacements. De plus, le fonctionnement en feuillage dense n'est pas garanti, ce qui peut affecter la qualité des données collectées dans le cadre d'études menées sur la sauvagine.

3. Réseaux de capteurs sans fil RCSF (WSN)

Les réseaux de capteurs sans fil WSN sont des réseaux spontanés constitués de nœuds déployés en grand nombre en vue de rassembler et de transmettre des données vers un ou plusieurs points de collecte (sink). L'un des principaux buts des WSN est de surveiller un environnement limité en termes de ressources. Pour y parvenir plusieurs objectifs doivent être mis en œuvre. Ces objectifs comprennent la synchronisation entre les nœuds, le choix de protocoles de communication, la localisation, la gestion de la topologie, l'agrégation et le stockage des données, l'optimisation de la consommation énergétique. Les capteurs sont conçus pour fonctionner d'une manière autonome durant des mois voire des années. Ainsi, la capacité énergétique des capteurs doit être utilisée efficacement afin de maximiser la durée de vie du réseau.

Parmi les exigences de ce projet la gestion de la consommation de l'énergie. Rappelons que l'énergie consommée par un nœud capteur est due essentiellement à ses différentes opérations suivantes en l'occurrence :

- **Énergie de détection d'évènements :** C'est l'énergie consommée par un nœud capteur lors de l'activation de son unité d'acquisition et de collecte de données. Le coût de cette énergie dépend du type spécifique du capteur (image, son, température, humidité, vitesse du vent, etc.) et des tâches (mesures, conditionnement des signaux et conversion analogique-numérique, etc.) qui lui sont assignées.
- **Energie de traitement :** C'est l'énergie consommée par un nœud lors de l'activation de son unité de traitement de données (opérations, lecture/écriture en mémoire). L'énergie de traitement se divise en deux parties : l'énergie de commutation et l'énergie de fuite. L'énergie de commutation est déterminée par la tension d'alimentation et la capacité totale commutée au niveau logiciel (en exécutant un logiciel). Par contre l'énergie de fuite correspond à l'énergie consommée lorsque

l'unité de calcul n'effectue aucun traitement. En général, l'énergie de traitement est faible par rapport à celle nécessaire pour la communication.

- **Energie de communication :** C'est l'énergie consommée par un nœud lors de l'activation de son unité de transmission. Des expérimentations ont montré que c'est l'étape la plus consommatrice en énergie. Il a été démontré que la transmission d'un bit d'information peut consommer autant que l'exécution de quelques milliers d'instructions.

Cette contrainte énergétique sera un des critères les plus pertinents dans le choix du protocole de transmission à adopter et aussi dans l'algorithme de traitement qui doit être implémenté dans chaque nœud. Ceci est d'autant plus important lorsqu'il s'agit de capteurs visuels. En effet, l'acquisition d'images nécessite généralement une compression basée souvent sur des transformations linéaires orthogonales qui consomment beaucoup d'énergie due à leurs complexités calculatoires. Ceci nous oblige donc à réduire le coût calculatoire de ces algorithmes de compression en utilisant entre autres des transformations entières ou même binaires. D'autre part, l'image contient souvent des objets d'un certain intérêt (exemple l'animal à surveiller) au détriment d'autres détails et objets de moindre intérêt (exemple l'arrière plan). Il est donc intéressant de faire appel à des techniques de compression basées sur des régions d'intérêt.

Il y a également d'autres exigences à satisfaire lors du déploiement d'un WSN, parmi lesquelles :

4. la distance moyenne entre les nœuds de capteur voisins :
5. le modèle de déploiement (modèle aléatoire, régulier ...etc).

La distance moyenne entre les nœuds de capteur voisins est importante pour les performances d'un réseau appliqué à la surveillance d'animaux dans leurs habitats naturels (à l'aide du captage des sons et/ou images d'une espèce animale), à la mesure de données climatiques et à la détection des feux de forêt. Cette distance inter-nœuds est d'une importance cruciale et elle joue un rôle pertinent dans :

- la localisation et la reconnaissance de l'animal à surveiller
- le délai de détection d'un départ de feu par exemple,
- ...etc

A cet effet, la distance moyenne entre les nœuds de capteurs voisins doit être raccourcie. Toutefois, cela nécessite le déploiement de davantage de nœuds de capteurs dans une zone donnée et peut entraîner des problèmes de collision de canaux. Il s'agit donc d'un compromis entre la réduction du temps de détection et la probabilité de collision.

La reconnaissance automatique des oiseaux dans leurs habitats naturels en utilisant des WSN est soumise à de nombreuses contraintes et difficultés, par exemple:

- Les enregistrements sur le terrain sont fortement corrompus en raison de bruits environnementaux (si on utilise les sons d'oiseaux), très volumineux (s'il s'agit d'images ou vidéo)
- Les oiseaux sont généralement très éloignés de l'enregistreur (ou du nœud du capteur), de sorte que données acquises (sons ou images) sont relativement faibles ou petits.
- Interférences avec d'autres données appartenant à d'autres espèces d'oiseaux ou animaux. Un nœud capteur peut capter simultanément des données de plusieurs oiseaux de différentes espèces.
- Les canaux utilisés par les réseaux de capteurs sans fil sont souvent hostiles d'où le problème de la qualité de service qui doit être assurée.

Les WSN comme beaucoup d'autres systèmes communicants embarqués peuvent être amenés à interagir avec d'autres équipements et appareils connectés voire même des opérateurs de télécommunications afin d'atteindre des objectifs communs.

4. Conclusion

Dans ce premier chapitre, nous avons commencé par poser la problématique qui nous a poussés à entamer un tel projet et présenter succinctement un état de l'art sur les solutions qui ont déjà été proposées. Il s'agit de la surveillance des certains oiseaux dans leur habita naturel. L'objectif principal est de développer un système, utilisant un réseau de capteurs sans fil, pour surveiller et identifier différentes espèces d'oiseaux des zones humides du nord-est de l'Algérie. Ce système doit pouvoir rassembler autant d'informations que possible pour aider à mettre en place un plan de protection des espèces d'oiseaux les plus vulnérables. La mise en œuvre de ce système de surveillance soulève plusieurs défis. Premièrement, le système doit être simple, à faible coût, flexible, il doit être en mesure de capter les images ciblées et de pouvoir les compresser. Deuxièmement, les WSN présentent de nombreuses limitations telles que la faiblesse relative de leur unité de traitement, la petite taille de la mémoire, les problèmes de communication et surtout la faible durée de vie de leurs batteries. Néanmoins, d'autres contraintes doivent également être résolues, telles que la distance entre la position de l'oiseau et le nœud du capteur qui peut atténuer la puissance du signal capturé et, par conséquent, affecter les performances du système de reconnaissance.

Solution adoptée

1. Introduction

Il s'agit de proposer une solution matérielle sous forme d'un "testbed" représentant un ou des nœuds de capteurs sans fil sur lesquels une solution globale sera implémentée, optimisée et évaluée. Cette solution est de mettre en œuvre des nœuds de capteurs sans fil ou "mote" (également appelé en anglais "node") éventuellement hétérogènes capables de fonctionner d'une manière hiérarchique, de supporter un algorithme de compression d'images et/ou vidéo, d'effectuer éventuellement un traitement préliminaire et bien évidemment les transmettre à travers un réseau WMSN (Wireless Multimedia Sensor Network) dit aussi WVSN (Wireless Visual Sensor Network).

2. RCSF comme solution

Au cours des deux dernières décennies, les réseaux de capteurs sans fil (WSN) ont de plus en plus intéressé les communautés scientifiques et professionnelles [4, 13, 14]. En effet, les perspectives d'applications et les domaines d'utilisation sont en perpétuelle augmentation [4, 14]: militaire, environnement, santé et sécurité. Un réseau WSN est un réseau de nœuds de capteurs qui détectent, contrôlent et transmettent informations pertinentes sur un environnement donné. Les réseaux de capteurs sans fil offrent de nombreux avantages, par exemple:

- Leur facilité de déploiement,
- Ils sont utilisés dans des environnements hostiles et difficiles d'accès,
- Ils sont évolutifs,
- Ils utilisent plusieurs types de technologies de communication sans fil

Les WSN présente également des inconvénients, notamment:

- Leur durée de vie est limitée en raison de leurs autonomies énergétiques,
- Ils ont des capacités de stockage et de calcul limitées,
- Ils ont une bande passante limitée,
- Leurs canaux de transmission sont souvent hostiles
- Peut être sujet à des attaques de sécurité.

Malgré les inconvénients ci-dessus, les WSN sont largement utilisés dans de nombreux domaines et représentent aujourd'hui, entre autres, le fer de lance de l'Internet des objets (IoT : Internet of Things) [14]. Un nœud de capteur, également appelé "mote", est un nœud d'un réseau de capteurs sans fil capable d'effectuer certains traitements, d'acquérir des informations sensorielles et de communiquer avec les autres nœuds du même réseau.

Généralement, un nœud de capteur est composé d'une :

- unité de traitement souvent un microcontrôleur,
- une unité de communication (émetteur-récepteur). La transmission utilisée est souvent une radiofréquence (RF). Cependant, certains nœuds de capteurs possèdent des unités de communication optique (laser) ou infrarouge. La technologie infrarouge, tout comme les lasers, a une capacité de diffusion limitée. En RF, le WSN utilise souvent les bandes radio industrielles, scientifiques et médicales (ISM) telles que les fréquences 173, 433, 868, 915 MHz; et 2,4 GHz
- d'une alimentation
- d'un ou plusieurs capteurs.

3. Prototype adopté

Il existe de nombreux petits modules embarqués, à l'instar du MICAZ, Telos, Waspote, cyclops, imote2, arduino due, raspberry PI3 ...etc, dotés de capteurs scalaires (température, humidité, polluants atmosphériques de type NOx,etc) et aussi de capteurs d'images (comme par exemple uCamIII, Caméra IMB400, Camera Module V2-8, CMUcam, ARducam ...etc) nous permettant de construire la solution que l'on adoptera pour des applications de surveillance de l'environnement.

3.1 Module Arduino Due

L'un des plus simples nœuds capteur de point de vue conception et programmation tout en assurant une grande flexibilité est la solution utilisant un arduino Due, basé sur un Atmel ARM Cortex SAM3X8E cadencé à 84 MHz sous 32 bits, surtout si nous nous limitons à une application basée sur l'acquisition d'images et non de vidéo. En effet, ce type de module dispose d'une technologie largement suffisante, en particulier la taille de sa mémoire SRAM (Static Random Access Memory) de 96koctets pour manipuler des images en format brut (RAW) d'une certaine résolution, par exemple de 128x128 pixels jusqu'à 640 x 480 codés sur 8 bits chacun.

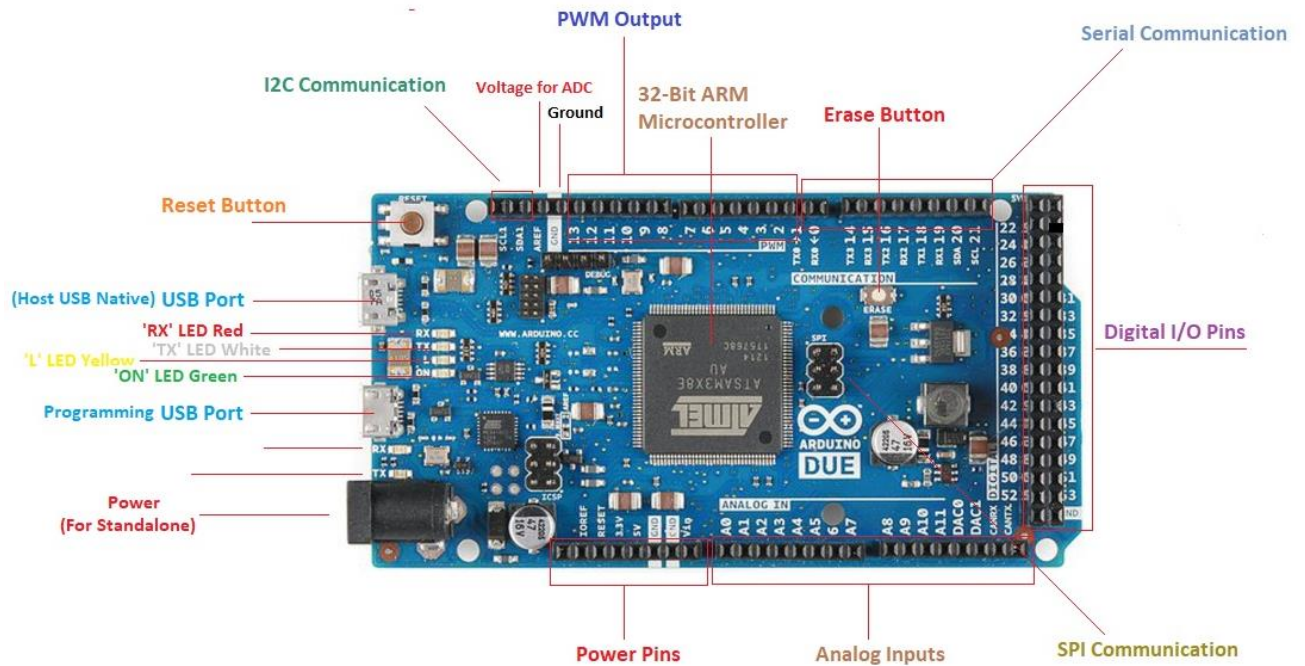


Figure 1 : Module Arduino Due

3.2 Module caméra OV7670 [15]

L'OV7670 est une petite caméra (capteur d'images) compatible avec arduino [15]. Elle a besoin d'une faible tension de fonctionnement et une faible consommation électrique. Elle offre toutes les fonctions de la caméra mono-puce VGA de 30 images/s et du processor d'image. Parmi ses caractéristiques nous pouvons citer :

- Alimentation : 2.8 VDC
- Résolution : 640 x 480
- Interface standard SCCB compatible avec l'interface I2C
- Haute sensibilité pour les applications de faible luminosité,
- Contrôle automatique de l'exposition, AGC,
- Contrôle de la qualité de l'image, y compris la saturation des couleurs, la teinte, le gamma, la netteté ...etc.
- ...etc



Pin	Function
3V3	Positive power supply pin.
GND	Ground pin.
SDIOC	Serial Clock
SDIOD	Serial Data
VSYNC	Vertical Sync
HREF	Horizontal Sync
PCLK	Pixel clock output
XCLK	System clock input
D7	Video parallel output bit 7
D6	Video parallel output bit 6
D5	Video parallel output bit 5
D4	Video parallel output bit 4
D3	Video parallel output bit 3
D2	Video parallel output bit 2
D1	Video parallel output bit 1
D0	Video parallel output bit 0
RESET	Reset (active low) pin.
PWDN	Power down (active high) pin.

Figure 2 : Caméra OV7670 & brochage [15]

L'interfaçage de la caméra OV7670 avec un module arduino Due peut être schématisé par la figure suivante. Une photographie de la réalisation utilisant cet interfaçage est représenté sur la figure 4.

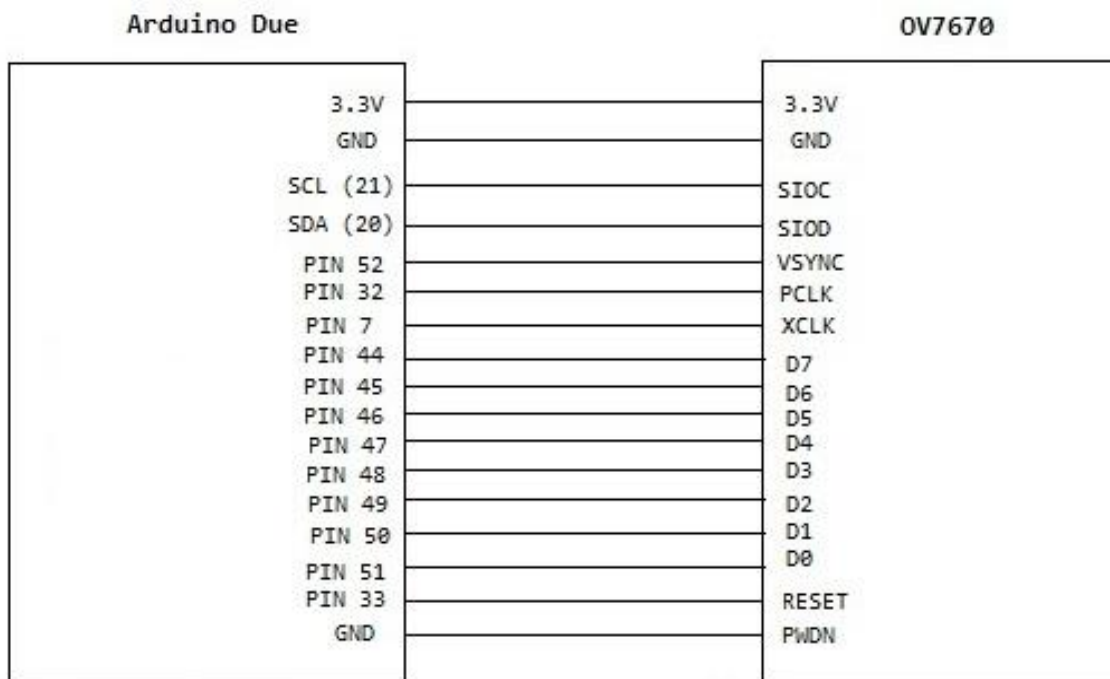


Figure 3 : L'interfaçage de la caméra OV7670 avec un module arduino Due [15]

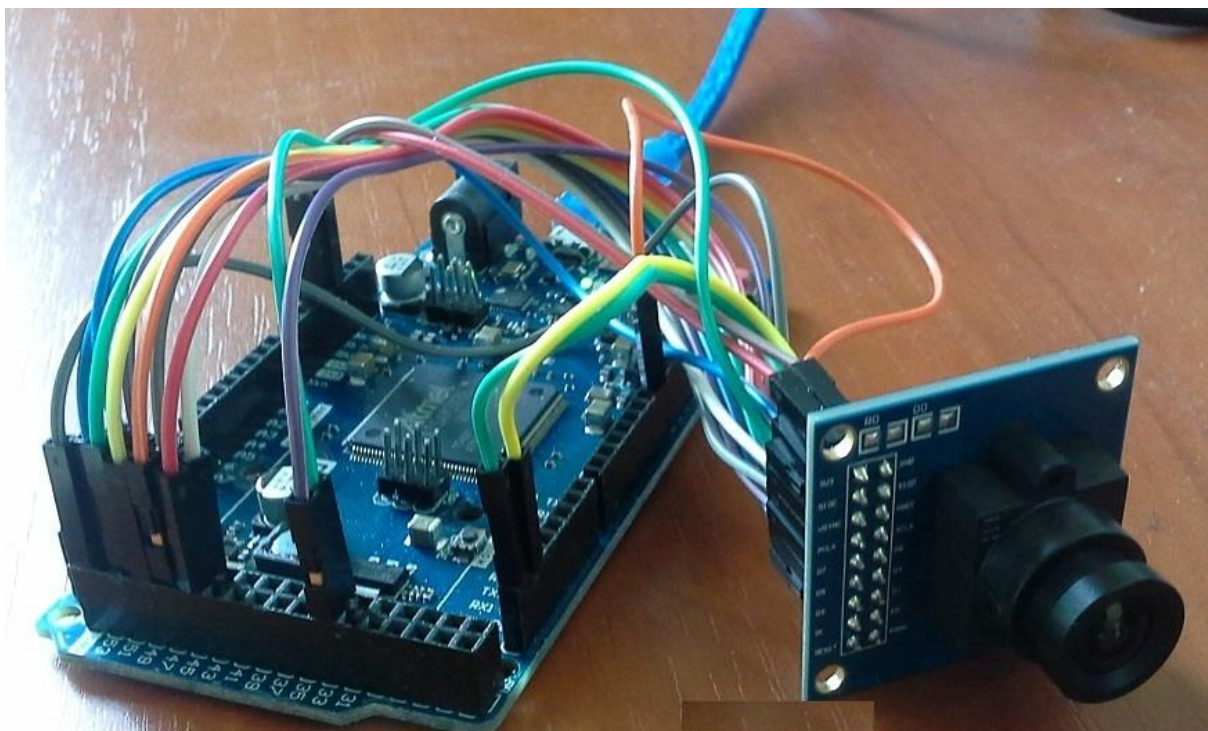


Figure 4 : Photographie de la caméra OV7670 interfacée avec un module Arduino Due

3.3 Module de communication XBee [16]

Le module XBee, sous forme de plusieurs catégories, est un microcontrôleur de Digi muni d'un émetteur-récepteur sans fil 2,4 GHz pour communiquer avec un autre module XBee selon le protocole 802.15.4. Ces modules sont capables de communiquer avec plus d'un module XBee, ce qui nous permet de créer un réseau de modules à condition qu'ils se trouvent à portée. Le module XBee standard a une puissance d'émission de 1mW avec une portée de 10 à 100 mètres (série 1 et 2) et le XBee Pro dispose d'une puissance d'émission de 60 mW avec une portée pouvant dépasser 1000 mètres (plus d'un km). Selon le d'antennes utilisée (filaire, chip, U.FL, RPSMA) on peut augmenter davantage sa portée.

- Alimentation: 3,3 Vcc (215 mA)
- Puissance: 60 mW (18 dBm) 10 mW EIRP
- Débit HF: 250 kbps
- Débit série: jusqu'à 115,2 kbps
- Portée moyenne:
 - 100 m en intérieur
 - jusqu'à 1000 m en extérieur
- Sensibilité en réception: -100 dBm
- Fréquence: 2,4 GHz
- Dimensions: 28 x 25 x 10 mm
- T° de service: -40 à +85 °C
- Poids: 3 g

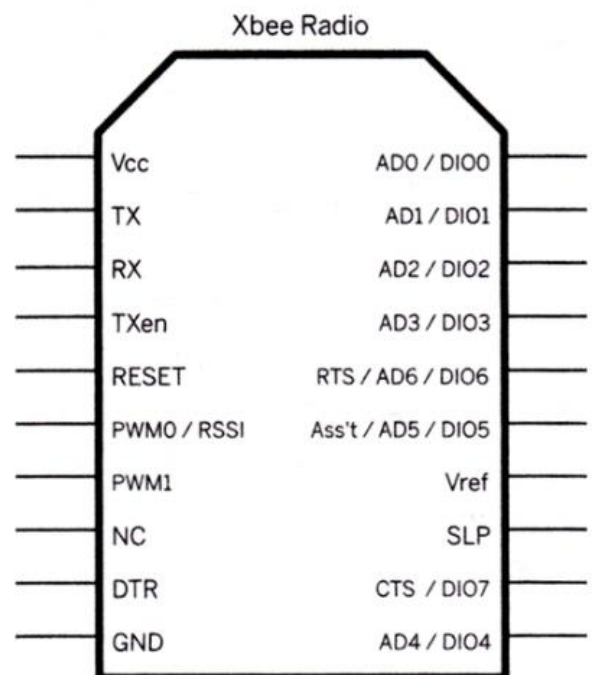


Figure 5 : Module XBee PRO S1 et brochage [16]

Les modules XBee utilisent les **protocoles 802.15.4 et Zigbee**. Ceci, permet de réaliser différents montages, d'une liaison série RS232 classique à un réseau maillé (mesh). Ces modules fonctionnent selon un modèle à quatre couches (figure ci-dessous). Le protocole IEEE 802.15.4, utilisé par XBee, est utilisé pour les couche 1 et couche 2 et permet la communication entre les modules. Quant au protocole Zigbee il est destiné à la couche 3, crée la hiérarchie du réseau et configure d'autres paramètres comme l'association, l'authentification, l'encodage, le routage ou encore les services de la couche d'application autrement appelés les clusters.

Comme dans toutes les communications sans fil, le 802.15.4 est un canal hostile pour les transmissions sans fil. En effet, des interférences ou du bruit peuvent venir perturber les données. Le 802.15.4 utilisent le Direct Sequence Spread Spectrum (DSSS) pour moduler les informations avant leur envoi à la couche physique. Chaque bit d'information est multiplié en 4 signaux différents. Ce processus utilise plus de bande passante mais une densité spectrale de puissance plus basse ce qui facilite la détection et le décodage du message côté récepteur.

Le 802.15.4 est un protocole à basse consommation. Il utilise des cycles courts, ce qui permet au transmetteur d'être mis en veille la plupart du temps. Les tâches de réception et d'envoi peuvent aussi être paramétrées de façon à ce qu'elles consomment peu d'énergie.

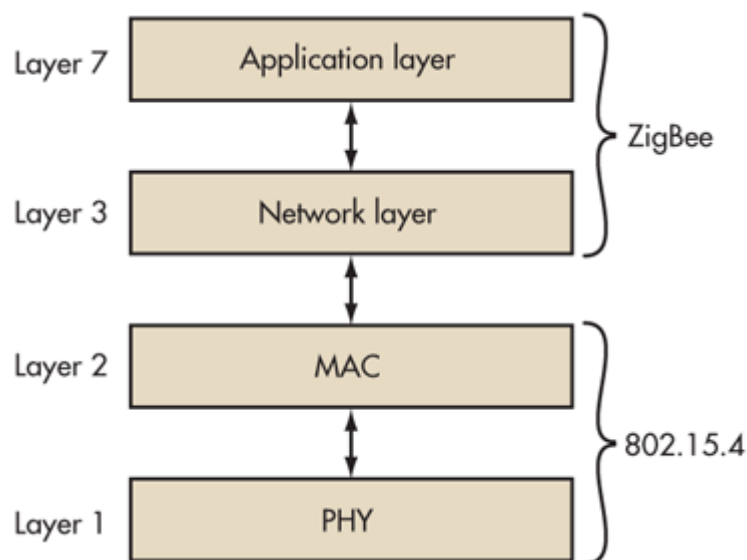


Fig. :

Figure 6 : Modèle en couche du fonctionnement de XBee [16]

Il existe trois types de nœuds dans un protocole Zigbee :

- Coordinateur : module "maître" qui contrôle le réseau.
- Routeurs : ils routent les informations envoyées par les terminaux.
- Terminaux : les capteurs qui prennent les informations sur l'environnement.

Le protocole Zigbee crée des topologies en étoile :

- Les terminaux sont connectés à un routeur ou à un coordinateur.
- Les routeurs peuvent être connectés entre eux ou avec un coordinateur.
- Les routeurs et les coordinateurs ne peuvent pas être mis en veille. Ils doivent conserver dans leur buffer les paquets qui sont destinés aux terminaux.
- Les terminaux peuvent être mis en veille.

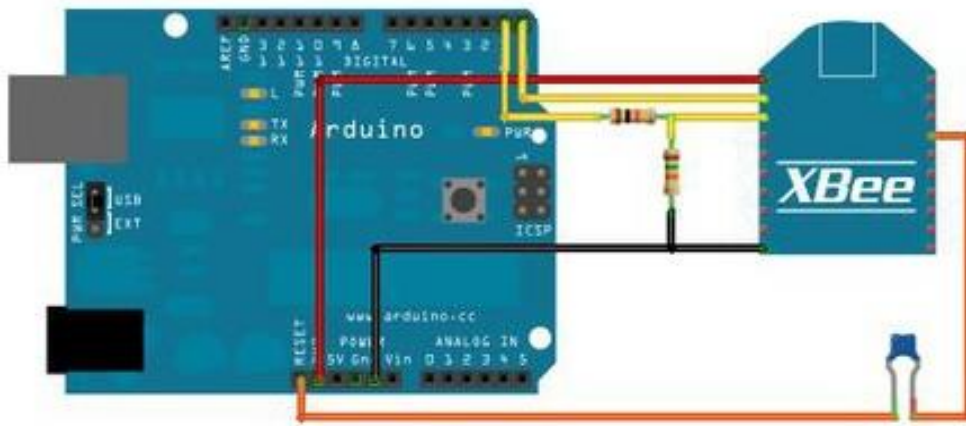


Figure 7 : Exemple d'interfaçage en un arduino et un XBee



Figure 8: Exemple d'interfaçage en un PC et un XBee

4 Conclusion

Dans ce chapitre nous nous sommes focalisés sur la proposition d'une solution surtout côté matériel (hardware). A ce propos nous avons définis les points les plus importants à atteindre:

- Choisir des équipements simples, à faible coût, flexibles pouvant supporter des traitements d'images à l'instar de la compression.
- Adopter le moyen de transmission adéquat. Pour les réseaux de capteurs sans fil, la transmission utilisée est souvent une radiofréquence (RF) à travers les bandes radio industrielles, scientifiques et médicales (ISM) telles que les fréquences 173, 433, 868, 915 MHz; et 2,4 GHz. Nous avons été tentés donc par des transmissions radios basés sur des protocoles de type 802.15.4 de 2.4GHz à faible portée et à faible débit. Mais nous pouvons aussi penser à adopter des protocoles de communication LoRa (Long Range).

Algorithmes de traitement proposés

1. Introduction

Après avoir choisi et mis en œuvre la partie matérielle de notre solution, nous devons maintenant nous intéresser à la technique à implémenter. Les points les plus importants à atteindre avec la partie software peuvent être résumés en ce qui suit :

- Réduire la consommation d'énergie du contrôleur et de l'unité de transmission du nœud de capteur (par exemple en optimisant l'algorithme adopté de point de vue complexité calculatoire et/ou mettre en veille les parties du nœud du capteur quand elles ne sont pas utilisées ...etc),
- Adapter les niveaux d'alimentation dans le nœud du capteur en fonction de la charge de travail,
- Les nœuds de capteurs sans fil mis en place doit pouvoir travailler d'une manière hiérarchique,
- ...etc.

Nous constatons donc que le défi à relever n'est pas tout simplement de mettre en œuvre un programme de capture d'images, de compression et de transmission mais aussi et surtout réduire la consommation énergétique des nœud de capteurs sans fil et d'assurer une qualité d'images reconstruite optimale en sachant que ce type de canaux sans fil sont très hostiles.

2. Capture d'images

Pour capturer des images à l'aide de la petite caméra OV7670, il y'a quelques étapes à suivre dans le programme de contrôle à implémenter sur arduino.

- Nous devons d'abord commencer par ajuster la vitesse d'horloge de la caméra. Cet exemple s'applique au numéro 7 sur 10.5MHz.

```
int32_t mask_PWM_pin = digitalPinToBitMask (7);
REG_PMC_PCER1 = 1 << 4; L'horloge d'activation pour // le contrôleur
PWM
REG_PIOC_PDR |= mask_PWM_pin; Les fonctions périphériques activate
Le // pour labroche (désactive toutes les fonctionnalités PIO)
REG_PIOC_ABSR |= mask_PWM_pin; Choisissez l' option périphérique //
B
REG_PWM_CLK = 0; // choisissez la fréquence d'horloge, 0 -> plein de
MCLK comme la référence du 84MHz de
REG_PWM_CMR6 = 0 << 9; // horloge de sélection de polarité et pour
le canal de PWM (de pin7) -> (CPOL = 0)
REG_PWM_CPRD6 = 8; // initialiser période PWM -> T = valeur / 84MHz
(valeur: jusqu'à 16 bits), valeur = 8 -> 10.5MHz
```

```
REG_PWM_CDTY6 = 4; // initialiser cycle de service, REG_PWM_CPRD6 /
valeur = rapport cyclique, pour 8/4 = 50%
REG_PWM_ENA = 1 << 6; // activer PWM sur le canal PWM (broche 7 =
PWML6)
```

- Ensuite, nous devons configurer l'appareil. pour des valeurs du registre compatibles avec le conditionnement de la bibliothèque utilisée.

```
void écriture (uint8_t REGID, octet Regdat) {
  Wire.beginTransaction (adresse);
  Wire.write (REGID & 0x00FF);
  Wire.write (Regdat & 0x00FF);

  si (Wire.endTransmission (true)) {
    Serial.print ( "erreur d'écriture reg");
    Serial.println (REGID);
  }
  retard (20);
}
```

- Réglage de la UART sera également standard. On optera d'un baud de 19200 pour être conforme avec le débit autorisé par le module de communication, utilisé à savoir XBee

```
Serial.begin (19200); // Parce que ç'est la vitesse maximale de
notre XBee
```

- utiliser la fonction avec un accès direct aux registres du microcontrôleur.

```
inline int uart_putchar statique (const uint8_t c) {
  while ((UART-> UART_SR & UART_SR_TXRDY)!);
  UART-> UART_THR = c;
  return 0;
}
```

- Nous pouvons utiliser le tampon pour envoyer les images.

```
captureImg vide statique (uint16_t wg, uint16_t hg) {
  uint16_t y, x;
  Serial.println ( "* RDY *");
  while ((REG_PIOB_PDSR & (1 << 21))!); // attendre haute
  while ((REG_PIOB_PDSR & (1 << 21))); // attendre faible
```

```

y = hg;
while (y -) {
    x = wg;
    while (x -) {
        while ((REG_PIOD_PDSR & (1 << 10))); // attendre faible
        myImage [y] [x] = (REG_PIOC_PDSR & 0xFF000) >> 12;
        while ((REG_PIOD_PDSR & (1 << 10))!); // attendre haute
        while ((REG_PIOD_PDSR & (1 << 10))); // attendre faible
        while ((REG_PIOD_PDSR & (1 << 10))!); // attendre haute
    }
}

pour (y = 0; y <hg; y ++) {
    for (x = 0; x <wg; x ++) {
        uart_putchar (myImage [y] [x]);
    }
}
}

```

3. Compression d'images

Malgré leurs avantages, les WSN se heurtent à une contrainte très importante en raison de la limitation de leur autonomie énergétique. En effet, ces nœuds de capteurs sans fil sont alimentés par des batteries dont la durée de vie dépend essentiellement de la quantité de données à transmettre ainsi que de la complexité informatique des algorithmes utilisés. Par conséquent, la solution la plus efficace pour surmonter ces problèmes critiques réside dans les techniques de compression d'images qui présentent une complexité de calcul et des débits binaires faibles. Dans l'algorithme JPEG (figure 9), norme de référence, la transformée en cosinus discrète (DCT) (1), découverte en 1974 [17-19] et appartenant à la classe des transformées unitaires sinusoïdales, consomme à elle seule plus de 60% de l'énergie totale du codeur [20 – 24].

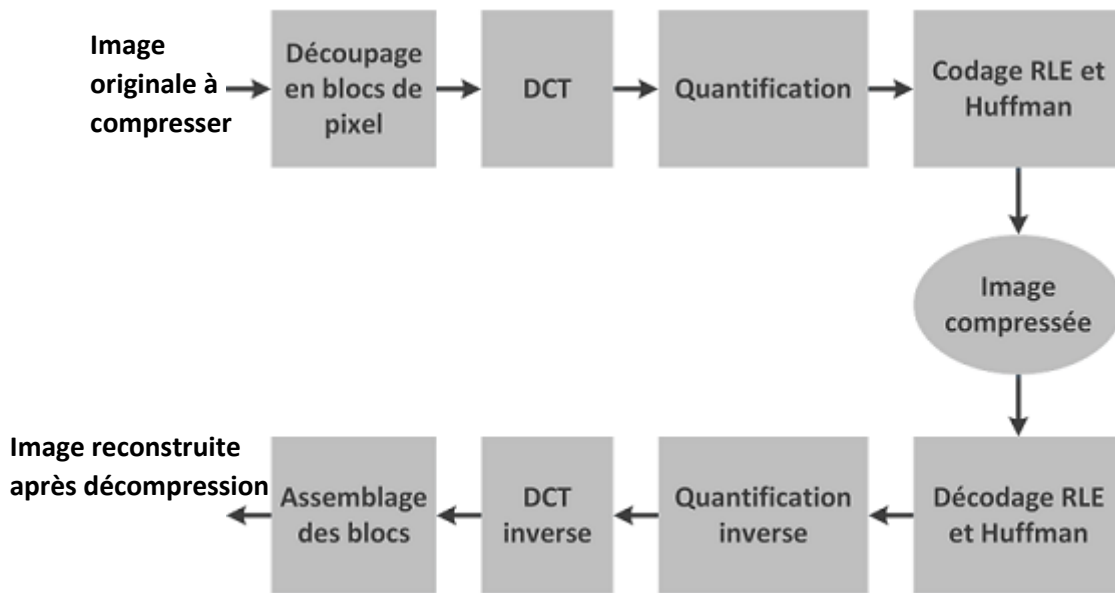


Figure 9 : Schéma de principe d'un codec JPEG

Ainsi, au cours des dernières décennies, de nombreux efforts ont été faits pour proposer des algorithmes rapides de la DCT exacte, permettant de gagner du temps, de la vitesse et de l'énergie, en diminuant encore sa complexité algorithmique.

$$X_k = A_k \times \sum_{n=0}^7 x_n \cos\left(\frac{\pi}{8} \times \left(n + \frac{1}{2}\right) \times k\right) \text{ for } k = 0, 1, \dots, N-1 \quad (1)$$

où $A_0 = \sqrt{\frac{1}{8}}$ et $A_k = \sqrt{\frac{2}{8}}$ ailleurs

avec les N sorties X_k de la transformée 1D réelle DCT-II pour N entrées x_n sont données.

Sous forme matricielle, la DCT-II 1D s'écrit :

$$[X_{8 \times 1}] = [C_{8 \times 8}^{II}] \times [x_{8 \times 1}] \quad (2)$$

Où $[X_{8 \times 1}]$ et $[x_{8 \times 1}]$ sont les vecteurs sorties et entrées de taille 8×1 . $[C_{8 \times 8}^{II}]$ est la matrice transformée 1D DCT-II donnée par :

$$[C_{8 \times 8}^{II}]_{n,k} = A_k \times \cos\left(\frac{\pi}{8} \times \left(n + \frac{1}{2}\right) \times k\right) \text{ pour } k, n = 0, 1, \dots, 7 \quad (3)$$

La matrice $[C_{8 \times 8}^{II}]$ dépend essentiellement de 7 valeurs réelles signées différentes notées $c_0=0.35355$, $c_1=0.49039$, $c_2=0.41573$, $c_3=0.27779$, $c_4=0.09755$, $c_5=0.46194$ et $c_6=0.19134$. En utilisant ces 7 valeurs la matrice $[C_{8 \times 8}^{II}]$ peut prendre la forme suivante [17-19]:

$$C_{8 \times 8}^{II} = \begin{bmatrix} c_0 & c_0 & c_0 & c_0 & c_0 & c_0 & c_0 & c_0 \\ c_1 & c_2 & c_3 & c_4 & -c_4 & -c_3 & -c_2 & -c_1 \\ c_5 & c_6 & -c_6 & -c_5 & -c_5 & -c_6 & c_6 & c_5 \\ c_2 & -c_4 & -c_1 & -c_3 & c_3 & c_1 & c_4 & -c_2 \\ c_0 & -c_0 & -c_0 & c_0 & c_0 & -c_0 & -c_0 & c_0 \\ c_3 & -c_1 & c_4 & c_2 & -c_2 & -c_4 & c_1 & -c_3 \\ c_6 & -c_5 & c_5 & -c_6 & -c_6 & c_5 & -c_5 & c_6 \\ c_4 & -c_3 & c_2 & -c_1 & c_1 & -c_2 & c_3 & -c_4 \end{bmatrix} \quad (4)$$

Compte tenu de la propriété de séparabilité, la DCT-II 2D est obtenue en effectuant la DCT-II 1D le long des N lignes de ce bloc, puis le long de ses N colonnes. Cette DCT-II 2D, sous forme de matrice, s'écrit:

$$[X_{8 \times 8}] = [C_{8 \times 8}^{II}] \times [x_{8 \times 8}] \times [C_{8 \times 8}^{II}] \quad (5)$$

La 2D DCT-II inverse est donnée par :

$$[x_{8 \times 8}] = [C_{8 \times 8}^{II}] \times [X_{8 \times 8}] \times [C_{8 \times 8}^{II}] \quad (6)$$

Mais l'utilisation directe de la transformation DCT-II exacte 1D nécessite un grand nombre d'opérations, ce qui n'est pas souhaitable pour les applications temps réel et à faible coût de calcul. Afin de réduire encore davantage le coût de calcul d'une telle transformation, utilisée par exemple dans une chaîne de compression d'images, nous utilisons souvent des valeurs entières ou même des approximations DCT. En effet, les valeurs $c_0, c_1 \dots c_6$ de la matrice, au lieu d'être des valeurs réelles, peuvent être remplacées par des entiers quelconques, où même remplacées par des 0 et des 1. Le calcul de ces approximations DCT nécessite généralement un nombre d'opérations beaucoup plus réduit par rapport à la DCT réelle, au détriment d'une dégradation supplémentaire. Les approximations DCT doivent préserver les principales propriétés mathématiques de la DCT d'origine, telles que la linéarité, l'orthogonalité et le compactage énergétique. Il est également recommandé que la matrice de

l'approximation DCT conserve la structure générale de la matrice, comme indiqué dans (16), et que ses éléments (0 et 1) vérifient autant que possible les conditions. L'approximation de $[C_{8 \times 8}^H]$, que l'on notera $[IC_{8 \times 8}^H]$, doit aussi vérifier les mêmes propriétés que la version réelle à savoir l'orthogonalité et la structure générale [17 – 19].

$$IC_{8 \times 8}^H = D \times \begin{bmatrix} I_0 & I_0 & I_0 & I_0 & I_0 & I_0 & I_0 & I_0 \\ I_1 & I_2 & I_3 & I_4 & -I_4 & -I_3 & -I_2 & -I_1 \\ I_5 & I_6 & -I_6 & -I_5 & -I_5 & -I_6 & I_6 & I_5 \\ I_2 & -I_4 & -I_1 & -I_3 & I_3 & I_1 & I_4 & -I_2 \\ I_0 & -I_0 & -I_0 & I_0 & I_0 & -I_0 & -I_0 & I_0 \\ I_3 & -I_1 & I_4 & I_2 & -I_2 & -I_4 & I_1 & -I_3 \\ I_6 & -I_5 & I_5 & -I_6 & -I_6 & I_5 & -I_5 & I_6 \\ I_4 & -I_3 & I_2 & -I_1 & I_1 & -I_2 & I_3 & -I_4 \end{bmatrix} \quad (7)$$

Où D est une matrice diagonale et I_0, \dots, I_6 sont 7 valeurs (des 0 et 1 signés).

L'approximation DCT arrondie (RDCT), introduite dans [20] et associée à la matrice de transformation $C_{8 \times 8}$, offre une performance compétitive par rapport aux approximations bien connues de l'état de la technique. Il est obtenu en appliquant l'opérateur round à la matrice DCT standard. La RDCT a un faible coût de calcul car elle est formée uniquement par $\{0, 1, -1\}$ dans ses entrées.

$$C_{8 \times 8} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & -1 & -1 & -1 \\ 1 & 0 & 0 & -1 & -1 & 0 & 0 & 1 \\ 1 & 0 & -1 & -1 & 1 & 1 & 0 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & 0 & 1 & -1 & 0 & 1 & -1 \\ 0 & -1 & 1 & 0 & 0 & 1 & -1 & 0 \\ 0 & -1 & 1 & -1 & 1 & -1 & 1 & -0 \end{bmatrix} \quad (8)$$

On sait que presque toute l'énergie d'un bloc d'image 8×8 est concentrée dans les 4×4 premiers coefficients DCT (correspondant aux basses fréquences). Ainsi, dans une chaîne JPEG, nous pouvons simplement calculer ces 4×4 coefficients DCT pour chaque bloc 8×8 de l'image. Cela entraînera une réduction significative du nombre d'opérations. Ainsi, les DCT-II 2D avant et inverse sont donnés respectivement par:

$$[X_{8 \times 8}] = [C_{4 \times 8}^H] \times [x_{8 \times 8}] \times [C_{8 \times 4}^H]^t \quad (9)$$

$$[x_{8 \times 8}] = [C_{8 \times 4}^{II}]^t \times [X_{4 \times 4}] \times [C_{4 \times 8}^{II}]$$

(10)

La matrice $[C_{N \times 4}^{II}]$ prend alors la structure suivante :

$$IC_{8 \times 4}^{II} = D \times \begin{bmatrix} I_0 & I_0 & I_0 & I_0 & I_0 & I_0 & I_0 & I_0 \\ I_1 & I_2 & I_3 & I_4 & -I_4 & -I_3 & -I_2 & -I_1 \\ I_5 & I_6 & -I_6 & -I_5 & -I_5 & -I_6 & I_6 & I_5 \\ I_2 & -I_4 & -I_1 & -I_3 & I_3 & I_1 & I_4 & -I_2 \end{bmatrix} \quad (11)$$

Ainsi, l'expression de la transformation matricielle semi-orthogonale obtenue $T_{4 \times 8}^*$ est la suivante:

$$T^* = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & -1 & -1 \\ 1 & 0 & 0 & -1 & -1 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \quad (12)$$

Ainsi, la DCT directe et inverse seront données par :

$$Y_{4 \times 4} = D^* \times (T_{4 \times 8}^* \times X_{8 \times 8} \times T_{8 \times 4}^{*t}) \times D^* \quad (13)$$

$$X_{8 \times 8} = T_{8 \times 4}^{*t} \times (D^* \times Y_{4 \times 4} \times D^*) \times T_{4 \times 8}^* \quad (14)$$

Où D^* est une matrice diagonale d'échelle, donnée par:

$$D^* = \text{diag}\left(\frac{1}{\sqrt{8}}; \frac{1}{2}; \frac{1}{2}; \frac{1}{\sqrt{2}}\right) \quad (15)$$

Notons que cette matrice T respecte la structure générale d'une matrice de la DCT. Cette approximation DCT 2D est calculée en fonction de la propriété de séparabilité, en utilisant une transformation 1D le long des lignes, puis le long des colonnes d'un bloc d'image 8×8 pixels. L'approximation P-RDCT 1D de chaque ligne donnera 4 coefficients. Ensuite, sur ce résultat intermédiaire, une approximation PDCT 1D est effectuée sur chaque colonne. Le résultat final est une approximation P-RDCT 2D de taille 4 × 4. Ainsi, la transformation 1D d'une seule ligne du bloc d'image 8 × 8 à l'aide de la transformation T peut être représentée

par le graphe de flux de signal (SFG) présenté à la suivante. Les flèches en pointillés représentent les multiplications par -1.

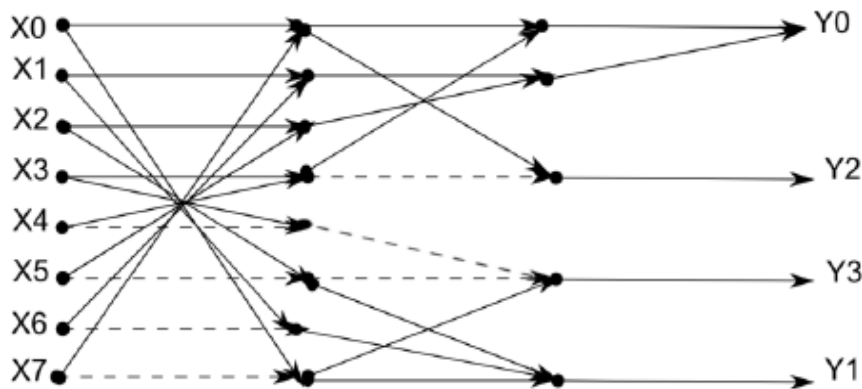


Figure 10: Le graphe de flux de signal (SFG) de la transformation ‘Pruned RDCT’ [20]

Le nombre d’opérations A_{1D} (seulement des additions) est donc 16 pour calculer cette approximation DCT-1D. L’expression nous permettant de calculer le nombre d’opérations (additions) $A_{2D}(4)$ pour un bloc de 8x8 est donnée par [20-24]:

$$A_{2D}(4) = (12) \times A_{1D} = 12 \times 16 = 192 \text{ additions}$$

4. Transmission d’images

La transmission des images est effectuée grâce au module XBee. Il s’agit d’une transmission série classique. Cependant, les modules XBee doivent être préalablement configurés en utilisant XCTU [25] qui est une application multiplate-forme. Elle a été créée pour permettre aux développeurs d’interagir avec les modules Digi RF via une interface graphique simple et facile à utiliser. Elle comprend de nouveaux outils qui facilitent la configuration, et le test des modules XBee RF. Le processus de mise à jour du microprogramme restaure de manière transparente les paramètres du module, la gestion automatique des modifications du mode et de la vitesse de transmission. Deux consoles API et AT spécifiques ont été conçues de toutes pièces pour communiquer avec ces périphériques radio, dans cette application nous avons utiliser le mode AT.

5. Réception des images au niveau du PC

Plusieurs possibilités s’offrent à nous pour pouvoir réceptionner les données numériques transmises par le module Arduino via XBee. Il s’agit de bien choisir le logiciel à utiliser au

niveau du PC. Nous avons opté pour le logiciel Eclipse Java. Eclipse est un environnement de développement (IDE) historiquement destiné au langage Java, même si grâce à un système de plugins il peut également être utilisé avec d'autres langages de programmation, dont le C/C++ et le PHP. Eclipse nécessite une machine virtuelle Java (JRE) pour fonctionner. Mais pour compiler du code Java, un kit de développement (JDK) est indispensable. Dans notre cas arduino transmettre les donnée via le module XBee ver le PC qui va lire les donnée à l'aide de ce logicielle avec le programme correcte qui va récupérer les image sou format .BMP.

Les images envoyées par l'arduino Due via XBee, vont être reçues au niveau du PC configuré en tant qu'un client FrameGrabber, en utilisant deux (02) programmes à savoir "SimpleRead.java" et "BMP.java". Pour le développement et l'exécution de ces 2 programmes on a utilisé Eclipse.

6. Conclusion

Ce chapitre représente l'essentiel de notre projet. Il a été consacré pour présenter et expliquer les algorithmes adoptés et qui doivent être implémentés au niveau du module Arduino Due. Il s'agit principalement :

- Le programme de captage d'images qui va contrôler le module OV7670. Les images acquises seront sous forme brute (RAW). Autrement dit sans aucune compression
- Une compression d'images utilisant l'algorithme de référence JPEG basée sur une transformée DCT approximée et élaguée (Pruned). Ceci dans le but de réduire au maximum sa complexité algorithmique et ainsi prolonger la durée de vie du nœud capteur en aménageant sa batterie d'alimentation.
- Une transmission des données compressées en utilisant un module XBee Pro S1
- Enfin une réception sur un PC en utilisant un programme adéquat sous Eclipse.

Le choix du module arduino due a été dicté par plusieurs contraintes. D'abord nous avons pris comme condition l'utilisation d'un module simple, à moindre prix, efficace et pouvant supporter des traitements numériques assez avancés comme la compression d'images JPEG. Ceci a été effectué avec une certaine réussite que nous allons pouvoir vérifier et évaluer dans le chapitre suivant.

Tests et résultats

1. Introduction

Après avoir réalisé et mis en œuvre les solutions matérielle et logicielle de notre prototype "testbed" nous nous sommes intéressés aux tests de ses fonctionnalités et à l'évaluation de ces performances. A ce propos nous avons commencé par vérifier son bon fonctionnement en captant, transmettant et réceptionnant plusieurs images sous différentes contraintes (plusieurs exemples d'images, portées différentes, débits différents ...etc). Ensuite, nous nous sommes intéressés à l'évaluation des performances de la compression, de la transmission et de la consommation énergétique.

2. Exemples d'expérimentations réalisées

2.1 PSNR et temps de compression/transmission en fonction du bitrate

Nous avons réalisé différents tests pour trois (03) images différentes. En effet, nous avons commencé par capturer trois (03) images différentes avec notre module arduino/caméra/XBee représentant trois vues différentes que l'on a noté Image_Exemple1, Image_Exemple2 et Image_Exemple3. Pour chaque cas d'image capturée nous avons effectué cinq (05) tests différents représentant cinq taux de compressions différents (ou cinq débits différents) (Tableaux 1, 2 et 3). Dans chacun de ces 05 cas nous avons effectué la transmission et ensuite mesuré la qualité de l'image reconstruite en termes de PSNR (Peak Signal to Noise Ratio) et calculé le temps de transmission. Ainsi, nous avons commencé par représenter l'image originale et les 05 images reconstruites correspondant aux cinq bitrates pour les trois exemples utilisés (figures 11, 14 et 17). Nous avons ensuite tracé les courbes représentant le PSNR en fonction du bitrate pour les trois exemples d'images (figures 12, 15 et 18). Enfin nous avons aussi tracé les courbes représentant les temps de codage/transmission pour les trois cas d'images capturées (figures 13, 16 et 19). A partir de ces tests plusieurs constatations ont pu être soulevées. D'abord les PSNR calculés, pour des débits relativement faibles, sont assez élevés ce qui se traduit par des images de qualités acceptables. Bien sûr certaines différences entre ces trois exemples d'images peuvent être expliquées par le contenu des images. En effet, certaines images contiennent plus de détails aigus, ou hautes-fréquences, par rapport à d'autres vont certainement subir un peu plus de dégradation. Mais ces PSNR dépendent uniquement des effets de la compression/décompression et non des effets des canaux de transmission réputés pour être hostiles à la transmission. En effet, pour évaluer efficacement les effets des canaux de transmission nous devons nous mettre dans des cas plus réalistes avec beaucoup de capteurs (un véritable réseau) et dans un environnement réel. Ceci doit générer de pertes de paquets qui auront pour conséquence une qualité de service QoS

moindre. A propos des temps de transmission il apparait clairement qu'il s'agit d'un temps trop élevé qu'il faudrait inévitablement le réduire. Ce temps de transmission est dû essentiellement à la liaison série utilisée par les modules XBee dont la vitesse a été configurée assez lente. La solution est de remplacer ces modules XBee par d'autres modules XBee plus rapide ou encore par des liaisons IP ou LoRa.

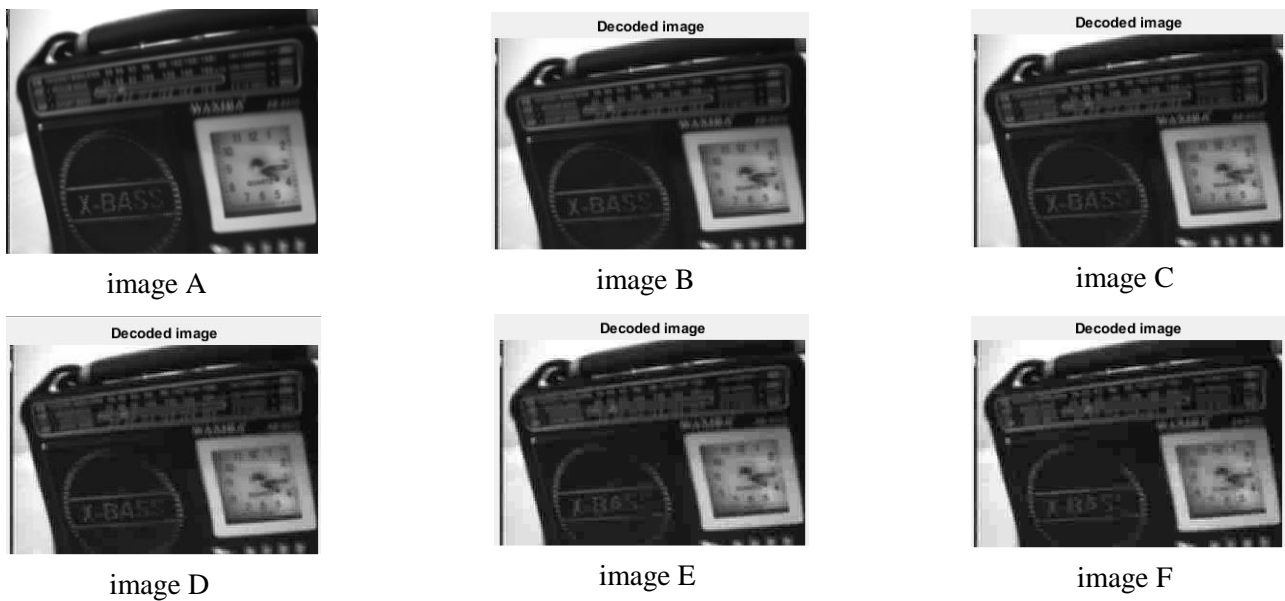


Figure 11 : Résultat de la compression et transmission Image_Exemple1

(A) : image originale { les images compressées (B), (C), (D), (E) et (F)

	Image B	Image C	Image D	Image E	Image F
bit Rate	+ 0.55 bpp	+0.39 bpp	+0.32 bpp	+ 0.27 bpp	+ 0.24 bpp
PSNR	+29.57 dB	+29.24 dB	+28.84 dB	+28.43 dB	+28.04 dB
temps	22002 ms	15531 ms	12593 ms	10797 ms	9535 ms

Tableau 1 : PSNR vs Bitrate et temps de calcul pour Image_Exemple1

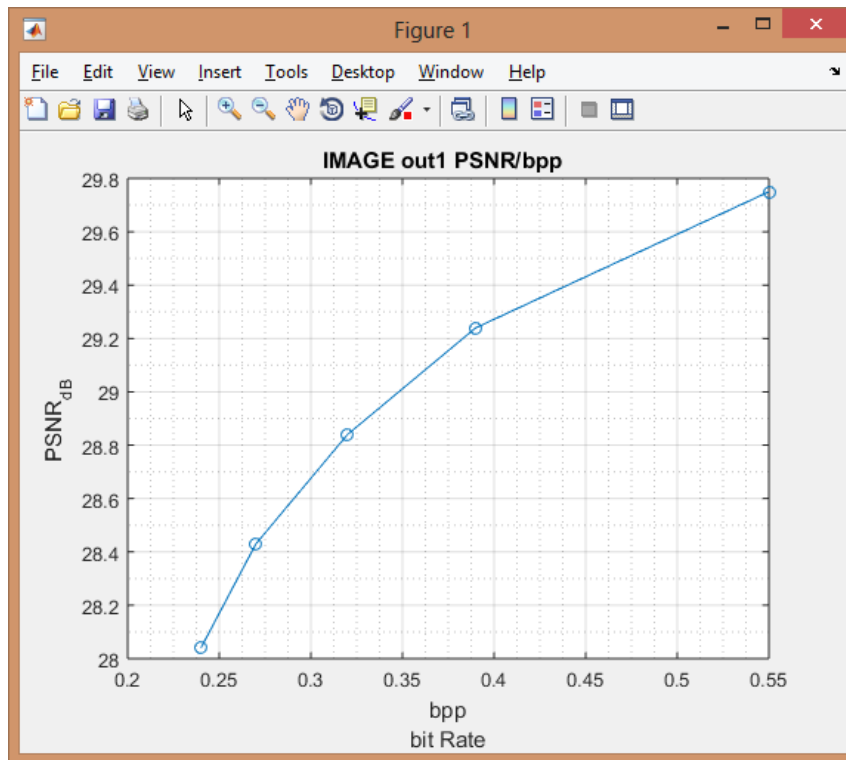


Figure 12: Courbe PSNR/ bit Rate bpp pour Image_Exemple1

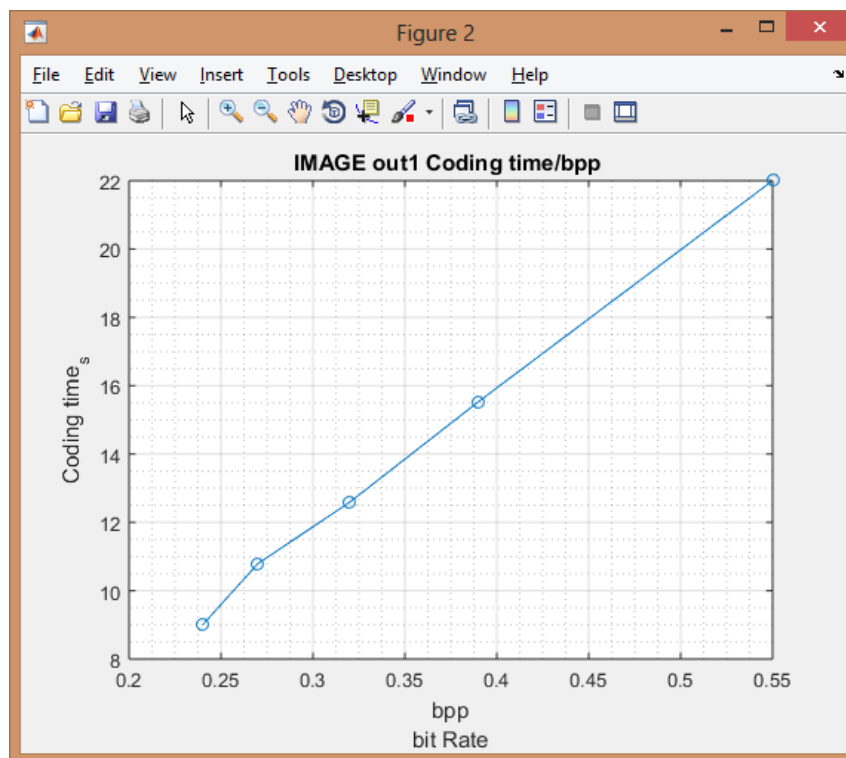


Figure 13 : Courbe Temps de codage et transmission / bit Rate bpp pour Image_Exemple1

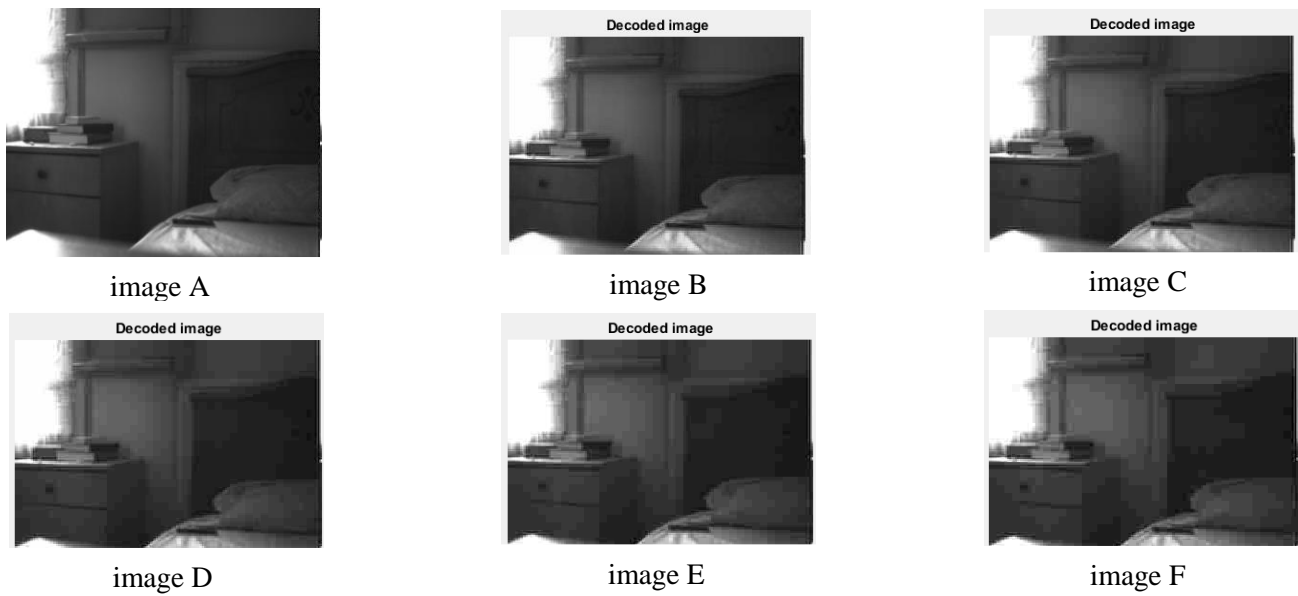


Figure 14 : Résultat de la compression et transmission Image_Exemple2
 (A) : image originale { les images compressées (B), (C), (D), (E) et (F)

	Image B	Image C	Image D	Image E	Image F
bit Rate	+ 0.36 bpp	+ 0.26 bpp	+ 0.21 bpp	+ 0.19 bpp	+ 0.17 bpp
PSNR	+30.68 dB	+30.42 dB	+30.11 dB	+29.78 dB	+29.39 dB
temps	14426 ms	10193 ms	8365ms	7338ms	6684 ms

Tableau 2 : PSNR vs Bitrate et temps de calcul pour Image_Exemple2

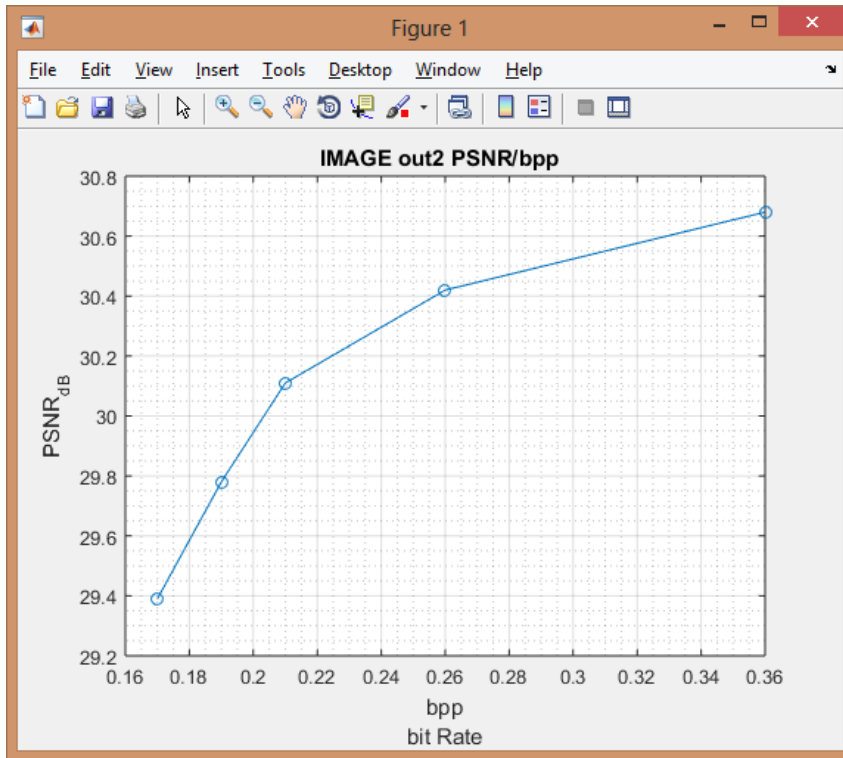


Figure 15: Courbe PSNR/ bit Rate bpp pour Image_Exemple2

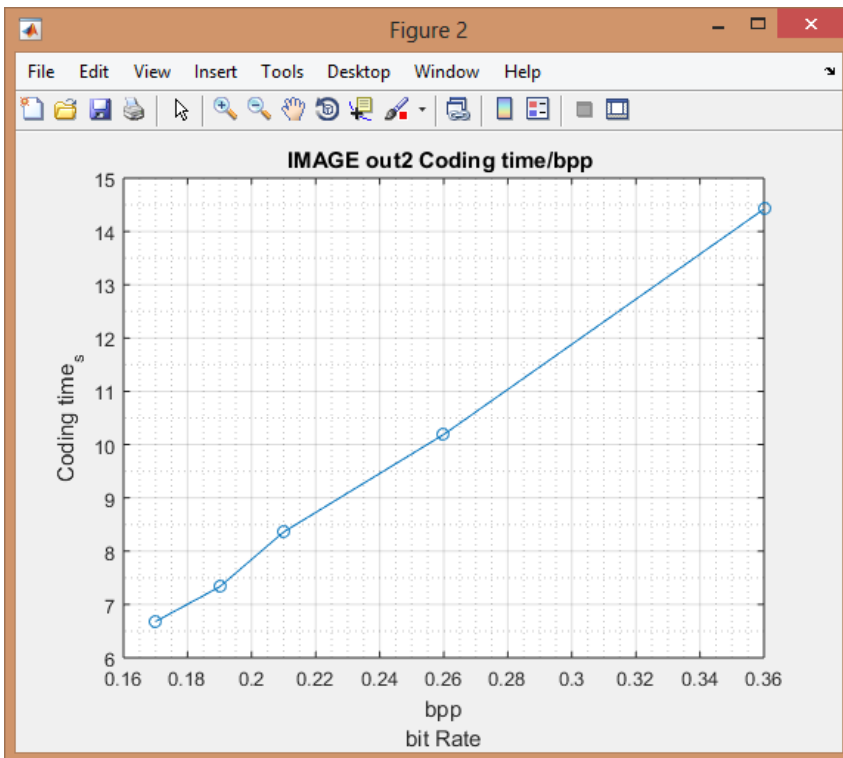


Figure 16 : Courbe Temps de codage et transmission / bit Rate bpp pour Image_Exemple2

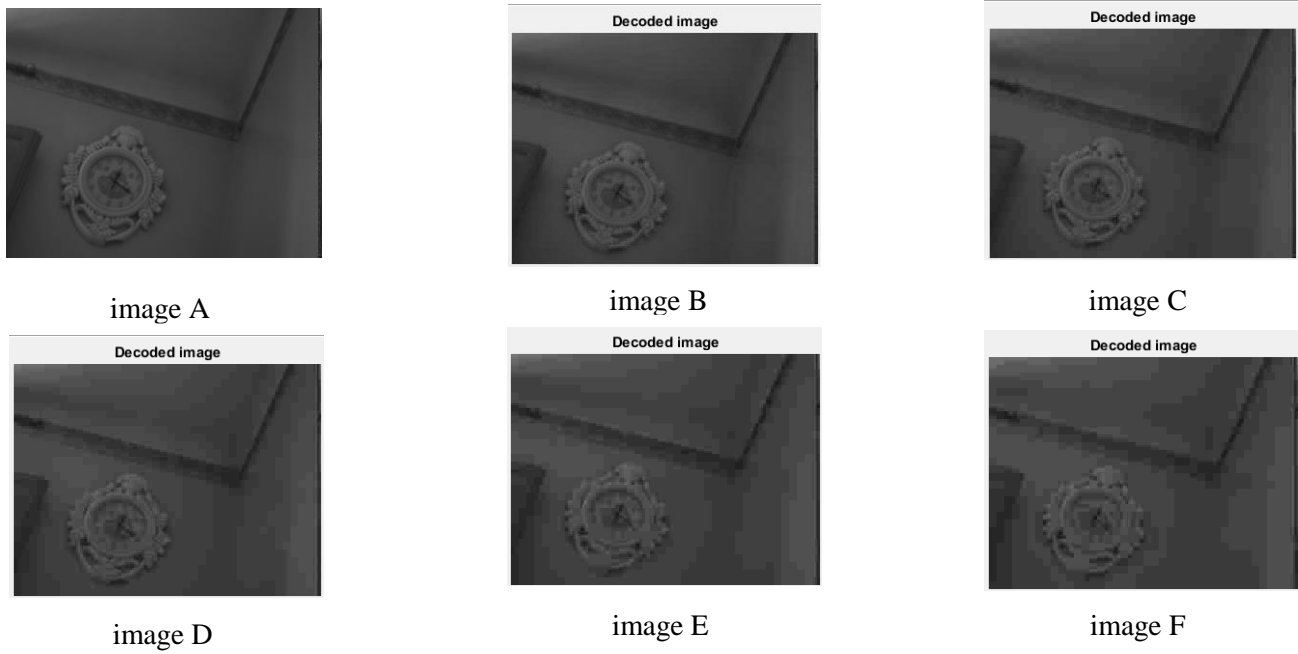


Figure 17 : Résultat de la compression et transmission Image_Exemple3

(A) : image originale { les images compressées (B), (C), (D), (E) et (F)

	Image B	Image C	Image D	Image E	Image F
bit Rate	+ 0.26 bpp	+ 0.18 bpp	+ 0.15 bpp	+ 0.13 bpp	+ 0.13 bpp
PSNR	+39.81 dB	+38.37 dB	+37.02 dB	+35.94 dB	+34.80 dB
temps	10353 ms	7148 ms	5963 ms	5305 ms	4932 ms

Tableau 3 : PSNR vs Bitrate et temps de calcul pour Image_Exemple3

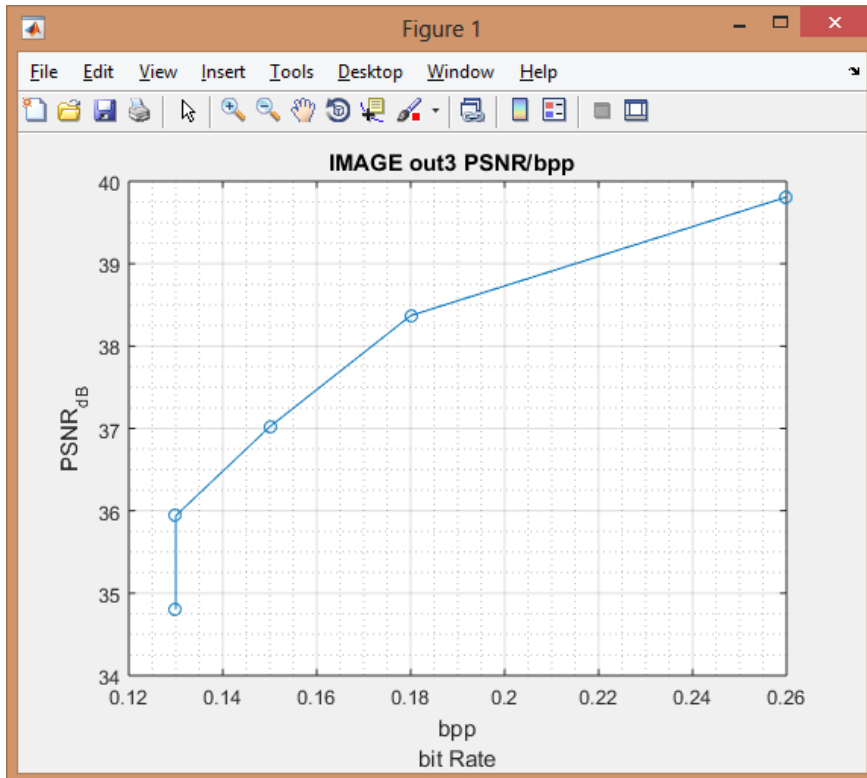


Figure 18: Courbe PSNR/ bit Rate bpp pour Image_Exemple3

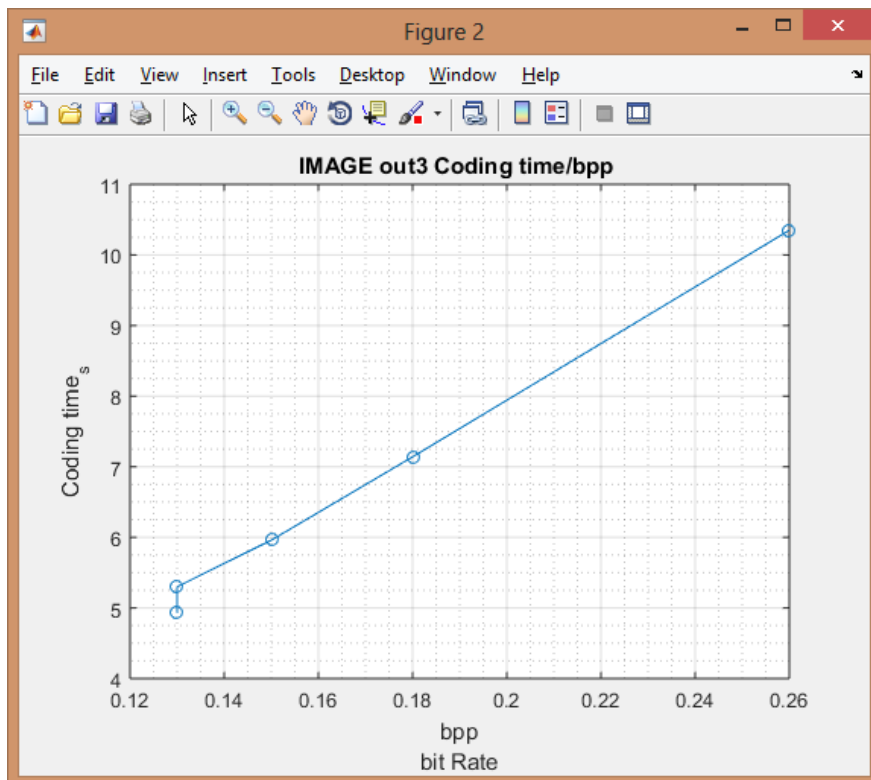


Figure 19 : Courbe Temps de codage et transmission / bit Rate bpp pour Image_Exemple3

2.2 Consommation énergétique due à la compression:

L'un des challenges les plus pertinents à lever est l'économie énergétique dans les nœuds capteurs. En effet, ces petits modules embarqués ont une certaine autonomie énergétique, assurée par des batteries, qui va réduire ou augmenter leurs durées de vie. Cette économie énergétique peut être réalisée sur les différentes étapes de fonctionnement du nœud capteurs comme le traitement et surtout la transmission.

2.2.1 Algorithme de compression à faible coût calculatoire

En ce qui concerne le traitement il s'agit principalement de l'algorithme de compression que nous avons implanté. Ainsi, la réduction significative des opérations arithmétiques minimise directement la consommation d'énergie des nœuds de capteur, ce qui prolonge la durée de vie du réseau WWSN [20 - 24]. Ainsi, pour évaluer la consommation d'énergie de la transformation adoptée, nous utilisons le modèle qui a été adopté dans plusieurs travaux [21 - 24]; il est spécialement conçu pour évaluer la consommation d'énergie des nœuds WSN. Pour une image d'entrée de taille $M \times N$, l'énergie consommée par la transformation proposée peut être donnée par:

$$E_{M \times N} = \frac{M \times N}{8 \times 8} \times N_{Add} \times \varepsilon_{Add} \quad (16)$$

Où, $N_{Add}=192$ le nombre d'additions nécessaire pour un bloc 8×8 et $\varepsilon_{Add}=3.3nJ$ [20 - 24] est la consommation de l'énergie pour une seule opération d'addition. On notera que ce modèle de consommation concerne plus précisément un microcontrôleur Atmel Atmeg128, utilisé entre autres par les nœuds capteurs Mica2. Notre nœud capteur est bien sûr différent mais nous allons adopter ce modèle pour notre cas car les microcontrôleurs sont souvent proches de point de vue consommation. On déduit donc pour un bloc de 8×8 pixels :

$$E_{8 \times 8} = N_{Add} \times \varepsilon_{Add} = 192 \times 3.3 = 0,63 \mu J \quad (17)$$

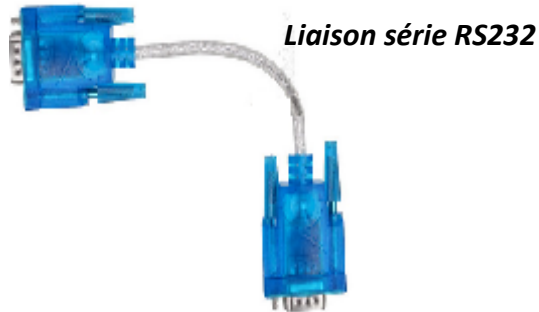
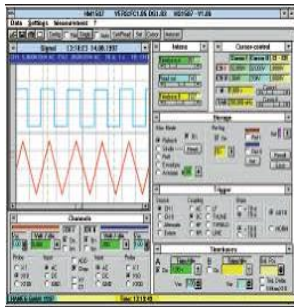
Alors que la consommation énergétique pour le calcul d'une DCT 2 D de tous les blocs d'une image de

$$E_{M \times N} = \frac{640 \times 480}{8 \times 8} \times N_{Add} \times \varepsilon_{Add} = \frac{640 \times 480}{8 \times 8} \times 192 \times 3.3 = 3,4 mJ \quad (18)$$

2.2.2 Consommation énergétique due à la transmission:

Il s'agit de la consommation énergétique la plus importante dans un nœud capteur. Plusieurs travaux, depuis plusieurs années, se sont intéressés à l'étude, l'évaluation et l'économie de ce type de consommation énergétique jugée toujours la plus pertinente dans l'augmentation ou la réduction de la durée de vie d'un réseau de capteurs sans fil. Plusieurs solutions ont été proposées aussi bien matérielles mais aussi et surtout logicielles sous forme de protocoles de communication. Dans cette partie expérimentale de notre travail nous nous sommes juste intéressés à l'évaluation et la mesure de cette consommation due à la transmission. Comme notre moyen de transmission est un module XBee nous avons commencé par lire ses caractéristiques, dont la consommation en courant, dans son "datasheet". Cette indication nous semble trop vague et ne prend pas en considération tous les cas de figures liés aux différentes utilisations possibles. Alors, pour avoir une idée claire et précise sur cette consommation énergétique nous avons proposé de monter une expérience relativement simple (figure 20) mais efficace permettant de mesurer instantanément cette consommation énergétique en utilisant entre autres un oscilloscope numérique. Le schéma de principe illustrant la méthode utilisée pour mesurer et enregistrer le courant consommé lors de la transmission. Ainsi, à l'aide d'une résistance de précision de 1 Ohm, mis entre l'alimentation et le module XBee, on arrive à convertir le courant consommé instantanément en tension. Cette dernière est visualisée sur un oscilloscope numérique de type Hameg. L'oscilloscope est interfacé avec le PC via une simple liaison série asynchrone RS232. Un logiciel dédié à cet oscilloscope et installé dans le PC va nous permettre de capter et enregistrer ce courant consommé pendant une durée de temps déterminé. En effet, à l'aide de ce logiciel, dont l'interface visualisée sur le PC est représentée sur la figure 21, nous pouvons alors configurer l'oscilloscope numérique (calibre, base de temps, tracesetc) et aussi lancer l'acquisition.

Dès lors, nous avons réalisé plusieurs tests pour pouvoir mesurer le courant consommé réellement par le module XBee durant les transmissions des images. Nous avons alors programmer l'arduino pour que des envoies d'images soient périodiques suivis par des temps de "silence". On remarque alors que durant l'envoi le courant consommé, traduit à l'aide d'une petite résistance de 10hm en tension de quelques centaines de mV, fait de brusques sauts très significatifs. Ainsi, sur les courbes des figures 22 et 23, pour des bases de temps différentes, le courant consommé atteints quelques 250mA pendant une durée de l'ordre quelques ms (approximativement 4ms) représentant le temps de l'émission d'un paquet de l'image codée ou compressée. A partir de là nous pouvons calculer le courant consommé pour toute l'image.



RF receiver
Xbee S2



PC avec le logiciel HM407_2

Oscilloscope
Numérique
HAMEG HM 407_2

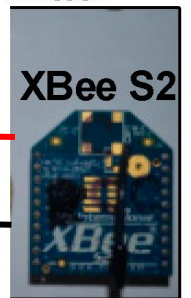


Alimentation en courant continu

Power supply



RF Transmitter



I_{active}

$1 \Omega, 1\%$

I_{active}

Module Xbee pour la
transmission d'images

Figure 20 : Schéma de principe utilisé pour la mesure expérimentale du courant consommé

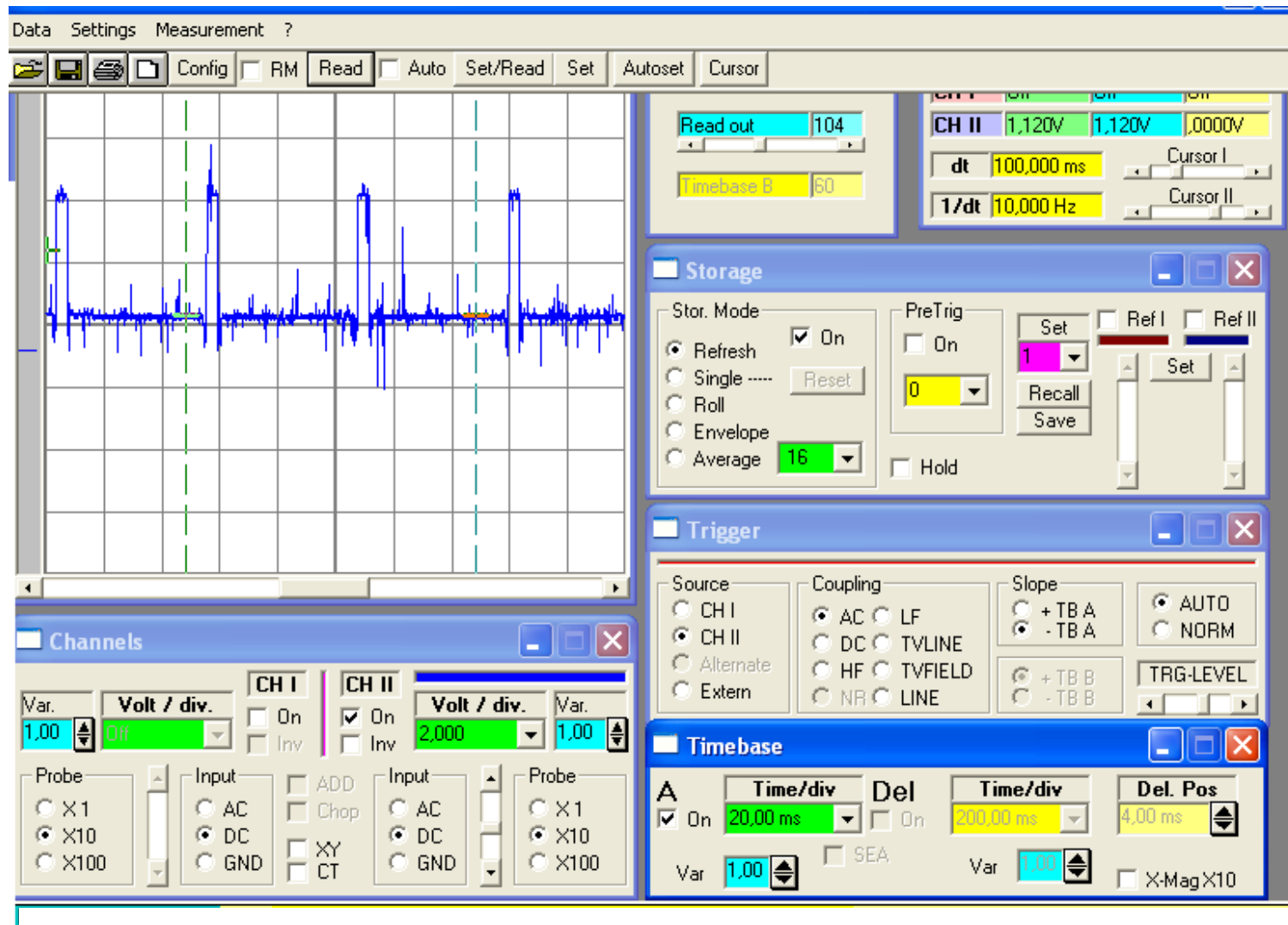


Figure 21 : Interface graphique du logiciel de configuration et contrôle de l'oscilloscope

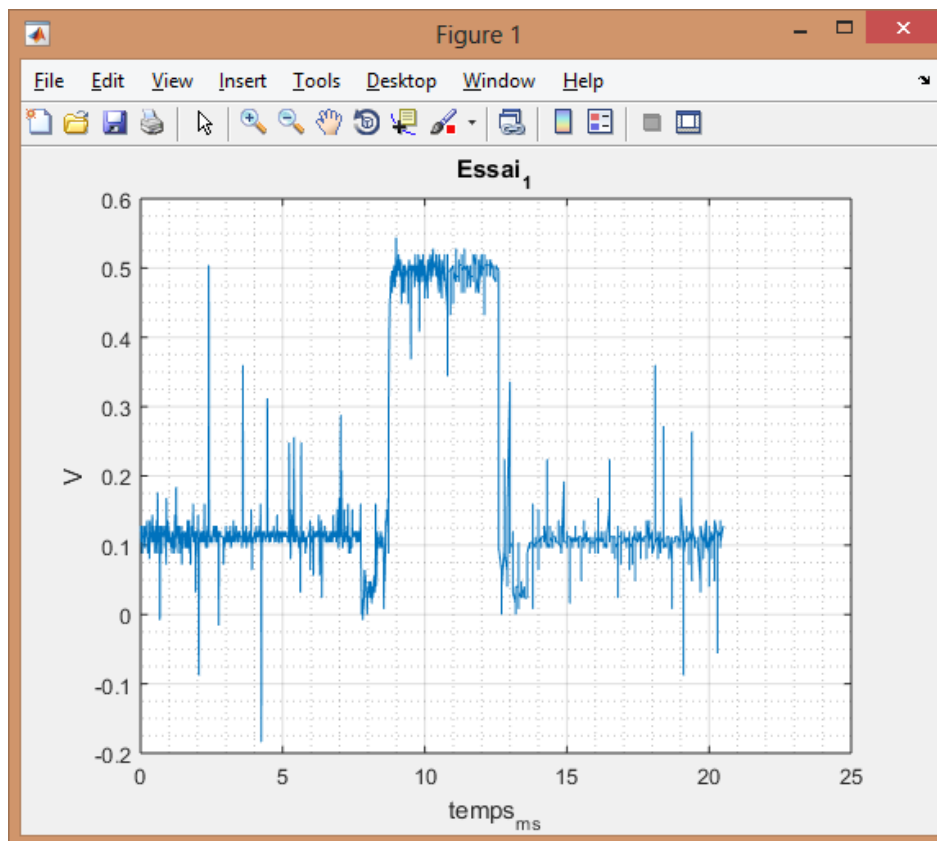


Figure 22 Courant consommé instantanément par le module XBee lors de la transmission Essai 1

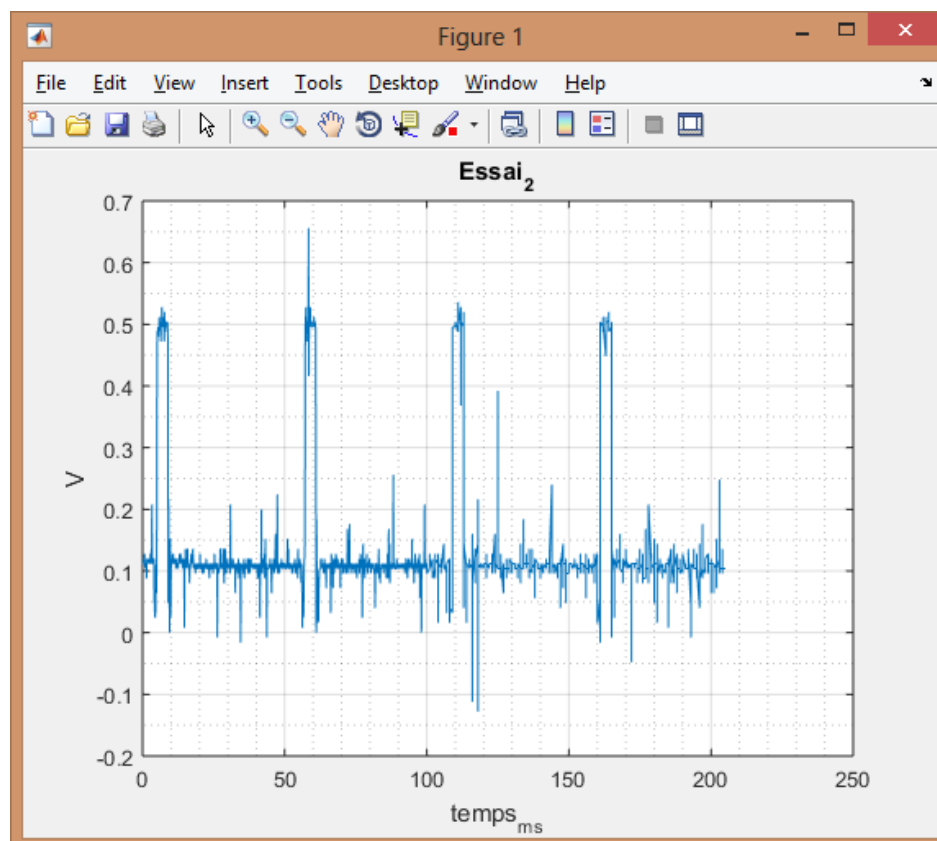


Figure 23 Courant consommé instantanément par le module XBee lors de la transmission Essai 2

6. Conclusion

Ce dernier chapitre a été consacré essentiellement aux tests, résultats et discussion. Plusieurs exemples de transmission d'images ont été effectués grâce à notre "testbed" en utilisant plusieurs configurations possibles de point de vue taux de compression. Les résultats obtenus sont très satisfaisants mais beaucoup d'améliorations sont nécessaires. L'inconvénient majeur de cette réalisation est le temps de transmission qui reste élevé. Il est dû essentiellement au support de transmission utilisé à savoir le XBee PRO. De même, nous avons tenté d'évaluer expérimentalement la consommation énergétique due à la transmission qui reste l'élément le plus pertinent dans l'augmentation de la durée de vie d'un réseau de capteurs sans fil surtout quand il s'agit de caméras ou de capteurs d'images.

Un autre point qui n'a pas été abordé dans ce travail expérimental est l'évaluation de la qualité de service (QoS) et aussi la qualité d'expérience ou perceptuelle (QoE). En effet, La QoS concerne plutôt des mesures techniques (Taux de perte de paquets, jigue, le retard, le délai ...etc) alors que la QoE tente de mesurer la satisfaction de l'utilisateur. Dans un scénario réaliste où nous aurons un nombre important de nœuds de capteurs sans fil la transmission sera confrontée à beaucoup d'autres problèmes qui vont engendrer des dégradations dans les images reçues. Nous devons donc prendre en considération tout ceci est essayez d'apporter des améliorations

Conclusion générale

Dans ce projet de fin d'études nous avons eu l'occasion de mettre en œuvre un prototype d'un nœud de capteur sans fil à faible consommation énergétique. Il doit pouvoir capter, compresser et transmettre des images avec une résolution donnée et une qualité de reconstruction acceptable. Ceci va devoir nous permettre plus tard de réaliser un système de surveillance de la faune sauvage et principalement des oiseaux migrateurs des zones humides d'Annaba. Pour atteindre cet objectif nous nous sommes fixés comme conditions l'utilisation d'un système de simple à utiliser, efficace et à faible coût. Le choix s'est porté automatiquement sur un arduino Due équipé d'un capteur d'images OV7670 et muni d'un support de transmission XBee Pro S1. Tous ces équipements sont d'un prix relativement modéré où l'ensemble ne dépasse pas 10000 DA.

Afin de réduire la consommation énergétique de traitement, en l'occurrence la compression jpeg à installer dans le module arduino due, nous avons opté pour l'utilisation d'une approximation de la DCT au lieu de la DCT réelle. Cette dernière présente une complexité calculatoire excessive qui atteint 60% de la complexité totale du JPEG. Ainsi, une approximation DCT va permettre une réduction drastique de cette complexité calculatoire et par conséquent elle va économiser l'énergie consommé dans le module arduino et augmente sa durée de vie.

Mais la consommation énergétique de la plus pertinent est due surtout à la transmission. D'où le choix d'un module XBee réputé pour être moins gourmand en énergie. Pour s'en assurer nous avons entrepris la mesure réelle de cette consommation moyennant un montage fort simple et efficace. Néanmoins, ce module de transmission souffre d'un autre handicap relatif à sa vitesse de transmission assez lente ce qui rend le délai de bout en bout pour une transmission d'une seule image prohibitif.

Ceci nous amène donc à plusieurs perspectives pour ce travail. D'abord il est crucial de doter notre système d'un moyen de protection contre les erreurs dues au canal de transmission sans fil réputé pour être hostile et corrompu. En effet, plusieurs types d'erreurs peuvent apparaître dans les données reçues causées par les bruits et les différentes interférences comme celles générées par les multitrajets (multipath). Rendre robuste les données transmises et une condition sine qua none pour le bon fonctionnement de ce système. Ainsi, la transmise sera résiliente aux erreurs. La seconde perspective est le choix d'un autre support de transmission plus rapide. Plusieurs solutions s'offrent à nous dont l'utilisation du wifi ou d'un moyen de transmission LoRa. L'évaluation de ce moyen de transmission doit prendre en considération l'amélioration de la QoS et aussi la QoE.

Une autre perspective qui nous semble très prometteuse surtout dans la surveillance de la faune sauvage est la stéréovision. Ainsi, à l'aide d'un petit système embarqué nous devons contrôler deux caméras simultanément afin de pouvoir déterminer la profondeur d'une image (la troisième dimension) et au lieu d'avoir des images à deux dimensions (2D) nous aurons des images 3D. Étant donné qu'une vision stéréoscopique est similaire au système biologique humain, certaines caractéristiques sont identiques. Par exemple, un humain a deux yeux pour voir des points de vue légèrement différents du même environnement. Un système de vision stéréo possède deux caméras situées à une distance connue et prend des photos de la scène en même temps. En utilisant la géométrie des caméras, nous pouvons appliquer des algorithmes et créer la géométrie de l'environnement. Dans notre projet ceci va nous permettre entre autres de déterminer la position exacte (x, y et z) de l'oiseau ou l'animal détecté et reconnu. Le challenge à lever est encore plus pertinent car il est difficile de pouvoir contrôler deux caméras en même temps, avec des angles de vues un peu différents, par un petit module embarqué comme l'arduino. Mais ce défi n'est pas impossible.

Références bibliographiques

- [1] JF Clements, TS Schulenberg, MJ Iliff, D Roberson, TA Fredericks, BL Sullivan et CL Wood. The eBird/Clements checklist of birds of the world: v2015. URL: <http://www.birds.cornell.edu/clementschecklist/download/IOC>, 2015.
- [2] <https://www.iucnredlist.org/>
- [3] <https://www.birdlife.org/news/tag/iucn-red-list>
- [4] A. Boulmaiz, N. Doghmane, S. Harize, N. Kouadria et D. Messadeg. The use of WSN (Wireless Sensor Network) in the surveillance of endangered bird species. A chapter of the book “Ubiquitous Computing: Cyber-Physical Systems, Smart Cities, and Ecological Monitoring”. will be published in Elsevier in 2019
- [5] David Lusseau. Evidence for social role in a dolphin social network. *Evolutionary ecology*, vol. 21, no. 3, pages 357–366, 2007.
- [6] Erick Stattner, Philippe Hunel, Nicolas Vidot et Martine Collard. Acoustic scheme to count bird songs with wireless sensor networks. In *World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2011 IEEE International Symposium on a, pages 1–3. IEEE, 2011.
- [7] Li Da Xu, Wu He et Shancang Li. Internet of things in industries: A survey. *IEEE Transactions on industrial informatics*, vol. 10, no. 4, pages 2233–2243, 2014.
- [8] Ladislav Ptacek, Lukas Machlica, Pavel Linhart, Pavel Jaska et Ludek Muller. Automatic recognition of bird individuals on an open set using as-is recordings. *Bioacoustics*, vol. 25, no. 1, pages 55–73, 2016.
- [9] Yang, Shuang-Hua. *Wireless sensor networks. Principles, Design and Applications. Signals and Communication Technology* Springer 2014.
- [10] Ke Xu, Yi Qu, Kun Yang A tutorial on the internet of things: From a heterogeneous network integration perspective *IEEE Network* 2016, pages 102-108
- [11] Varsha Bapat, Prasad Kale, Vijaykumar Shinde, Neha Deshpande et Arvind Shaligram. WSN application for crop protection to divert animal intrusions in the agricultural land. *Computers and Electronics in Agriculture*, vol. 133, pages 88–96, 2017.
- [12] Amira Boulmaiz, Djemil Messadeg, Nouredine Doghmane et Abdelmalik Taleb-Ahmed. Robust acoustic bird recognition for habitat monitoring with wireless sensor networks. *International Journal of Speech Technology*, vol. 19, no. 3, pages 631–645, 2016.
- [13] Amira Boulmaiz, Djemil Messadeg, Nouredine Doghmane et Abdelmalik Taleb-Ahmed. *Design and implementation of a robust acoustic recognition system for waterbird species using TMS320C6713 DSK*. *International Journal of Ambient Computing and Intelligence (IJACI)*, vol. 8, no. 1, pages 98–118, 2017.

- [14] Li Da Xu, Wu He et Shancang Li. Internet of things in industries: A survey. IEEE Transactions on industrial informatics, vol. 10, no. 4, pages 2233–2243, 2014.
- [15] XBee/XBee-PRO S1 802.15.4 (Legacy), RF Modules. User guide DIGI
<https://www.digi.com/.../documentation/digidocs/PDFs/90000982...>
- [16] OV7670 datasheet. web.mit.edu/6.111/www/f2016/tools/OV7670_2006.pdf
- [17] Rao K, Yip P, Discrete Cosine Transform: Algorithms, Advantages, Applications, Boston: Academic Press,1990.
- [18] Ahmed N, Natarajan T, Rao K. Discrete Cosine transform.IEEE Trans. On Computers 1974; C-23(1):90-93.
- [19] Britanak V, Yip PC, Rao K.Discrete Cosine and Sine Transforms. 1st Edition, Sept 2006 Elsevier.
- [20] Cintra RJ, Bayer FM. A DCT approximation for image compression. IEEE Signal Process.Lett2011;18(10):579–582
- [21] N. Kouadria, N. Doghmane, D. Messadeg, S. Harize . Low complexity DCT for image compression in wireless visual sensor networks. Electronics Letters 49(24):1531-1532 · November 2013
- [22] K. Mechouek, N. Kouadria, N. Doghmane, N. Kaddeche. Low Complexity DCT Approximation for Image Compression in Wireless Image Sensor Networks. Journal of Circuits, Systems and Computers 25(8) · August 2016
- [23] N. Kouadria, K. Mechouek, D. Messadeg, N. Doghmane. Pruned discrete Tchebichef transform for image coding in wireless multimedia sensor networks. AEU - International Journal of Electronics and Communications 74 · February 2017
- [24] N. Kouadria, K. Mechouek, S. . Harize, N. Doghmane. Region-of-interest based image compression using the discrete Tchebichef transform in wireless visual sensor networks. Computers & Electrical Engineering 73:194-208 · November 2018
- [25]https://www.digi.com/resources/documentation/digidocs/90001526/tasks/t_download_and_install_xctu.htm