Design and implementation of power management strategies for long range radio modules with energy harvesting

Philip-Dylan Gleonec

To cite this version:

Philip-Dylan Gleonec. Design and implementation of power management strategies for long range radio modules with energy harvesting. Hardware Architecture [cs.AR]. Université de Rennes 1 [UR1], 2019. English. NNT : . tel-02131685

THESE DE DOCTORAT DE

L'UNIVERSITE DE RENNES 1
COMUE UNIVERSITE BRETAGNE LOIRE

ECOLE DOCTORALE N° 601
Mathématiques et Sciences et Technologies
de l'Information et de la Communication
Spécialité : *Télécommunications*

Par
Philip-Dylan GLEONEC

Design and implementation of power management strategies for long range radio modules with energy harvesting

Thèse présentée et soutenue à Rennes, le 08/02/2018
Unité de recherche : IRISA -- UMR6074 | Equipe GRANIT

Rapporteurs avant soutenance :

Alain Pégatoquet	Maître de Conférences – HDR, Université de Nice-Sophia Antipolis
Hervé Barthélémy	Professeur d'Université, Université de Toulon-Var

Composition du Jury :

Président :	Laurent Clavier	Professeur d'Université, Institut Mines-Télécom
Examinateurs :	Jeremy Ardouin	CTO, Wi6labs
	Jean-François Diouris	Professeur d'Université, Université de Nantes
	Sylvie Kerouedan	Maître de Conférences – HDR, IMT Atlantique
Dir. de thèse :	Oliver Berder	Professeur, Université de Rennes 1
Co-dir. de thèse :	Matthieu Gautier	Maître de conférence, Université de Rennes 1

Time's up, let's do this.
Leeroy Jenkins

Remerciements

Je remercie tout d'abord Hervé Barthélemy et Alain Pégatoquet pour avoir accepté la charge de rapporteurs et pour leurs précieux retours sur ce manuscrit. Merci également à Laurent Clavier, Jean-François Diouris et Sylvie Kerouedan, qui m'ont fait l'honneur de participer au jury de soutenance.

Je remercie mes directeurs et encadrants de thèse Olivier Berder, Matthieu Gautier et Jeremy Ardouin pour m'avoir guidé, encouragé, soutenu tout au long de cette thèse. Ce manuscrit ne serait pas le même sans leurs conseils et expertises.

Je remercie également Wi6labs pour m'avoir fait confiance en me confiant et finançant ces travaux, ainsi que l'équipe Granit de l'IRISA pour son accueil.

Pour leurs conseils et leurs retours sur mes travaux, je remercie mes collègues de laboratoire, et notamment Fayçal Ait Aoudia pour son aide dans mes travaux. Je remercie également chaudement mes collègues de Wi6labs pour ces 3 années passées. Faire partie de cette équipe fut un réel délice.

Mersi bras à tous ceux qui m'ont fait sourire et soutenu durant cette thèse. À ma famille, parents, frères et sœurs, neveux et nièces, pour leur soutien sans faille durant ces 3 années, mais aussi les années passées et les années futures. À mes amis grâce à qui ces années ont été bien courtes, avec une pensée particulière pour ceux de Rennes et d'ailleurs qui ont toujours été à mes cotés.

Et enfin à tous ceux que je n'ai pas pu nommer mais qui demanderont “Et alors, cette thèse ?”

Et bien voilà, elle est finie.

Résumé en français

Contexte

La miniaturisation et la baisse du coût des systèmes électroniques connectés ont permis de créer de nouveaux services en connectant de nombreux appareils et objets. Cette révolution, appelée l'Internet des Objets (IoT), engendre notamment le déploiement de nombreux capteurs connectés, dont les mesures doivent permettre d'optimiser les processus auxquels ils s'intègrent afin d'en réduire les coûts. Ces capteurs visent un large nombre de cas d'usage, comprenant notamment l'optimisation de la production agricole ou de processus industriels, ainsi que la mise en place de villes intelligentes afin d'en réduire la consommation énergétique.

L'architecture de tous ces capteurs est généralement similaire. Chaque objet embarque un ou plusieurs transducteurs, qui transforment une donnée physique en une valeur électrique. Un microcontrôleur, composé d'un processeur, de ses mémoires et d'entrées/sorties afin de communiquer avec d'autres composants. Afin de communiquer les données mesurées, il intègre également une interface réseau. En plus de ces composants matériels, les capteurs connectés embarquent un logiciel embarqué ("firmware"), qui a pour fonction de récupérer les données des différents transducteurs, d'effectuer d'éventuels traitements sur ces données, et de les transmettre via le protocole de communication choisi.

Afin de réduire les coûts d'installation de ces réseaux, les capteurs communiquent général via des technologies sans-fil. De nombreux protocoles de communication sans-fil ont été développés afin de répondre à la demande croissante de systèmes IoT. Parmi les technologies utilisées, la technologie LoRaWAN permet l'envoi d'un message du capteur à une passerelle réseau sur plusieurs kilomètres avec une faible consommation d'énergie. Cette longue portée permet au réseau de couvrir une large zone avec un nombre réduit de passerelles, ce qui a pour effet de baisser le coût d'installation du réseau.

Ces systèmes électroniques doivent cependant être eux-mêmes alimentés en énergie. Alors que plus de 8 milliards d'objets connectés pourraient être déployés dans les prochaines années d'après certaines études, la question de la consommation énergétique de ces appareils ne peut être ignorée. L'alimentation sur secteur de ces systèmes pose des problèmes de déploiement, certains capteurs pouvant être installés à distance de toute infrastructure électrique. De plus, les courants de fuite de plus de 8 milliards de transformateurs AC/DC provoqueraient une forte hausse de la consommation électrique globale. L'utilisation de piles permet le déploiement de réseaux de capteurs à distance de toute infrastructure électrique, mais limite la durée de vie des capteurs. En outre, le remplacement de ces piles afin d'entretenir le réseau de

Source	**Puissance de la source**	**Puissance récupérée**
Lumière ambiante		
Intérieur	$0.1\,\mathrm{mW} / \mathrm{cm}^2$	$10\,\mu\mathrm{W} / \mathrm{cm}^2$
Extérieur	$100\,\mathrm{mW} / \mathrm{cm}^2$	$10\,\mathrm{mW} / \mathrm{cm}^2$
Vibration/Mouvement		
Humain	$0.5\,\mathrm{m}$ à $1\,\mathrm{Hz}$	
	$1\,\mathrm{m} / \mathrm{s}^2$ à $50\,\mathrm{Hz}$	$4\,\mu\mathrm{W} / \mathrm{cm}^2$
	$1\,\mathrm{m}$ à $5\,\mathrm{Hz}$	
	$10\,\mathrm{m} / \mathrm{s}^2$ à $1\,\mathrm{kHz}$	$100\,\mu\mathrm{W} / \mathrm{cm}^2$
Energie thermique		
Humaine	$20\,\mathrm{mW} / \mathrm{cm}^2$	$30\,\mu\mathrm{W} / \mathrm{cm}^2$
Industrielle	$100\,\mathrm{mW} / \mathrm{cm}^2$	1 — $10\,\mathrm{mW} / \mathrm{cm}^2$
RF		
Station de base GSM	$0.3\,\mu\mathrm{W} / \mathrm{cm}^2$	$0.1\,\mu\mathrm{W} / \mathrm{cm}^2$

Table 1: Aperçu de différentes sources d'énergie et leurs caractéristiques (source [1]).

capteurs serait coûteux et les piles usagées deviendraient de nouveaux déchets chimiques à traiter. Si l'utilisation de batterie rechargeable permet de limiter le volume de déchets chimique produit, elle ne réduit cependant pas le coût d'entretien des réseaux de capteurs.

En conséquence, il est intéressant d'utiliser l'énergie naturellement présente dans l'environnement (sous forme de lumière, chaleur, mouvement...) pour alimenter les futurs réseaux de capteurs connectés. Cette méthode permet de réduire le coût d'entretien des réseaux, puisque la batterie se recharge de manière autonome, sans besoin d'intervention humaine. L'essor récent des énergies renouvelables, couplé aux progrès de la miniaturisation, a permis le développement de récupérateurs d'énergie (panneaux solaires, éoliennes...) suffisamment petits pour être intégrés à des capteurs de taille réduite pour un coût modéré. Bien que délivrant de faibles quantités d'énergie, illustrées dans la Table 1, la faible consommation électrique des composants électroniques récents permet d'envisager des capteurs connectés entièrement autonomes en énergie.

Les sources d'énergie ambiante sont cependant généralement variables, et ne délivrent pas continuellement de l'énergie. C'est le cas de la récupération d'énergie solaire, qui ne délivre d'énergie qu'en journée. Afin de concevoir des systèmes électroniques entièrement autonomes en énergie, il est nécessaire de prendre en compte cette variabilité de l'alimentation. Pour cela, des gestionnaires d'énergie ont été développés, afin de moduler l'activité et la consommation énergétique du capteur à sa source d'énergie, permettant ainsi un fonctionnement théoriquement perpétuel. Cependant, ces technologies ont uniquement été testées sur des réseaux de capteurs à courte portée et très faible consommation, et jamais sur des réseaux longue portée tels que LoRaWAN. En outre, la possibilité d'exploiter plusieurs sources d'énergie ambiante pour alimenter le capteur reste peu explorée. L'objectif de cette thèse est donc la conception d'un capteur connecté à un réseau LoRaWAN alimenté par plusieurs sources de récupération d'énergie ambiante, et au fonctionnement entièrement autonome.

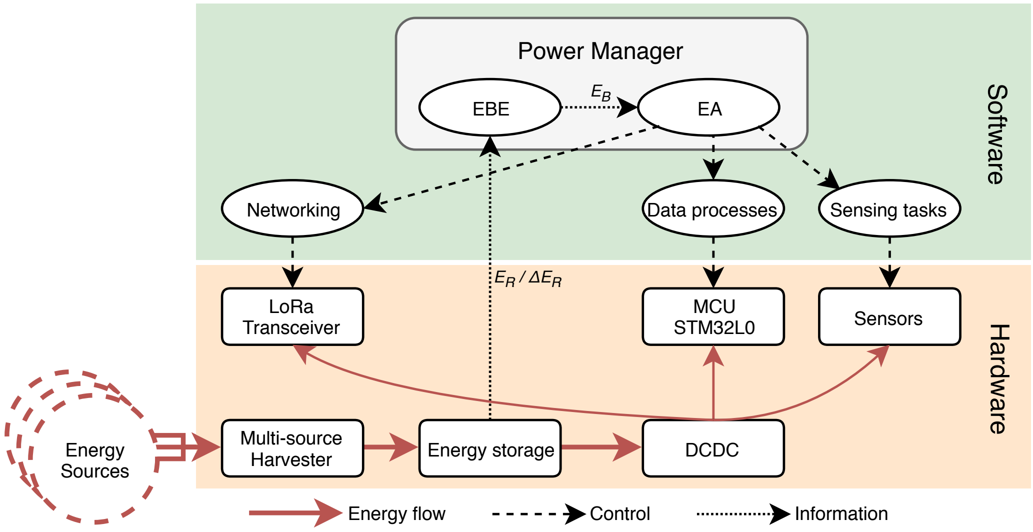


Figure 1: Diagramme bloc complet de l'architecture du capteur développé.

Récupération d'énergie multi-source pour capteur connecté

Ce travail commence par une étude de la fiabilité des différents composants constituant le capteur connecté. En effet, lorsqu'un gestionnaire d'énergie est utilisé afin d'assurer l'alimentation en continu du capteur, l'usure de ses composants devient l'un des facteurs limitant sa durée de vie. Cette étude, basée sur le référentiel MIL-HDBK-217, permet de comparer la durée de vie estimée des différents composants, afin d'isoler le maillon faible du capteur. L'étude démontre que les composants de stockage d'énergie (super-condensateur et batterie lithium-ion) sont les éléments limitant la durée de vie globale de l'appareil. Ces composants doivent ainsi être dimensionnés afin d'assurer une marge permettant au capteur de fonctionner sur toute la durée requise. Les batteries lithium-ion se montrent particulièrement sensibles aux contraintes thermiques et électriques, et doivent ainsi faire l'objet d'une attention particulière.

Après un état de l'art des circuits de récupération d'énergie ambiante, cette thèse présente un système permettant d'alimenter le capteur en exploitant simultanément plusieurs sources d'énergie. Afin de réduire le coût d'implémentation de la solution, ce circuit utilise un seul convertisseur de tension. Chaque source d'énergie stocke l'énergie récupérée dans un condensateur. Le convertisseur est successivement connecté à chaque condensateur, dont il puise l'énergie pour charger la batterie du capteur. Malgré une efficacité limitée, ce système permet de réduire le coût de la récupération d'énergie multi-source lorsqu'un grand nombre de sources d'énergie est utilisé. Sur la base de cette étude, une carte a été développée afin de récupérer simultanément l'énergie de 3 sources.

Gestion de l'énergie récupérée pour un capteur LoRaWAN

Afin d'assurer que le capteur ne vide pas son stockage d'énergie trop rapidement, il est nécessaire de réduire sa consommation énergétique. S'il est possible de réduire la consommation des composants en réduisant leurs tensions d'alimentation et leurs fréquences de fonctionnement (DVFS), le plus efficace est de mettre le plus de composants possible en veille le plus souvent possible. Cette technique, appelée "duty-cycling", est particulièrement adaptée aux capteurs connectés, qui n'ont généralement pas de tâches à exécuter entre deux mesures et envois de données. La plupart des composants, notamment le microcontrôleur et le composant de communication radio, peuvent ainsi être mis en veille la plupart du temps, réduisant leur consommation électrique à quelques μW contre plusieurs dizaines de mW lorsqu'ils sont actifs. Ainsi, en modulant l'activité du capteur, et donc sa qualité de service (QoS), il est possible de moduler sa consommation électrique.

En adaptant la consommation énergétique du capteur à la quantité d'énergie récupérée, il est théoriquement possible de faire fonctionner un capteur pour une durée illimitée, en évitant de complètement vider la réserve d'énergie. C'est le rôle du gestionnaire d'énergie. Ce module peut se diviser en deux sous-modules : un estimateur de budget énergétique (EBE) et un alloueur d'énergie (EA). Le premier calcule le budget énergétique pouvant être consommé par le capteur, tandis que le second décide de la meilleure manière de dépenser ce budget énergétique.

Deux types d'EBE sont utilisés. Les premiers sont dits "model-based", et se basent sur une modélisation de la source d'énergie ambiante utilisée pour prévoir la quantité d'énergie qui pourra être récupérée dans le futur. Ainsi, il est possible de délivrer un budget énergétique plus élevé, et donc obtenir une meilleure QoS, si l'on sait que de l'énergie sera disponible dans un futur proche. Ces EBE sont particulièrement bien adaptés pour les sources d'énergie périodiques, telles que la récupération d'énergie solaire. Cependant, ils sont peu adaptés pour la récupération d'énergie multi-sources, où il est complexe d'extraire un modèle d'une somme de sources d'énergie. Pour ce type d'applications, il est préférable d'utiliser des EBE dits "model-free", qui se passent de modèles de source et utilisent des algorithmes proches de ceux utilisés en automatismes afin de calculer le budget énergétique.

Cette thèse présente les travaux réalisés afin d'adapter ces EBE "model-free" à une utilisation avec des réseaux longue portée de type LoRaWAN. En effet, le délai entre deux transmissions peut s'étendre de plusieurs minutes à plusieurs heures au sein d'un réseau LoRaWAN, contre plusieurs secondes à plusieurs minutes au sein des réseaux courte portée utilisés dans les travaux existants. Cette thèse présente également une comparaison de plusieurs EBE, à la fois en simulation et sur de réels capteurs connectés, basés sur une plate-forme et un réseau commercial. Les résultats obtenus montrent que le choix de l'EBE a peu d'impact lorsque l'on cherche à maximiser leurs performances, ce qui permet l'utilisation d'algorithmes plus simples à régler et implémenter.

Le problème de l'allocation d'énergie pour les capteurs à récupération d'énergie a été peu étudié au sein des travaux de recherche existants, qui s'attachent pour la plupart à l'étude des systèmes temps-réel alimentés par récupération d'énergie

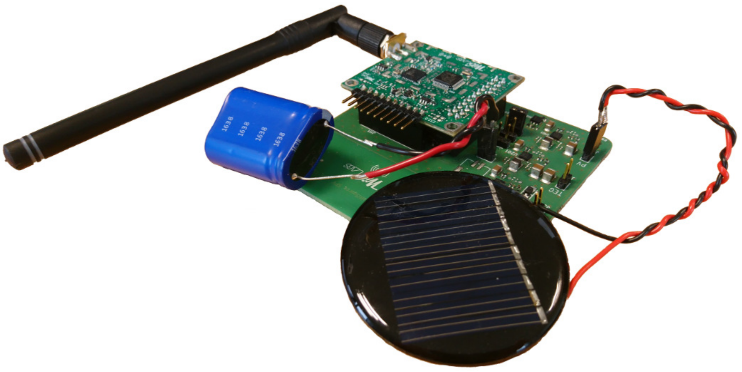


Figure 2: Capteur connecté développé et utilisé pour ce travail, ici avec un panneau solaire.

ambiante. En particulier, aucun travail antérieur n'étudie l'allocation d'un budget énergétique pour plusieurs tâches pouvant avoir des priorités et QoS requises différentes. Cette thèse présente en détail le problème, et propose une répartition optimale du budget énergétique pour plusieurs tâches avec différentes priorités. En outre, cette thèse présente un algorithme intégrant ce calcul optimal pour allouer un budget énergétique à plusieurs tâches en prenant en compte leurs contraintes de QoS. Cet algorithme a été validé sur un capteur réel effectuant des mesures de température, humidité, bruit et concentration de gaz, communiquant sur un réseau LoRaWAN et alimenté par récupération d'énergie ambiante.

En conclusion, cette thèse répond à l'objectif initialement fixé : la conception d'un capteur communiquant sur un réseau LoRaWAN alimenté par plusieurs sources de récupération d'énergie et entièrement autonome. Ce document détaille la conception d'un nouveau circuit de récupération d'énergie multi-sources et la solution finalement choisie. Plusieurs algorithmes d'EBE sont comparés en simulation et sur des capteurs réels. Un nouvel algorithme d'allocation d'énergie, basé sur une allocation optimale, est développé et implémenté. Le prototype ainsi réalisé est alimenté par plusieurs sources d'énergie, et dispose d'un gestionnaire d'énergie capable d'estimer comment dépenser au mieux l'énergie récupérée.

Table of Contents

List of Figures

List of Tables

Introduction

New Advances in Internet of Things

A famous principle in engineering, management and many other disciplines is that what can not be measured, can not be improved. Measurements are mandatory to improve systems, in order not to act with simple guesses of the effects. In order to optimize their processes, multiple industries have used sensors to monitor the performance of their machines and processes. Applications of such sensor systems include predictive maintenance, process automation, workload optimization, and many others. In such systems, sensors can communicate with other systems, and can thus be named Machine to Machine (M2M) communications. With the increasing importance of the Internet, it has become easier to directly connect sensors and objects to the web than designing specific systems. This phenomenon where objects are directly connected and accessible from Internet forms the basis of the Internet of Things (IoT).

For a long time, communicating devices have been connected by wired technologies, as it was cheaper and more reliable than early wireless communications. However, installing a wired network can be costly and requires more work. The recent advances of wireless communication technologies drastically reduced their cost and improved their reliability, making them a contender for traditional wired systems in a variety of use-cases. In particular, for data gathering applications, using a fleet of connected sensors, the use of wireless technologies makes sense, as it reduces the installation cost of the network and eases the deployment. Such a system is called a Wireless Sensor Network (WSN), and is typically composed of nodes which measure a physical value and transmit it over the network.

A WSN node is usually a simple system, as shown in Fig. 3. It is composed of a micro controller, which integrates in a single package a processor core, its required memories and interfaces to communicate with peripheral components. These periph-

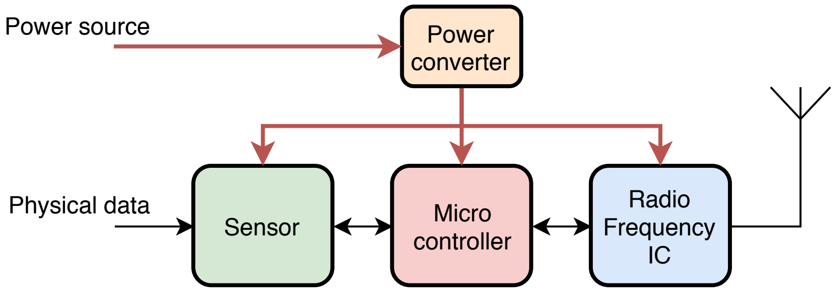


Figure 3: Block diagram of a usual WSN node.

eral components include one or more sensors, to sense physical data, and a Radio Frequency (RF) Integrated Component (IC) used to communicate with the WSN. Some recent components directly integrate the radio IC in the same package as the micro controller, to lower the cost of the system and ease its design. Around these core components, power supply components are used to power the system, typically from a primary battery.

In order to follow with the rise in demand for WSN, multiple technologies have been developed and adapted for these use-cases. An overview of their different characteristics is given in Table 2. WiFi and Bluetooth, which are commonly found in consumer products, have been adapted for low power use-cases, through the development of WiFi Hallow and Bluetooth Low Energy (BLE), respectively. Other networking technologies, such as Zigbee, Z-Wave, the 802.15.4 Media Access Control (MAC) layer or the 6LoWPAN routing layer, have been created specifically to address these applications. In some use-cases, cellular networks (GSM, EDGE, HSPA, LTE) can also be used, in order to take advantage of their wide coverage. By extension, some cellular technologies, as LTE-M and NB-IoT, have been standardized to more efficiently address these use-cases. As cellular technologies are usually power hungry, new types of long range wireless communication systems have been developed in order to enable communications with a long range (multiple km) and low power consumption, at the cost of lower data throughput. Such technologies include Weightless, Sigfox from the eponymous society, and LoRa technology from Cycleo, later bought by Semtech.

The LoRa technology has attracted many actors, due to its open nature which enable multiple MAC and applicative layers to be used on top of it. These actors include semiconductor companies, network operators and equipment design companies. In particular, an alliance has been created to develop LoRaWAN, a networking technology based on LoRa transmissions, used to set up Low Power Wide Area Networks (LPWAN) at a lower cost than cellular technologies. The LoRa technology is based on a chirp spread spectrum modulation, which enables communications with a range up to 10 km with a limited transmission power. As it transmits messages on unlicensed frequency bands, LoRa communications are limited by duty cycle regulations. Therefore, the data throughput is limited to a few kB per day. On top of this physical layer, the LoRaWAN technology implements a star network configuration, in which all devices transmit their data to a gateway. The gateway then transmits the messages to a network server, which decrypts and decodes the message, and

Wireless network type	Advantages	Drawbacks	Examples
Local Area Network	Possible high data rate Wide userbase Existing networks	Low range Higher power consumption	Bluetooth, BLE, WiFi, Zigbee, Z-Wave
Low Power Wide Area Network	Low power consumption Low cost of ownership High range	Small userbase Low data rate	LoRa, Sigfox, Weightless, NB-IoT
Cellular Network	Possible high data rate Wide userbase Existing coverage	High cost of ownership High power consumption	GSM, HSPA, LTE

Table 2: Overview of different wireless communications possibilities for IoT nodes.

eventually forwards the payload to an application server. All LoRaWAN messages are encrypted, except their metadata, so that the data payload can only be read by the receiving server.

Although LoRaWAN systems are limited in throughput, they are suitable for a large variety of use-cases. LoRaWAN sensors are typically used in metering applications. In this type of application, a measurement is periodically sent to a server. LoRaWAN actuators can also receive commands from a server, for example to power on a radiator when a sensor detects a low temperature. Finally, LoRaWAN devices can be used for alarm systems, where a message is only transmitted at an event detection. To generalize, LoRaWAN is a suitable solution for a device which has to transmit occasional small data packets with a low energy consumption. The use of long range communications also eases the deployment of the network and lowers its cost.

Powering the Internet of Things

IoT devices are expected to be used in a variety of applications and contexts. Due to the large number of applications for these technologies, it is expected that billions of connected devices will be deployed in the coming years. Some reports point that up to 20 billions IoT nodes could be in use by 2020 [2]. However, deploying so many devices comes with its own challenges. One of them is powering these billions of devices efficiently, in order to limit their environmental impact.

A first way to power these sensors is simply to plug them on a classical electric grid. However, power converters have losses that can easily reach hundreds of mW, due to the high voltages on main networks. Thus, powering 20 billions devices each with a power supply that has 100 mW of power leakage, an optimistic estimation, would leak 2 GW of power, which is not desirable.

An alternative is to power the IoT nodes with batteries, which work around the leakage power in AC power converters. Modern IoT nodes can last a few years on a suitable non-rechargeable primary battery, provided the node is designed for low power. Moreover, non rechargeable batteries are usually cheap, which can help the adoption of WSN technologies. However, these batteries have a limited life cycle, and they are hard to recycle when they are depleted. Thus, the use of non rechargeable batteries for 20 billions devices would create too much chemical waste for it to be sustainable.

The use of rechargeable batteries could circumvent this problem and increase the life cycle of the device, provided the battery has been characterized so that its wear-out does not limit it. When the battery is depleted, it could simply be recharged and put back in operation. However, like changing non-rechargeable batteries, this operation requires a technician to fetch the node, recharge its battery and set it back to its place. This operation is not instantaneous, as rechargeable batteries require time to be charged. Moreover, accessing the WSN node is not always possible. In some environmental measurement campaigns, for example, the WSN node could have been dropped in a forest, set up on a cliff or near a volcano. Even in an urban setting, a sensor placed in a high place would require specific equipment and operators to fetch the sensor, which may have a high cost. This makes recharging batteries hard

Source	**Source Power**	**Harvested Power**
Ambient light		
Indoor	0.1 mW / cm^2	10 μW / cm^2
Outdoor	100 mW / cm^2	10 mW / cm^2
Vibration/Motion		
Human	0.5 m at 1 Hz	
	1 m / s^2 at 50 Hz	4 μW / cm^2
	1 m at 5 Hz	
	10 m / s^2 at 1 kHz	100 μW / cm^2
Thermal energy		
Human	20 mW / cm^2	30 μW / cm^2
Industrial	100 mW / cm^2	1 — 10 mW / cm^2
RF		
GSM base station	0.3 μW / cm^2	0.1 μW / cm^2

Table 3: Overview of different energy sources and their characteristics (from [1]).

to justify economically, as it would increase the cost of ownership and maintenance of the WSN.

In order to enable sustainable operation of IoT nodes, the use of energy harvesting technologies has been considered. Indeed, energy is present in most environments, and even in living being, either vegetal [3] or animal [4]. If the WSN node can recharge its batteries by harvesting energy from its environment, the need for maintenance is reduced, which lowers the cost of ownership of the WSN. Multiple energy sources, shown in Table 3, can be used, including light, wind or vibrations, which can respectively be harvested with solar panels, wind turbines and piezoelectric elements. Although this table does not take into account state of the art energy harvesting sources, it shows the order of magnitude of the expected harvested energy for each type of source. Due to the specificity of each of these sources, and their variable nature, specific hardware has to be used to implement energy harvesting capabilities. Although these added components increase the cost of the node, they also increase its life cycle, thus making it more sustainable. As most WSN applications require little power to run, energy harvesting is an efficient solution to efficiently power IoT nodes in a sustainable manner.

In order to maximize the use of harvested energy by the WSN node, multiple approaches can be imagined. Using a bigger energy harvesting source enables the node to harvest more energy. However, the size of the energy source may be limited by the cost and size requirements of the product. An alternative approach to increase the quantity of harvested energy is the use of multi-source energy harvesting, where energy is harvested from multiple different energy sources simultaneously. The use of multiple sources enable the node to still harvest energy when a source is not active (e.g. a solar panel at night), thus increasing its QoS.

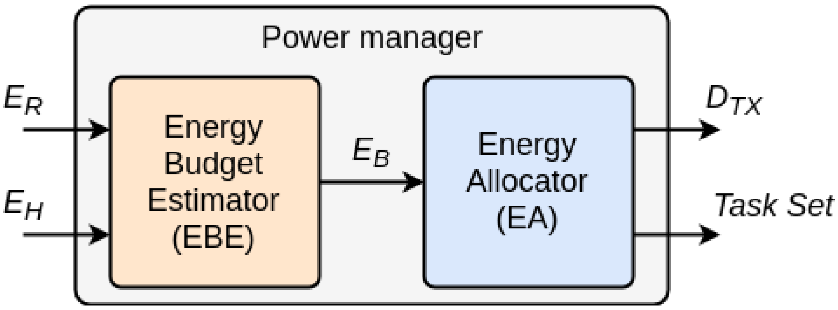


Figure 4: Structure of the power management logic.

Power Management for Wireless Sensor Nodes

While an energy harvesting device can be used to power IoT nodes, it is usually a variable power source, which can deliver no energy over some periods of time. Due to this, the IoT node should embed functionalities to properly manage its energy and avoid battery depletion. In energy harvesting WSN nodes, a power manager is implemented to fill this task. To adapt the node power consumption and ensure it does not deplete its battery, the power manager can reduce the QoS of the node, for example by increasing the delay between two consecutive transmissions to let the components a longer time in a low power mode. The power manager can also increase the QoS of the node when the battery is sufficiently filled.

In this thesis, we decompose the power manager (shown in Fig. 4) in two elements: an EBE and an EA. The EBE monitors the energy capabilities of the WSN nodes, i.e. the residual energy in storage E_R, the harvested energy E_H, etc., and uses these measurements to compute an energy budget E_B which can be spent by the node over a time-slot. The EA takes this E_B, its knowledge of the tasks the node has to perform and their required QoS as inputs. Using this information, it manages the activity of the node, such as the delay D_{TX} between two transmissions or the task set it executes, so that it does not consume more energy than E_B.

The design of power management schemes for energy harvesting WSN nodes has been an active area of research in recent years. However, it can be noted that most works focus on the use of meshed networking technologies [5][6][7], closer to 802.15.4 and Zigbee protocols than to the LoRa technology. Due to the differences in use-cases and power consumption between these systems, many aspects of these algorithms have to be adapted to fit long range wireless communications applications.

Moreover, these power management systems only consider the case of single-task devices, where the task consists of measuring a value and transmitting it immediately. As IoT systems are used in more and more applications, their complexity have increased and some devices now have to execute multiple tasks. These tasks can be different measurement or data processing tasks. As this case has not been considered in previous works, there is a need to adapt energy harvesting power management systems to multi-task WSN nodes.

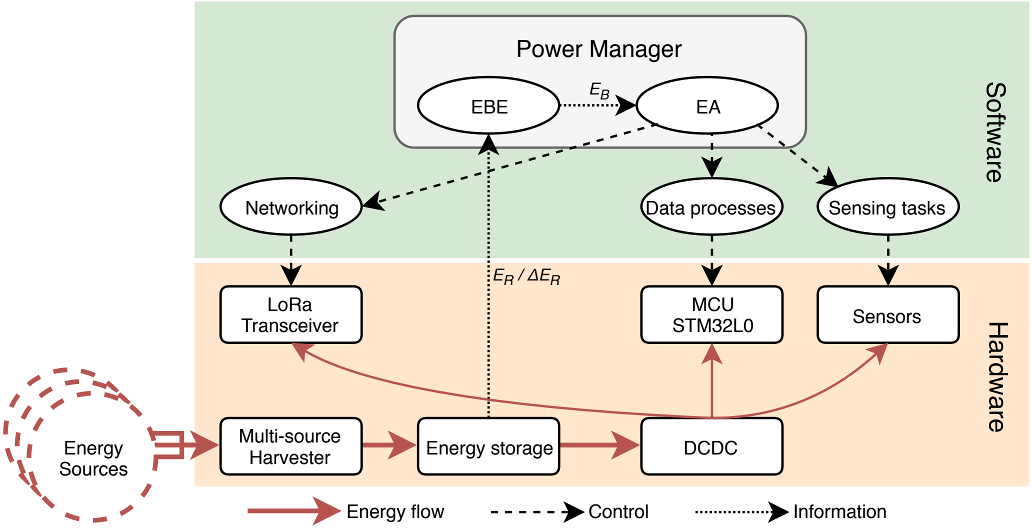


Figure 5: Full block diagram of the proposed energy harvesting WSN node.

Contributions and Thesis Outline

Contributions

This thesis aims at developing an energy harvesting WSN node, as shown in Fig. 5, communicating over a LoRaWAN network and based on an available commercial platform. A classical battery powered WSN node is composed of a Micro-Controller Unit (MCU), one or more sensor and a radio transceiver, with embedded software to manage the sensing tasks, data processing and networking stack. In addition, the use of energy harvesting requires the use of one or more energy harvester and the use of a rechargeable energy storage. Moreover, the embedded software is more consequent, as a power manager is integrated to ensure the proper operation of the WSN node. The final objective is to add multi-source energy harvesting capabilities, as well as a new power manager which enables the implementation of multiple sensing tasks with different QoS requirements. The main contributions of this thesis are:

- A study of the reliability of components involved in energy harvesting WSN node design, in order to detect which component has the highest probability to break earlier than other. This study shows that the life cycle of an energy harvesting system is limited by its energy storage life cycle. To our knowledge, this is the first application of this type of analysis to energy harvesting WSN nodes.

- A new multi-source energy harvesting architecture, aimed to lower the implementation cost of such solutions [8]. The system uses a single power converter and dynamically switches between multiple energy harvesting sources. The new architecture has a limited efficiency, but the use of a single power converter is more cost effective when a high number of different energy sources is used.

- An implementation and comparison of EBE algorithm on a real world commercial platform [9] using LoRa communications. To the best of our knowledge, this is the first industrial implementation of this type of system, and the first energy harvesting power management system adaptation for LoRa communications. This study shows that, for long range systems, the choice of EBE algorithm has little impact over the performance of the system, provided that all algorithms have been properly tuned to maximize performance, which enables the use of simpler algorithms.

- An exploration of energy allocation techniques for energy harvesting WSN nodes. The problem is detailed for both single task and multi-task nodes. Multiple solutions are shown to adapt the QoS of the node to its energy capabilities, detailing their advantages and different use-cases. To the best of our knowledge, this is the first work that considers multi-task energy harvesting WSN nodes.

Thesis Outline

This thesis is divided in two parts. Part 1, composed of Chapters 1 and 2, focuses on the hardware components of energy harvesting WSN nodes, and the design of multi-source energy harvesting systems. Part 2, which comprises Chapters 3, 4 and 5, concentrates on the firmware of WSN nodes, and especially on the design of power management systems for energy harvesting devices.

Chapter 1 - Energy Harvesting Components The Chapter 1 presents the different elements of an energy harvesting WSN node. For each possible energy source and energy harvesting component, a detailed state of the art shows the recent advances in energy harvesting technologies. The first contribution of this thesis, a reliability analysis of these components, is presented, as a way to estimate the life cycle of energy harvesting WSN nodes.

Chapter 2 - Multi-source Energy Harvesting The Chapter 2 describes the second contribution of this thesis, a new multi-source energy harvesting circuit which aims at cost reduction. Our architecture is compared to state of the art architectures to detail the use-cases where it is more efficient and enables cost reduction of the system. Based on this study, a multi-source energy harvesting board is designed for an industrial long range IoT platform.

Chapter 3 - Power Management for Energy Harvesting IoT nodes The Chapter 3 details the challenges of power management for Energy Harvesting IoT nodes. A state of the art of power reduction techniques and harvested energy management concepts are presented. This chapter introduces many techniques which are used to reduce the energy consumption of energy harvesting WSN nodes and the purposes of managing harvested energy.

Chapter 4 - Energy Budget Estimators for Long Range IoT nodes The Chapter 4 introduces energy budget estimation concepts and their application to long range WSN nodes. A state of the art of different EBE algorithms is presented. The constraints of power management for LoRa IoT nodes are introduced, and our approach to integrate these constraints is presented. The third contribution of this thesis, a comparison of state of the art EBE algorithms, both in simulation and implementation on real-world IoT nodes, is detailed.

Chapter 5 - Energy Allocation The Chapter 5 presents the energy allocation problem for energy harvesting WSN nodes. Multiple solutions to adapt and increase the QoS of a node are presented, as well as their limits and the use-cases they are fit for. In particular, the problem of energy allocation for multi-task is detailed, and multiple solutions are presented to efficiently allocate the energy budget to different tasks with different energy and QoS requirements.

Part I

Multi-source energy harvesting

Chapter 1

Energy Harvesting Components

Energy harvesting devices are complex systems, which integrate multiple technologies. In particular, the low power consumption required by energy harvesting needs specific design trade-offs, e.g. between cost, size, performance of the solution. This chapter presents the different component technologies available for the design of energy harvesting systems, and the trade-offs associated with each technology. Moreover, an analysis of energy harvesting components lifetime is presented to show the design choices impact on the system lifetime.

1.1 Energy Harvesting Sources

Energy can be harvested from a variety of environmental sources. Each energy harvesting source can be used for different use-cases, with its own design trade-offs. In the following subsections, an overview of the different energy harvesting possibilities is given. For each source, state of the art implementations and applications are described, to show the activity and technical progress of these devices.

1.1.1 Solar Energy

Out of all energy harvesting sources, solar panel has been one of the most researched energy source, as shown in the best research cell efficiencies compiled by the National Center for Photovoltaics [10]. Indeed, light is nearly everywhere, making it an ideal energy harvesting source used in multiple systems [11][12][13]. However, in most of these environments, light is only present intermittently and the node has to manage the period where no energy is present. Energy from light is harvested using photovoltaic cells, which provide power when exposed to light. A simple model of a single solar cell is provided in Fig. 1.1. To form a solar panel, multiple solar cells can be connected in series to generate a higher voltage or in parallel to generate a higher current. As solar energy harvesting is an active research field for both small and large scale systems, a number of materials has been proposed to increase the energy harvesting efficiency. However, most systems use traditional silicium based solar cells.

Traditional solar cells are manufactured from crystalline silicon. Solar cells manufactured from mono-crystalline silicon usually yield a higher conversion efficiency (up to 26.7 % [14]) than poly-crystalline based solar cells (up to 21.9 %), but are

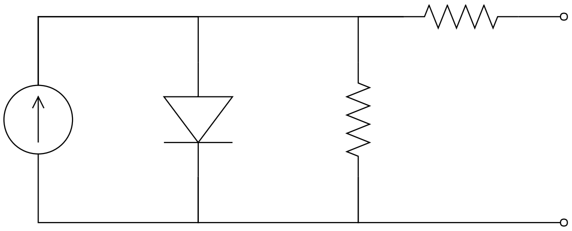


Figure 1.1: Model of a solar cell.

also more expensive. Their high efficiency and widespread adoption make these cells a first-rate candidate for light energy harvesting. However, these cells have a lower efficiency under low illumination [15], and thus are unsuitable for indoor light energy harvesting. In this case, amorphous silicon solar cells [16] are well suited, as their efficiency stays relatively similar under low illumination. However, they are more expensive and have a lower overall conversion efficiency (up to 10.2 % [14]).

Furthermore, multiple new structures have emerged in recent years to create cheaper and more efficient solar cells. Thin-film solar cells, where a thin film of photovoltaic material is deposited on a rigid or flexible substrate, have been developed as a cheaper alternative to traditional crystalline silicon cells. However, they usually have a lower efficiency (4.3 to 11.0 % [17]) even though recent research shows promising results with efficiency up to 22.3 % [18]. An alternative to reduce the cost of photovoltaic energy harvesters is the use of organic solar panels. This technology enables the manufacturing of solar panels printed on flexible substrate, which can be easily manufactured [19] but yields a low efficiency ($<$ 4 %). However, recent research has reported a 15 % efficiency for an organic photovoltaic solar panel [20] which shows the potential of this technology.

The popularity of solar energy harvesting for energy harvesting WSN has led to closer integration between the energy source and the power consumer. [21] proposes the implementation of an array of photodiodes on a chip with associated storage capacitors. This implementation enables an energy harvesting device in a small form factor. [22] describes the design of a system which integrates a thin-film solar panel, a thin-film battery and a battery management circuit implemented using thin-film components, which enables the addition of solar energy harvesting capabilities in a plug and play manner.

1.1.2 Thermal Energy

Energy can be generated from a ThermoElectric Generator (TEG) such as shown in Fig. 1.2 due to the Seebeck effect, a phenomenon which creates a voltage difference between two different conductors when their temperatures are different. This effect can also be reversed to create Peltier cells for cooling/heating applications. The voltage difference is usually directly proportional to the temperature difference. This voltage is low, from a few tens to a few hundreds of mV, and has to be converted to a higher voltage in order to be usable by a WSN node, usually in a range from 1.8 to 5.0 V. However, the delivered current is higher than other energy harvesting sources, and a characterization of commercial thermoelectric modules [23] has shown that up to a few tens of mW can be extracted from a TEG. In an application, the hot side

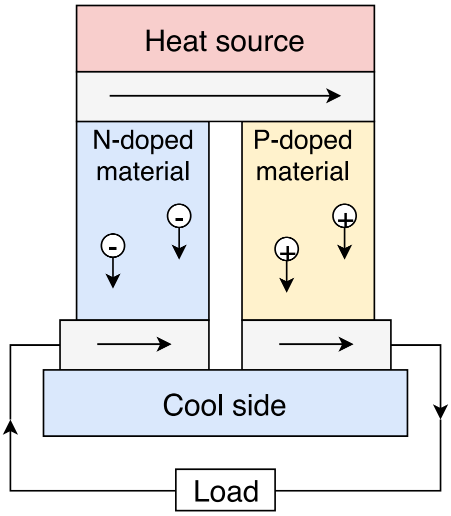


Figure 1.2: Structure of a TEG.

of the TEG is attached to a heat source, while the cold side has to be refreshed, typically with a heat sink.

The use of TEGs is particularly interesting to get energy from wasted heat. A good example of such applications is presented in [24]. In this work, a WSN is deployed in a server farm to monitor temperature, humidity, light, supply voltage and concentration of carbon-monoxide and volatile organic compounds. Each sensor is powered by a TEG placed between the Central Processing Unit (CPU) and its heat sink, thus taking advantage of the energy wasted by the servers. In [25], an application-specific TEG is developed to take advantage of human heat, which is able to generate a voltage up to 9 V, to power an electroencephalogram (EEG) sensor with 18 μW. This work shows the potential use of thermo-electricity for biomedical applications, typically to power sensors such as EEG modules and to ease the maintenance requirements for the patient.

Thermoelectric generator can also be an alternative to other energy sources when they are not available. In [26], a TEG is used to power autonomous sensors located in walls, where no light is available. The sensors are powered by the heat dissipated by walled-in pipes of a heating plant where hot air is fluxed. A heat sink is mounted on the cold side to maximize heat exchange with the environment. A similar installation could be deployed in residential areas near water heaters, where cold and hot water pipes are usually close to each other.

One of the drawback of TEGs is the complex mechanical and thermal integration. Indeed, the hot side of the TEG must be placed as close as possible to the heat source, but electronic components are sensitive to heat and could be damaged if the temperature of the heat source is too high. [27] proposes an intelligent integration of the TEG between two Printed Circuit Boards (PCBs) fabricated using aluminum instead of usual epoxy and copper, shown in Fig. 1.3. The two PCBs act as heat collector and spreader due to the high thermal conductivity of aluminum. They are separated by an insulated foam and the electronic components are mounted on the PCB attached to the cold side of the TEG. This proposed form factor helps packaging the solution and avoids separating the thermoelectric module from the electronics.

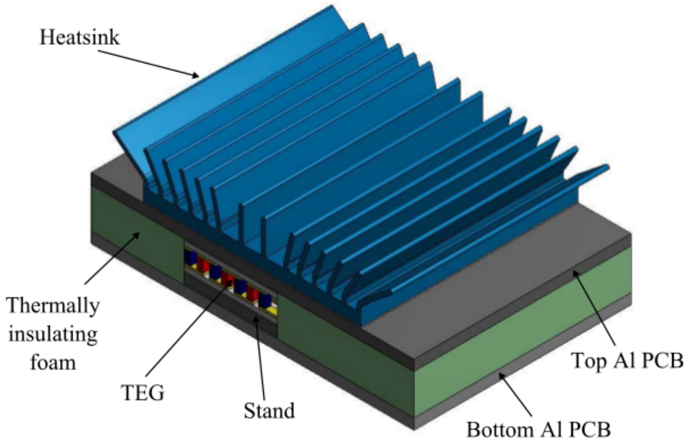


Figure 1.3: Thermoelectric generator application presented in [27]

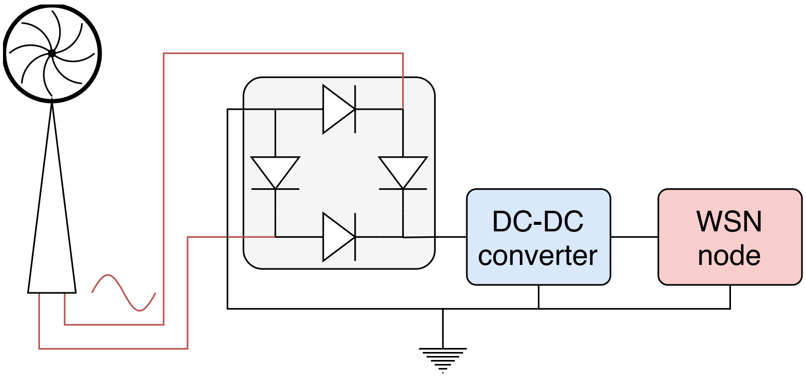


Figure 1.4: Setup of a wind turbine. The alternative voltage delivered by the energy harvester has to be rectified, typically with a diode bridge.

In a similar way, [28] shows the implementation of a solar TEG by using a TEG between an aluminum heat sink and a large aluminum panel exposed to the sun.

1.1.3 Wind Energy

Wind turbines are well known for their use in large scale electricity production, but they can also be used to power WSN node. Wind turbines can easily deliver tens of mW, which is interesting to quickly recharge a WSN node battery or power devices that require more energy, such as image sensors. They provide an alternative voltage, which has to be rectified to be used, as shown in Fig. 1.4. Despite their high power supply potential, few systems have been designed for wind energy harvesting. [29] highlights the difficulties and challenges of designing small scale wind energy harvesting systems. In particular, the size of small scale wind turbines is heavily constrained, and the small blades severely reduce the aerodynamic force. Moreover, these small devices are deployed at lower height, where the wind speed is lower and the wind flow is more likely to be disturbed by the surrounding environment.

Wind energy has a good complementary with solar energy. There is often less wind on sunny days, when a solar energy harvester gives energy, than on cloudy days, where the wind turbine can provide energy instead of the solar panel. This approach is used to increase the global harvested energy in multiple works. In [30], for example, the authors use both a solar panel and a wind turbine in a multi-source energy harvesting setup, backed up by a hydrogen micro fuel cell when both battery level and harvested energy are low. [31] proposes a platform which can harvest energy from a solar panel, a wind turbine, a TEG and vibrations. In an agricultural context, [32] harvests energy from the sun, the wind and the water flow of nearby irrigation to monitor fields.

Another method for harvesting energy from wind is presented in [33]. In this work, an anemometer, usually designed to measure wind speed, is used to generate electricity. The anemometer vertical shaft is used to turn a compact alternator, which is used to recharge a battery. This approach results in a small energy harvesting device, but the output power is less than 1 mW (less than 700 μW after power conversion), which is much lower than usual wind turbines.

It must be noted that most wind energy harvesting examples are used in an outdoor setting, where wind is obviously more present than indoors. However, there are also indoor use-cases for wind energy harvesting. [34] for example presents WSN node powered by a wind turbine in an underground tunnel of a metro line. In this use-case, no other energy sources (solar, thermoelectric, radio-frequency...) are available other than the wind generated by the train when it passes near the wind micro-turbine. [35] is another example, where a micro-generator harvests the wind flow from aeration pipes with magnetic and piezoelectric materials.

1.1.4 Electromagnetic Energy

Since electricity and radio communications are now pervasive, energy can be found everywhere in the form of electro-magnetic fields. Thus, scavenging this energy would enable the installation of energy harvesting WSN in a large variety of use-cases. Due to its wave nature, energy harvested from electro-magnetic field creates an alternating voltage, which is then rectified and converted to be used by the WSN node.

RF is now universally used as a communication medium. The ambient electro-magnetic field can be harvested to power low power devices, with different methods depending on the distance between the energy harvester and the RF emitter. Near-field RF energy harvesting typically uses induction to transmit power from a source to a receiver. Use-cases of such technology typically include Radio-Frequency Identification (RFID) and Near-Field Communication (NFC) systems, where an active reader is used to power and read measurements from a passive tag. Since some of these tags only have to be powered when they are asked for measurement, such systems can be implemented without use of any battery [36]. Although near-field energy harvesting is usually used for short range power transmission, the range can reach up to 5 m [37].

Since the power of an electro-magnetic wave decreases quadratically with the distance it crosses, and even quicker in non line-of-sight cases, long range RF energy harvesting usually yields very low power output. Nevertheless, due to the omnipresence of electro-magnetic fields, a large number of work have been published to develop such technologies. [38] uses RF energy harvesting to power a tag to monitor the heart of patients. However, this device only harvests 15 μW when an RF source located

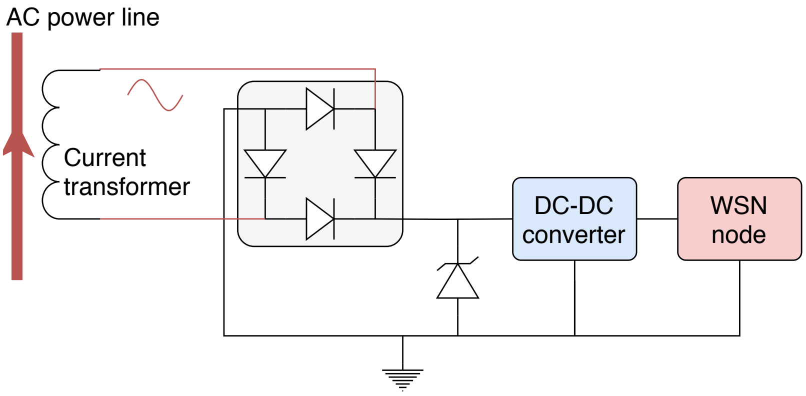


Figure 1.5: Setup of a current transformer used to harvest energy from an AC power line. The generated voltage is rectified, and a Zener diode is used to protect the power converter from overvoltages generated by the transformer.

3.1 m away emits 800 mW Equivalent Isotropic Radiated Power (EIRP) at 800 MHz, which shows the low efficiency of this technology. A similar system presented in [39] also requires an input power of -19.7 dBm to operate, which is not always available in the environment. In order to avoid setting up RF emitters for the sole purpose of transmitting power, with low efficiency, to energy harvesting devices, some works have focused on the use of ambient field. For example, [40] manages to harvest up to 8 μA from a 1.584 MHz amplitude modulated radio signal, and [41] harvests up to 6.93 and 18.75 μW for a 1 and 18 MΩ load, respectively. These power outputs are promising but are still too small to efficiently power a WSN node.

Electricity flowing through cables also creates an electro-magnetic field around the conductor. In this case, the energy which can be harvested is a function of the power flowing through the line. [42], for example, presents a system which harvests up to 1 or 2 mW power from AC power lines, which is sufficient to supply a low power device. [43] explores different methods to harvest energy from a high power cable, notably a magnet attached to a piezoelectric energy harvester which provides up to 108.24 μW and a flux concentrator, which can provide up to 257 mW. Finally, [44] uses a current transformer, as shown in Fig. 1.5 to harvest energy from lower power cables, such as electrical motor power supplies.

1.1.5 Piezoelectricity

Some materials have a property to accumulate electric charges when they are subjects to mechanical constraints. This property can be exploited to create devices used to generate mechanical vibrations from electricity (e.g. piezoelectric buzzers) or to generate electricity from mechanical movements. Most piezoelectric devices in the industry are made of PZT (lead zirconate titanate — Pb Zi Ti) and PVDF (polyvinylidene fluoride) [45] instead of traditional quartz, which are still used in watches. Piezoelectric devices generate an alternative voltage, which has to be rectified to be used by electronic systems. However, piezoelectric devices are usually designed to give a maximum output power at a specific resonant frequency, and thus have to be properly matched to the mechanical energy source.

The mechanical structure of the piezoelectric device is directly linked to its efficiency, its frequency and its output voltage. The output energy can typically be amplified with the use of a weighted cantilever [46][47]. However, a compromise has to be made to maximize output power and minimize the form factor at the same time. This is especially true for small systems for which the size of the energy harvester is an important factor. In order to ease integration of such energy harvesters, the design of MicroElectroMechanical Systems (MEMS) piezoelectric generators has been considered [48][49]. Although these implementations can easily be integrated in constrained space, their output power, respectively 16 and 66.75 μW peak power for the two cited solutions, limits their application.

Despite their mechanical integrations constraints, piezoelectric generators have found multiple use-cases. Indeed, vibrations sources can be found in industrial context, e.g. produced by machines such as motors, residential areas with the vibrations from the compressor of a refrigerator, or multiple machines powered objects such as cars. An industrial solution which can be used anywhere is proposed by EnOcean [50], which gathers enough energy from a button pressure to send a radio transmission.

Another source of mechanical energy is the human body. For example, multiple works [51][52] have explored the use of piezoelectric generators embedded in footwear to generate electricity from steps, which can then power a wearable device, typically for health monitoring or sportive measurements. The use of piezoelectric generators has also been considered as a power source for biomedical devices. In [53], the use of an *in vivo* wireless implant sensor powered by a piezoelectric is proposed to detect excessive implant wear and/or incipient failure in orthopedic implants. By using the knee movement to power the sensor, the state of the implant can be monitored and the implant can be changed before failure, which reduces both medical costs and patient trauma. [54] proposes the use of piezoelectric generators implanted on the heart, lung and diaphragm of a patient to continuously power a pacemaker, for which a power failure can be fatal to the patient.

1.1.6 Other Energy Sources

Outside of the previously detailed energy sources, there exists multiple ways to harvest energy from the environment. These sources are less used in industrial applications, due to their limited power output and/or limited use-cases. For example, [55] proposes the use of ultrasonic-based wireless power transmission for implantable sensors. A 2-D MEMS energy harvester is designed to harvest energy over a 302 Hz bandwidth around 38.5 kHz. However, this energy harvester only delivers 22.7 nW at most, which is not sufficient to power a sensor node.

Energy can also be harvested from vibrations without the use of piezoelectric materials. Electrostatic energy harvesters rely on a variable capacitor. Vibrations are used to move a mobile plate of the capacitor, and thus modifying its capacitance. [56] shows an example where an electrostatic energy harvester is used to power an array of active pH sensors, which aims to be powered by human motion. Electrostatic energy harvesters can be fabricated using standard Complementary Metal-Oxyde-Silicium (CMOS) process, and thus can be integrated with electronic components, and even on flexible substrate such as in [57]. However, they require an external power source to initially charge the capacitor, which limits their applications.

Another way of harvesting vibrations takes advantage of Faraday's law of electromagnetic induction. Electromagnetic energy harvesters use the vibrations to move a magnet in the magnetic field created by a coil, producing an induced voltage. [58] presents an example of such a system which harvests 54 Hz vibrations to produce up to 115.1 μW. Another example presented in [59] uses a mechanical barrier structure to increase 10 Hz external vibrations to 394 Hz, increasing the efficiency of the electromagnetic harvester to generate 88.6 mV and 544.7 μW RMS power. Electromagnetic harvesters have a high relliability, due to the absence of mechanical contact in the harvesting process, but are more difficult to integrate than electrostatic energy harvesters.

Another mechanical energy harvesting source uses the water flow to generate electricity. [32], for example, uses a commercial hydrogenerator attached to a pipe in a field to generate up to 18 mW. Water flow energy harvesting is not much used, notably because the installation of a hydrogenerator implies the alteration of existing pipes, which can be costly and risky.

Another way to use water as an energy harvesting source is to use naturally occurring chemical reactions. Seawater batteries are an example of such energy harvesters, which can be used, for example, to monitor the structure of offshore oilrigs [60]. In such a system, seawater is used to create a battery with two electrodes, providing power to the system, making it a good solution for applications in seas and oceans. Other alternatives include the use of Microbial Fuel Cells (MFC), where electricity is generated from the microbial activity of the environment. [61] uses the microbial activity present in sediments at the bottom of lakes, rivers and oceans to generate up to 3.8 mW. Some other applications of this technology, such as a robot fed by sludge which can dispose of its own organic waste [62], a mobile phone powered from urine [63] or a WSN node [64], show the potential of this technology for power generation.

1.2 Power Conversion

Different power converter architectures can be used to regulate the energy from the energy harvesting source voltage V_{IN} to a suitable output voltage V_{OUT}, each with their advantages and shortcomings. The simplest architecture is the Low DropOut (LDO) regulator. In this architecture, shown in Fig. 1.6, a feedback loop is used to regulate a pass transistor between the energy source and its load. This system is cheap, simple, and provides a clean power rail for sensitive electronic components such as analog components. However, as the current is the same at the input and output of the circuit, this system has a limited efficiency $(V_{OUT} - V_{DROP})/V_{IN}$ where V_{DROP} is the voltage drop across the pass transistor. This inefficiency makes LDOs unsuitable for small scale energy harvesting systems where very few power is harvested and losses have to be minimized, although they can be used in systems where more energy is harvested [65]. Moreover, this regulator can only reduce a voltage to another, which makes them unsuitable for multiple low voltage energy sources such as small solar panels and TEGs. [66] uses LDOs to reduce the quiescent power consumption of the power converter, but this system can not extract power from a low voltage energy harvester and can only be used with sources reaching at least

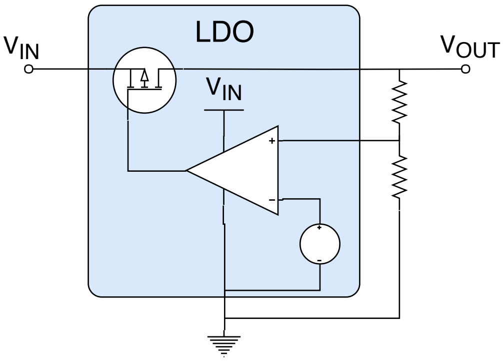


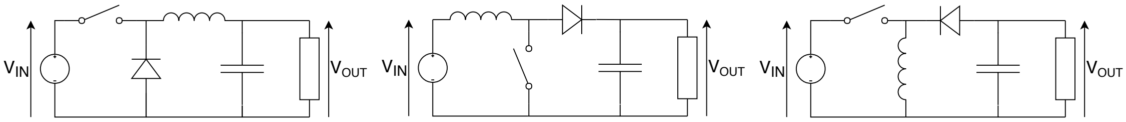
Figure 1.6: Low DropOut (LDO) regulator architecture.

3.2 V.

An alternative to linear regulators is to use a Switched Mode Power Supply (SMPS). In this kind of system, a switch between the energy source and the energy load is rapidly opened and closed. The switching frequency and the duty cycle (i.e. the percentage of time the switch is closed over a switching period) are usually regulated to provide a constant output voltage. Multiple topologies exist, which enable implementation of buck (i.e. $V_{IN} > V_{OUT}$, see Fig. 1.7a), boost (i.e. $V_{IN} < V_{OUT}$, see Fig. 1.7b) or buck-boost (i.e. $V_{IN}^{min} < V_{OUT} < V_{IN}^{max}$, see Fig. 1.7c), where an inductor is used as a storage element to smooth the current output and a capacitor is used to smooth the voltage output. Charge-pumps are an alternative topology which relies on capacitor-diode voltage multipliers, and are usually used for low currents. SMPS can easily provide good conversion efficiency, from 70% to 90%, but are more complex and expensive than LDOs and charge pumps, take more space due to the use of external components (e.g. inductor) and generate noise, which can disturb sensitive analog and radio components.

As SMPS can boost a low harvested voltage (e.g. from a TEG) to a higher voltage and have better efficiency than linear regulators, they are usually used in small scale energy harvesting systems. [67] proposes the use of a charge pump for indoor solar energy harvesting, but the output power is very limited, from 0 to 80 μW. In [68], a buck-boost converter is used to provide multiple regulated power rails directly from an indoor photovoltaic energy harvester. This avoids using multiple regulators, reducing the system size and increasing its global efficiency. In [69], a buck regulator is associated with an active bias-flip rectifier for piezoelectric energy harvesting, and a boost regulator is used to power internal components of the IC. In order to mitigate the size of the system, one of the drawbacks of SMPS, the authors propose the use of a shared inductor between the rectifier, buck and boost regulators, with an arbiter managing access to the inductor. For ultra-low voltage energy sources such as TEG, [70] proposes a boost converter which accepts inputs from 20 mV to 250 mV, but its output capacitor has to be pre-charged to 650 mV before it can start operating.

In order to reduce the voltage required by a boost IC to start-up, [71] proposes a two-stage circuit, where a first boost converter is used to generate energy from a voltage as low as 0.5 V with a 2 μA current. This energy is used to start-up a second, more efficient boost regulator, which is used to power the load and charge the battery. However, this system requires two inductors, which increase the size of the solution. [72] uses a similar architecture, but the use of a charge pump as



(a) Buck circuit. (b) Boost circuit. (c) Buck-boost circuit.

Figure 1.7: Switched-mode power supply architectures.

the start-up circuit avoids the use of a second inductor. Industry contributions have also been made, such as [73] where a boost converter which can start from 330 mV and 5 μW is proposed for solar and thermoelectric energy sources. Based on these works, semiconductor manufacturers have released multiple off-the-shelf components, such as the LTC3108 [74] from Linear Technology, the BQ25570[75] from Texas Instruments or the SPV1050 [76] from STmicroelectronics.

1.3 Storing Energy

The choice of energy storage has an important impact on the lifetime and efficiency of the energy harvesting WSN node. Embedded systems usually rely on Nickel-Metal Hybrid (NiMH) batteries [11] or lithium based batteries [77] to store energy. NiMH batteries provide a voltage between 1.1 and 1.4 V with a limited self-discharge, are packaged in standard form factors and are cheap. Lithium based batteries have a higher nominal voltage. Standard Lithium-ion (Li-Ion) and Lithium-polymer (LiPo) batteries provide a voltage ranging from 3.2 V to 4.2 V, and Lithium-Iron-Phosphate (LiFePo) batteries provide a constant 3.2 V voltage. As electronic components are usually powered from 3.3 V or 1.8 V power rails, the choice of lithium based batteries enables using buck regulators to lower the voltage, which are generally cheaper and less power-hungry than boost regulators.

The lifetime of a rechargeable battery is counted as the number of charge cycles before it loses 20 % of its nominal capacity. A charge cycle is considered as a complete discharge at standard discharge current, followed by a complete charge at the maximum charge current. NiMH batteries typically have a lifetime of 500 charge cycles, while Li-Ion/LiPo and LiFePo batteries can hold around 1000 and 2000 charge cycles, respectively. However, it is possible to increase the lifetime of rechargeable batteries by discharging and charging them with a lower current, or stopping the charge/discharge before the battery is completely full/empty. It is commonly estimated that dividing by $\mathcal{N}$ the charge/discharge current multiplies by $\mathcal{N}$ the maximum number of charge cycles. Moreover, even when the battery has reached its maximal number of charge cycles, it is still functional, although with lower performance and capacity than specified.

The maximum discharge current of a rechargeable battery is usually equivalent to its capacity $\mathcal{C}$, i.e. a 2400 mAh battery will deliver up to 2.4 A before risking damage. The charge method, however, differs depending on the chemistry of the battery. The simplest safe charging method is the trickle charge, where a low current (up to $0.1\mathcal{C}$) is used to recharge the battery up to its maximal voltage. For NiMH batteries, the voltage of the battery is monitored, and the battery is considered to be charged when a voltage drop is detected. If this voltage can be reliably detected, the charge

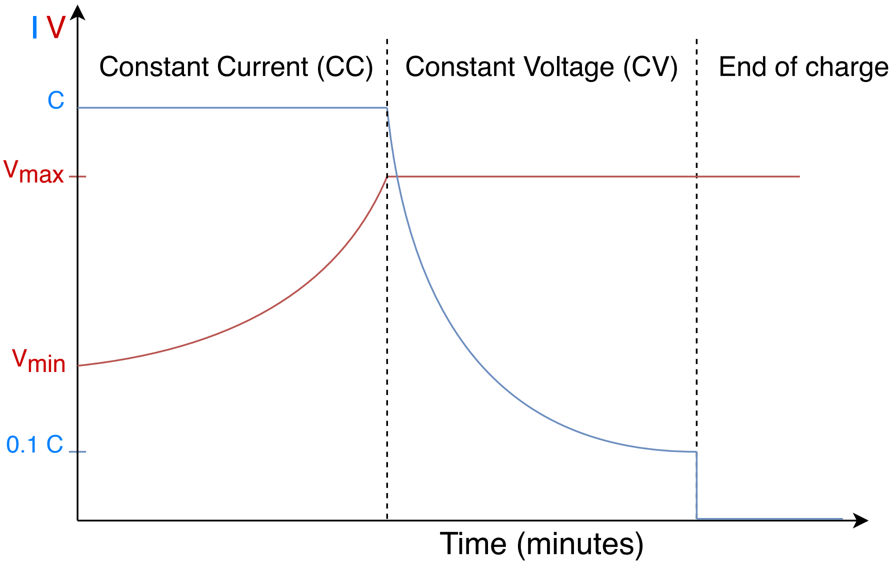


Figure 1.8: Illustration of a CC/CV charging scheme.

current can be increased up to $1\mathcal{C}$. Lithium based battery are charged with a two step process, illustrated in Fig. 1.8. In a first step, a Constant Current (CC) of up to $1\mathcal{C}$ is applied to the battery, until it reaches its maximal voltage. In the second step, a Constant Voltage (CV) equal to the maximum voltage of the battery is applied until the charging current becomes lower than a threshold, usually fixed to $0.1\mathcal{C}$. In the case of LiFePo batteries, the use of the CC step only is enough to recharge the battery to 95 % of its State of Charge (SOC) [78].

Due to the popularity of rechargeable batteries, a number of off-the-shelf ICs are available to properly manage the charge and discharge of rechargeable batteries. Such components can be used to increase the durability of the battery by limiting its charging current. Some of these devices also include circuitry to estimate the SOC of the battery. Indeed, the state of charge of these batteries can not be directly measured, and must be estimated by measuring the current from and to the battery. Another possibility is to measure the battery voltage and calculate the SOC based on the charge/discharge curves of the battery, but this method is less accurate and is specific to each battery model. Finally, most battery management ICs include circuitry to detect and prevent damages on the battery. Indeed, due to electrical, thermal and mechanical constraints, batteries can become unstable and cause damage to the device or user [79].

Energy harvesting devices repeatedly charge and discharge their energy storage. Thus, the limited number of charge cycles of usual batteries is concerning for long term deployment of energy harvesting WSN. An alternative is to use supercapacitors, also sometimes called ultra-capacitors. Supercapacitors are simply capacitors with high capacitance. Thus, they behave like usual capacitors. Compared to usual rechargeable batteries, they have a lower energy density and a much higher self-discharge rate, which is specified in days or weeks compared to months for NiMH and lithium based batteries. On the other side, supercapacitors can withstand up 100.000 to 1.000.000 charges cycles and deliver a higher current than conventional batteries. In the context of supercapacitors, C refers to the capacitance of the device,

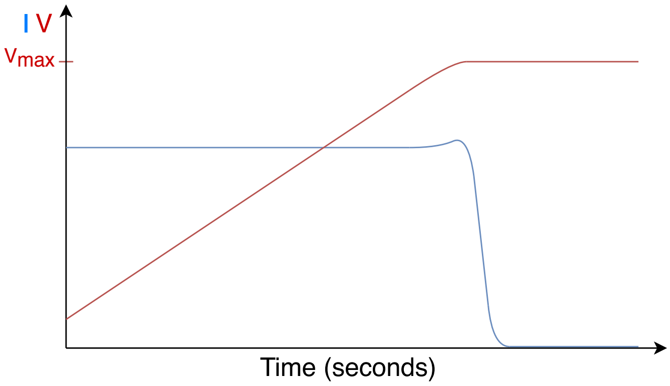


Figure 1.9: Supercapacitor charging curve.

and is measured in Farads (F). The SOC of a supercapacitor of a capacitance C can easily be estimated by measuring its voltage V_C and calculating the stored energy E_R as $E_R = 0.5 \times C \times V_C^2$. These characteristics make them suitable for energy harvesting WSN.

Supercapacitor voltages are usually limited to a few volts. Moreover, using supercapacitors near their maximum voltage increases the self-discharge current. Multiple supercapacitors can be connected in series in order to create a higher voltage energy storage. However, a balancing circuit has to be used to prevent charge imbalance between the different capacitors and to increase the lifetime of the supercapacitors [80]. This circuit raises the cost of the storage and reduces its efficiency [81], thus a compromise has to be made. It is also possible to connect supercapacitors in parallel to increase the global storage capacity, but this increases the self-discharge of the global storage.

Charging supercapacitor is straightforward and is similar to the charge of a capacitor. The charge profile of a supercapacitor is shown in Fig. 1.9. The voltage of the supercapacitor rises linearly while it draws as much current as it can from the power supply. Once the supercapacitor voltage reaches the power supply voltage, it starts to draw less current, until it is completely charged. This charge process is simple to implement and can be very quick if there is sufficient current. However, multiple works [82][12] show that charging a supercapacitor with pulses of energy instead of a continuous power supply increases the charging efficiency. It can be noted that, due to that charge redistribution in the supercapacitor, these energy storages tend to demonstrate a memory effect, where the voltage increases without an external power source after a sudden discharge or decreases after a sudden charge [83].

Energy storage has long been a limiting factor when reducing the size of embedded systems. In order to enable smaller systems, a number of manufacturers has developed thin-film batteries [84]. Moreover, in order to bridge the gap between rechargeable batteries and supercapacitors, research has begun on the design of hybrid storage called Li-Ion capacitors [85]. The goal of this work is to create an energy storage which can withstand a large number of charge cycles while keeping a limited self-discharge rate. Even if some commercial products have been launched [86][87], these components are still rare, hard to source, and thus are difficult to use in industrial applications. Table 1.1 shows an overview of the advantages and drawbacks of

	Supercapacitor	Li-Ion capacitor	Lithium battery
Energy density	Low	Medium	High
Internal resistance	Low	Medium	High
Charge duration	Very fast	Fast	Medium
Self-discharge	High	Low	Very low
Charge cycle lifetime	∼100k	∼10k to 100k	Up to ∼2k
Safety	High	High	Medium
Cost	High	Very high	Low

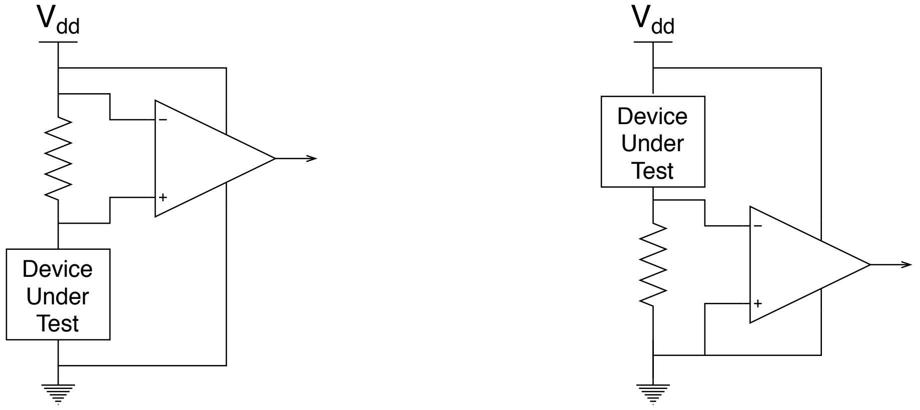
Table 1.1: Comparison of energy storages for WSN nodes.

the different energy storage choices. As can be seen, there is no “one size fits all” solution, and the choice of energy storage has to be adapted to the design constraints, especially in terms of cost, space and performance.

1.4 Measuring Energy

Recent systems rely more and more on batteries, thus energy consumption measurement has become more and more important. In particular, measuring the power consumption of different parts of the system enables a better knowledge of the global power consumption. As these measurements require active circuitry, multiples work propose to use simulation and model-based energy consumption estimations to get more information on the system power consumption. [88] and [89] propose models to estimate the power consumption of acquisition, communications and processing elements of the node. These models rely on measurements of the WSN node hardware components and application profiles to give a more accurate estimation, once these design choices are defined. Other models [90][91] use instruction level energy consumption models to calculate the power consumption of the system based on its application binary, but do not take into account external components, such as sensors and communications ICs. All these works can be used as power consumption estimation tools during the design process, but can not be used to measure the harvested energy, nor the immediate power consumption.

The easiest way to measure the power consumption P_C is to measure the current consumption I_C and multiply it by the system supply voltage V_{DD}, as $P_C = I_C \times V_{DD}$. To measure the current, a shunt resistor can be set in series on the power rail, creating a voltage drop of $I_C \times R$ where R is the shunt resistance in Ω. High-side circuit (Fig. 1.10a) requires a special amplifier to support the high common mode voltage. Low-side circuit (Fig. 1.10b) can use a cheaper amplifier, but the resistor creates an unstable ground reference. In the case of WSN nodes, this drop is often small and has to be amplified before being measured, typically by an Analog to Digital Converter (ADC) like in [92], which adds a power consumption overhead to the system. [93] reduces this power consumption overhead by using a Voltage to Frequency Converter (VFC) and hardware counters to quantify the energy flowing through the shunt resistor. However, the power consumption of this system is not quantified, and its 25$ cost is high for industrial applications. This approach is also used in off-the-shelf components such as the LTC4150 [94], which consumes up to



(a) High-side shunt resistor current measurement circuit.

(b) Low-side shunt resistor current measurement circuit.

Figure 1.10: Shunt resistor circuit configurations.

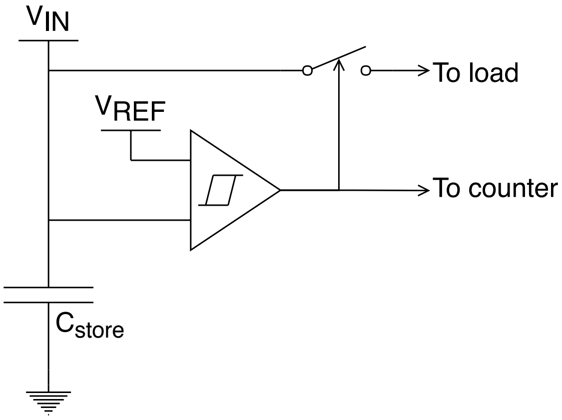


Figure 1.11: Schematic of a basic coulomb counter. A comparator periodically empties a known amount of energy from C_{store} capacitor. The number of impulsions is counted to calculate the total energy from the input to the load.

140 μA at 8.5 V. The precision of such systems can however be limited, as the shunt resistor size is a compromise between the smallest measurable voltage drop and the maximum acceptable voltage drop on the power rail.

Another approach is to use a coulomb counter to count the electrical charges flowing from the source to the load, as shown in Fig. 1.11. For example, the BQ2019 [95] uses this type of approach to measure the long-term charge and discharges of batteries in mobile systems. An original implementation of such a system is presented in [96], which uses switched capacitors to measure a processor power consumption at the instruction level. However, this system is implemented using a FPGA and a host PC, and can not be used on a WSN node. An interesting circuit is presented in [97]. Using small capacitors as intermediate energy buffers and dynamic threshold comparator, this system implements an efficient coulomb counter. Its low power consumption (1.7 μA) and high precision, more than 90%, make it suitable for ultra-low power WSN node. The off-the-shelf LTC3335 [98] is another original implementation of a coulomb counter, which is integrated with a buck-boost converter to monitor

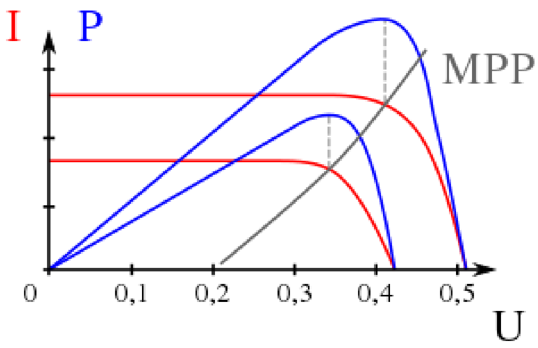


Figure 1.12: MPP of a solar cell for two different illumination - Image from Wikimedia Commons[a], distributed in public domain

---

[a] https://commons.wikimedia.org/wiki/File:MaximumPowerPoint.svg

the converted energy.

1.5 Maximizing the Harvested Energy

Depending on their load, energy harvesting sources can deliver significantly more or less power. Therefore, it is important to match the input impedance of the power converter to the output impedance of the energy source. When these impedances are matched, the energy harvester delivers its maximum power output. This point is called the Maximum Power Point (MPP). However, the characteristics of an energy source varies with its environment, and the MPP is modified. Thus there is a need to develop Maximum Power Point Tracking (MPPT) techniques, circuits and algorithms to efficiently track the variations of the MPP. In particular, MPPT implementations manage the power converter so that the voltage of the energy harvesting source is equal to V_{MPP}, the voltage of the MPP. Although different energy harvesters have different I-V characteristics curve (cf. Fig. 1.12) and thus different MPP, similar MPPT techniques can often be applied.

It can be noted that, due to the popularity of solar energy harvesting for both small scale WSN systems and large scale solar power generators, a large number of MPPT systems has been tested with solar panels. However, solar power plants are subjects to more constraints than small scale devices. In particular, large solar panels and solar panel arrays can be subjects to partial shadowing, where only a part of the panel or array is shadowed [99]. Depending on the arrangement of the panel or array, this can create multiple local MPPs. Thus, complex algorithms are required to sort the different local MPPs and determine which one effectively delivers the most power. Energy harvesting WSN nodes, which use smaller panels, are not affected by this phenomenon. This relaxed constraint enables the use of significantly simpler and cheaper MPPT implementations.

[100] shows an early implementation of an MPPT technique. In this work, the I-V curve of a photovoltaic panel is periodically swiped to find the MPP location. Although this technique can be applied to a vast variety of energy harvesters and is not affected by partial shadowing, this technique requires significant resources to

measure both voltage and current provided by the source, store the measurements and process them. Moreover, swiping the I-V curve takes time during which the energy harvester is inefficient, but the process has to be repeated as often as possible to stay locked on a accurate MPP value. Therefore, a design trade-off has to be made.

Two methods, Perturb & Observe (P&O) and Hill Climbing [101], simplify the MPPT by supposing that the power curve $P = f(U)$ has a single maximum and no local maxima. Thus, the power curve is supposed to be rising while the MPP has not been reached, and falling once the MPP is reached. These methods periodically try to move the current operating point, and measure the resulting power from the energy harvester. If the measured power is higher than the previous measurement, the algorithm keeps moving the operating point in the same direction, and changes direction otherwise. This algorithm is able to dynamically track the MPP but requires significant components to measure the harvested power. This increases the cost and power consumption of the solution, although some lower cost analog implementations can be imagined [102].

A low complexity method is the Fractional Open Circuit Voltage (FOCV) method, presented in [103]. This method is based on the observation that the voltage of the MPP V_{MPP} can often be approximated by multiplying the open-circuit voltage V_{OC} of the source by a coefficient K_{focv} fixed between 0 and 1 (around 0.75 for a solar source and 0.5 for a TEG). Thus, the energy source is periodically disconnected in order to sample V_{OC}. V_{MPP} is then calculated from V_{OC} and used as a reference for the power converter until the next V_{OC} sampling. This method, shown in Fig. 1.13, does not dynamically track the MPP, and induces some power losses since the energy source is periodically disconnected for a short time. However, its implementation is simple and low cost, and can easily be adapted for a new power source simply by modifying the K_{focv} coefficient. Moreover, no current measurement of the source is required, which reduces the number of components to implement this technique. [104] has presented another method for MPP estimation which relies on V_{OC} measurement, but this technique only slightly improves the MPP tracking efficiency (99.2 % *vs.* 98 % for FOCV at 25 °C) while being significantly more complex.

Due to the tight coupling between the power converter control and the MPPT circuit, a number of circuits has integrated the MPPT implementation directly with the power converter control circuit. This is the case in multiple research works [105][106][107], but also in multiple industrial off-the-shelf solutions [76][75]. It can be noted that the later implementations rely on the FOCV method, due to its simplicity and the possibility to adapt it to a vast panel of energy sources.

1.6 Electronic Components Lifecycle

Provided with proper power management, energy harvesting enables the deployment of WSN for a theoretically infinite duration. However, practical lifetime is limited by the wear life of the components. Thus, a long term WSN deployment has to take into account the dependability of each element to ensure that the deployed sensors run as long as required.

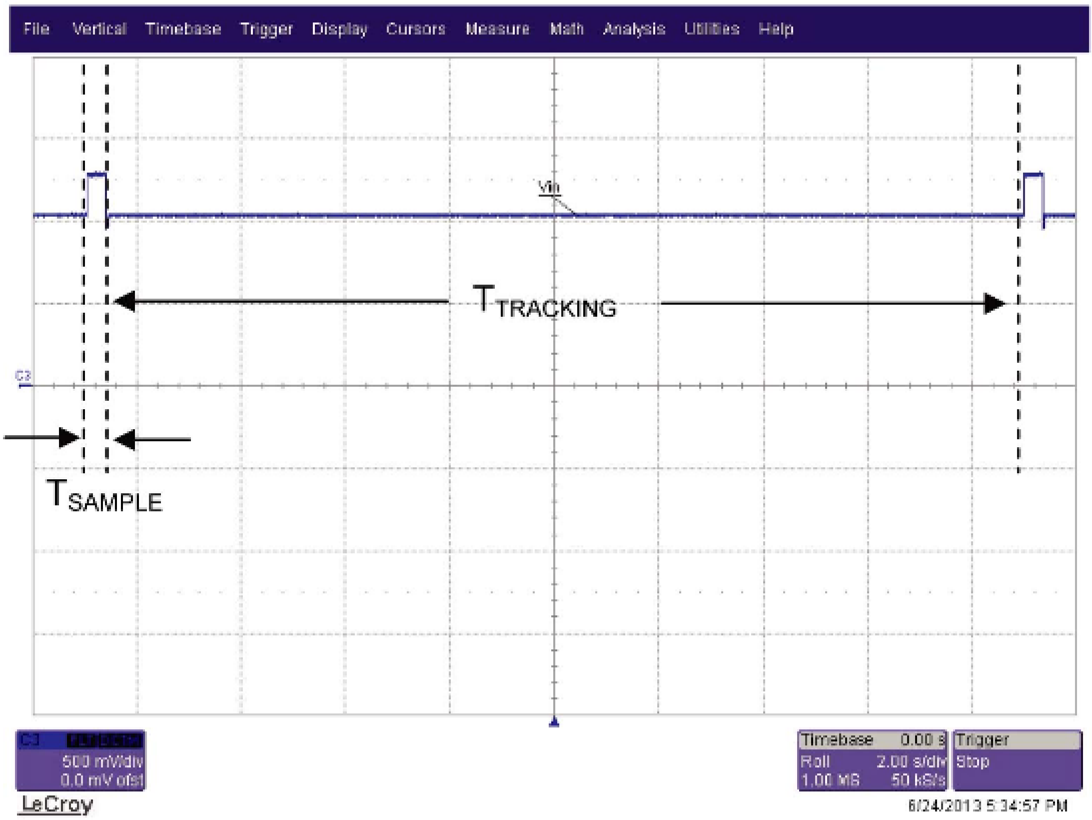


Figure 1.13: Illustration of the FOCV MPPT with $V_{OC} = 1.25V$ and $V_{MPP} = 1.05V$ from the SPV1050 [76] datasheet.

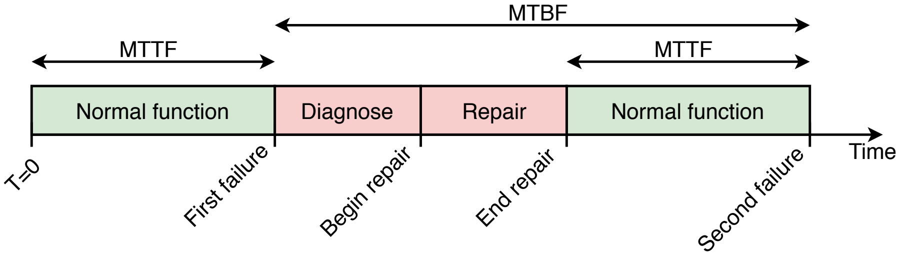


Figure 1.14: Comparison of MTBF and MTTF of a product.

1.6.1 Electronic components dependability

Dependability is an engineering field which aims to characterize the availability, reliability, maintainability and security of a system. When estimating the lifetime of a system, a particular attention is paid to its reliability, defined as its capacity to perform its expected function during a time interval, in specified conditions. Reliability is often expressed as the Mean Time Between Failures (MTBF) of a system, although this notion can only be applied to repairable systems. For non-repairable systems, the lifetime of the product is expressed with the Mean Time To Failure (MTTF), which characterizes the mean duration between a repair and the next failure. The difference between the two notions is shown in Fig. 1.14. Electronic components are considered as non-repairable, as the manpower and machinery required to diagnose and repair an IC is significantly more expensive than simply replacing the component. By extension, a low cost device such as a WSN node can also be considered as non-repairable.

In the context of reliability, failures are defined as events which prevent the device from operating as specified. It can be noted that this definition can vary depending on the applications. For example, lithium batteries lifetime is specified as the number of charge cycle before they lose 20 % of their nominal capacity. However, a battery with 80 % of its nominal capacity is still functional and may still be used in an energy harvesting WSN node with proper power management, especially if the energy storage has been properly sized to anticipate its wear. Moreover, the lifetime specified in charge cycles is specified at specific temperatures and does not take into account the mechanical and thermal constraints applied to the system. This illustrates the complexity of reliability estimation.

MTTF and MTBF can be measured in-field, by gathering reliability data of deployed units. However, this approach is severely limited. In order to gather statistically reliable data, a very large number of products has to be deployed and monitored, for a significant amount of time before being able to advertise a certain reliability. Multiple standards have been proposed which define methodologies for electronic components MTBF estimation. There is no such method to estimate the MTTF of electronic subsystems. Due to this, MTBF is often used as an equivalent to MTTF for non-repairable systems [108].

The oldest MTBF estimation methodology for electronic components, MIL-HDBK-217 [109], was developed in the early 1980s and last revised in 1995. Although it tends to underestimate the reliability of recent component technologies due to its age, it is still one of the most popular MTBF estimation method. The ANSI/VITA 51.1

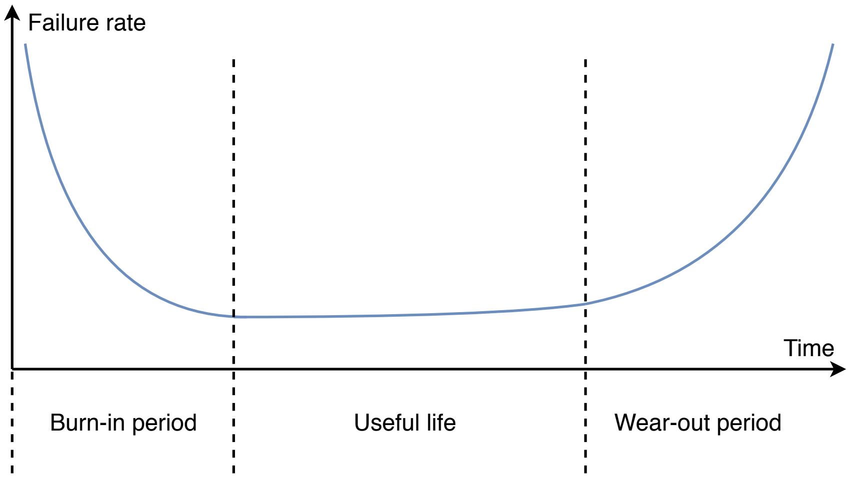


Figure 1.15: Evolution of the failure rate during the life-cycle of a product.

specification aims to complement the MIL-HDBK-217 for more recent components. Other methodologies include the Telcordia SR-332 method, which tends to be simpler and more up-to date, or the IEC 61709:2017 developed by the French industry which tends to be more complex and much less used. Out of these methodologies, only the MIL-HDBK-217 original specification is easily accessible, while engineers must pay a charge to access the others, making it a viable choice despite its age. Depending on the used parameters and the environmental conditions of the system, the efficiency of these methodologies can vary greatly [110]

It can be noticed that these methodologies assume that the system operates at a stable failure rate. However in reality, the failure rate of a system is sensibly higher during the beginning of its service (infant mortality) and its end of service (wearout region). The evolution of the failure rate of a system is illustrated in Fig. 1.15. Moreover, these methodologies only take into account failure modes related to electronics, and mechanically induced failures, such as solder cracking, are not considered. Despite these limitations, these methodologies are still widely used to give an estimation before deployment of a product lifetime and to identify the points of failure.

1.6.2 MTBF/MTTF of a WSN node

In order to simplify the reliability estimation of a WSN node, MTTF and MTBF will be considered as equivalent. This estimation relies on the WSN node model presented in Fig. 1.16. In addition to the components used to implement this model, real world circuits implement a number of passive components: resistors, capacitors and inductors. A huge variety of energy harvesting sources can be used, each with their own electrical, mechanical and thermal constraints which impact the MTTF of the harvester. Moreover, the environment has a huge impact on the lifetime of the source. This makes it difficult to model the energy source, and explains why very few modeling tools for energy harvesting devices exist. Thus, the energy harvesters will not be included in this study.

Electronic components lifetime can be evaluated through the MIL-HDBK-217

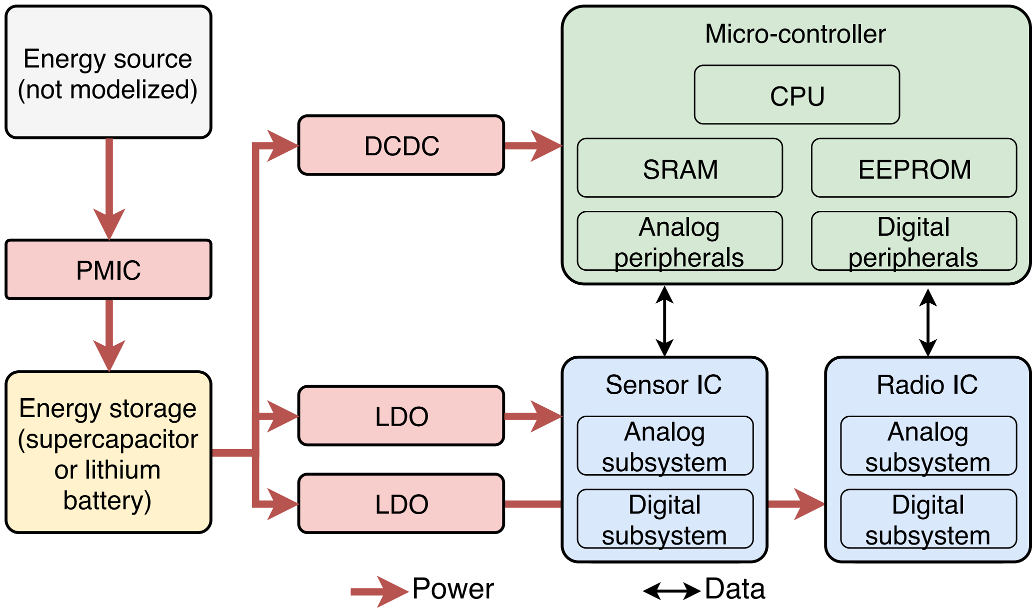


Figure 1.16: WSN model used for the lifetime estimation of components.

framework, a military standard used to calculate the reliability of military equipments. Each component model is associated with a base failure rate λ_b, associated with an environment factor π_E, a quality factor π_Q and a temperature factor π_T, calculated with a formula adapted to the component. Different models can also have more specific associated risk factors. Thus, active components take into account the age of the component through a learning factor π_L, resistors use a power factor π_P and a power stress factor π_S, and capacitor models include a capacitance factor π_C, a voltage stress factor π_V and a series resistance factor π_{SR}. These risk factors π_i are calculated according to formulas and abacus which are specific for each component model. The complete failure λ_p, expressed in failures/10^6 Hours, is calculated as the product of the base failure rate and all risk factors π_i that apply:

$$\lambda_p = \lambda_b \times \prod \pi_i \tag{1.1}$$

The lifetime L of the component in hours as a function of the failure rate λ_p can then be calculated as:

$$L(\lambda_p) = \frac{10^6}{\lambda_p} \tag{1.2}$$

All components lifetimes are calculated for temperature ranging from -20 °C to 55 °C. The use of commercial quality components is assumed in a Ground Fixed (GF) environment, as defined by the MIL-HDBK-217 standard.

Active components models used in this lifetime evaluation are shown in Table 1.2. Power components are modelled as purely analog ICs, where a SMPS converter is assumed to be more complex than a LDO, and an energy harvesting Power Management Integrated Component (PMIC) is more complex than a SMPS due to the added functionalities. Due to the complexity of modern ICs, radio communications and sensors ICs are modelled as hybrid components composed of an analog subsystem, corresponding to the core functionality of the IC, and a digital subsystem which

Component	Parameter	Value
LDO	Die Complexity Failure Rate	MIL-HDBK-217 section 5.1, MOS Linear IC with 101 to 300 transistors
	Number of functional pins	3 to 8 pins, non-hermetic DIP, PGA or SMT package is assumed
SMPS	Die Complexity Failure Rate	MIL-HDBK-217 section 5.1, MOS Linear IC with 301 to 1000 transistors
	Number of functional pins	3 to 10 pins, non-hermetic DIP, PGA or SMT package is assumed
PMIC	Die Complexity Failure Rate	MIL-HDBK-217 section 5.1, MOS Linear IC with 1001 to 10000 transistors
	Number of functional pins	6 to 24 pins, non-hermetic DIP, PGA or SMT package is assumed
Radio IC	Die Complexity Failure Rate	MIL-HDBK-217 section 5.5, hybrid microcircuit composed of an analog and digital subsystem
	Number of functional pins	Not applicable
	Circuit Function Factor	Microwave, f >1GHz
Radio IC analog subsystem	Die Complexity Failure Rate	MIL-HDBK-217 section 5.1, MOS Linear IC with 1001 to 10000 transistors
	Number of functional pins	Not applicable as part of a hybrid microcircuit
Radio IC digital subsystem	Die Complexity Failure Rate	MIL-HDBK-217 section 5.1, MOS Digital IC with 3001 to 10000 gates
	Number of functional pins	Not applicable as part of a hybrid microcircuit
Sensor IC	Die Complexity Failure Rate	MIL-HDBK-217 section 5.5, hybrid microcircuit composed of an analog and digital subsystem
	Number of functional pins	Not applicable
	Circuit Function Factor	Linear, f <10MHz
Sensor IC analog subsystem	Die Complexity Failure Rate	MIL-HDBK-217 section 5.1, MOS Linear IC with 301 to 1000 transistors
	Number of functional pins	Not applicable as part of a hybrid microcircuit
Sensor IC digital subsystem	Die Complexity Failure Rate	MIL-HDBK-217 section 5.1, MOS Digital IC with 3001 to 10000 gates
	Number of functional pins	Not applicable as part of a hybrid microcircuit

Table 1.2: Models used for active components lifetime assessment.

models the digital control and communications interfaces included in the component. Micro-controllers, which include a high number of functionalities, are modelled as a hybrid system composed of a CPU subsystem, a RAM memory block, a ROM memory block, a digital peripherals subsystem and an analog peripherals subsystem. The parameters of the different micro-controller subsystems scale with the CPU subsystem, so that an 8-bit CPU is associated with smaller memories and less complex peripherals than a 32-bit CPU. The different micro-controller configurations are as follows:

- **8-bits:** analog and digital peripherals subsystem is estimated with MIL-HDBK-217 section 5.1, respectively as a MOS linear IC of 101 to 300 transistors and a MOS digital IC of 1001 to 3000 gates, RAM and ROM memories are estimated with MIL-HDBK-217 section 5.2, respectively as MOS SRAM and MOS EEPROM of 16 kbits to 64 kbits, with 500 to 1000 EEPROM programming cycles during the life of the product.

- **16-bits:** analog and digital peripherals subsystem is estimated with MIL-HDBK-217 section 5.1, respectively as a MOS linear IC of 301 to 1000 transistors and a MOS digital IC of 1001 to 3000 gates, RAM and ROM memories are estimated with MIL-HDBK-217 section 5.2, respectively as MOS SRAM and MOS EEPROM of 64 kbits to 256 kbits, with 500 to 1000 EEPROM programming cycles during the life of the product.

- **32-bits:** analog and digital peripherals subsystem is estimated with MIL-HDBK-217 section 5.1, respectively as a MOS linear IC of 1001 to 10000 transistors and a MOS digital IC of 3001 to 10000 gates, RAM and ROM memories are estimated with MIL-HDBK-217 section 5.2, respectively as MOS SRAM and MOS EEPROM of 256 kbits to 1 Mbits, with 500 to 1000 EEPROM programming cycles during the life of the product.

The active component lifetimes calculated from these ranges are shown as box plots on a logarithmic scale in Fig. 1.17, with the median indicated on the graph. Fig. 1.18 shows how a box plot is traced.

Passive components use different modeling parameters than active components. Inductors are modeled as coils, and only use the temperature, quality and environment coefficients π_T, π_Q and π_E. Resistors are modeled as fixed film resistors. In IoT applications, where the power consumption must be as low as possible, there is always little power dissipated in resistors. Therefore, resistors are modeled for power dissipation ranging from 1 μW to 1 mW with 10 values per decade, and for a power stress, defined as the ratio between real dissipated power and maximal dissipated power, ranging from 0.001 to 0.05 with a 0.001 step between values. Capacitors are modeled as fixed value Multiple Layer Ceramic Chip (MLCC) capacitors, and the series resistance factor π_{SR} can thus be ignored and set to 1. The voltage stress is defined as the ratio between the effective capacitor voltage and the nominal capacitor voltage. In these calculations, it ranges from 0.05 to 0.65 with 0.05 steps between values. The tested capacitances range from 1 pF to 1 F, with 12 values per decade, calculated with the E12 Series from the IEC60063:2015 standard [111]. The component lifetimes calculated from these ranges are shown as box plots on a logarithmic

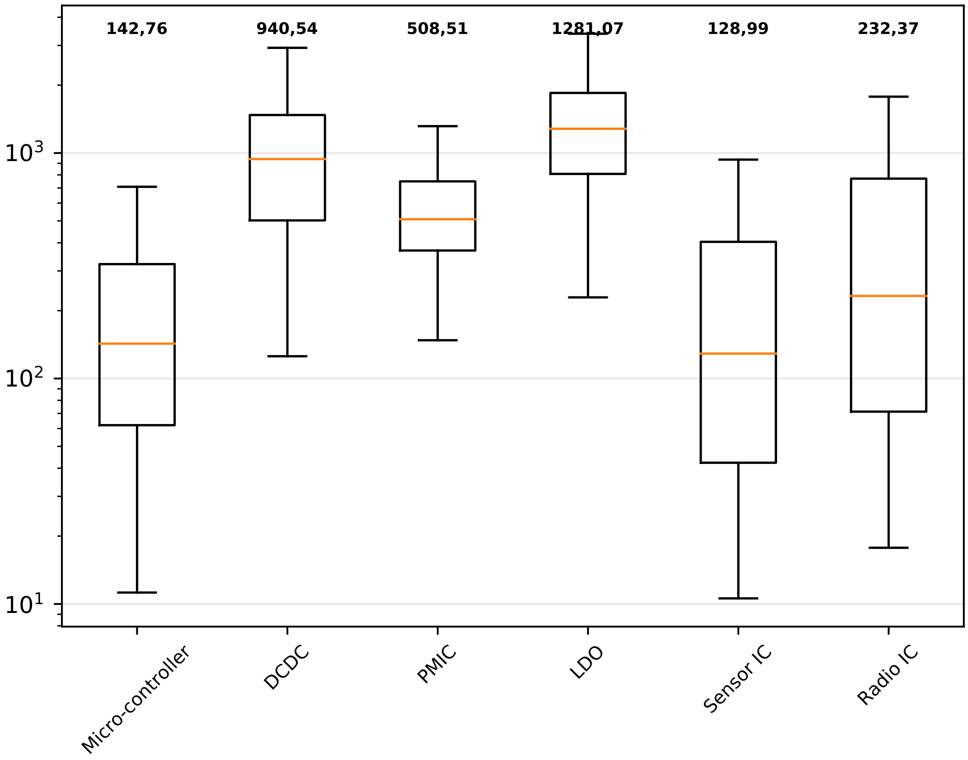


Figure 1.17: MTTF estimation, in years, of active components.

scale in Fig. 1.19, with the median indicated on the graph. As can be seen in this graph, passive components are resistant to wear-out, especially if they are properly dimensioned to avoid too much power of voltage stress.

Energy storages are not included in the MIL-HDBK-217 framework. Typically, lithium battery lifetime is affected by the environment temperature, its mean SOC, the depth of discharge of each charge/discharge cycle, the charging and discharging current, its age and mechanical constraints. Due to this complexity, there is still no MTTF model for lithium batteries which does not require consequent measurements. In [112], the authors propose a model which is able to predict the lifetime of the battery based on its temperature, mean depth of discharge, mean SOC and age. Although this model requires measurements to calibrate it to a specific battery model, these measurements are provided for the tested samples. This enables us to obtain

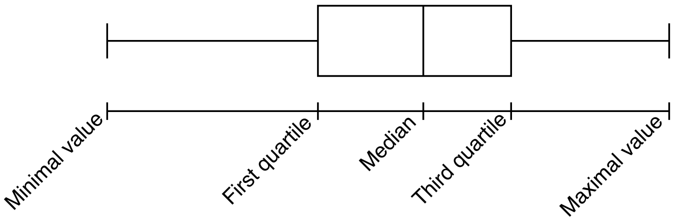


Figure 1.18: Box plot explanation.

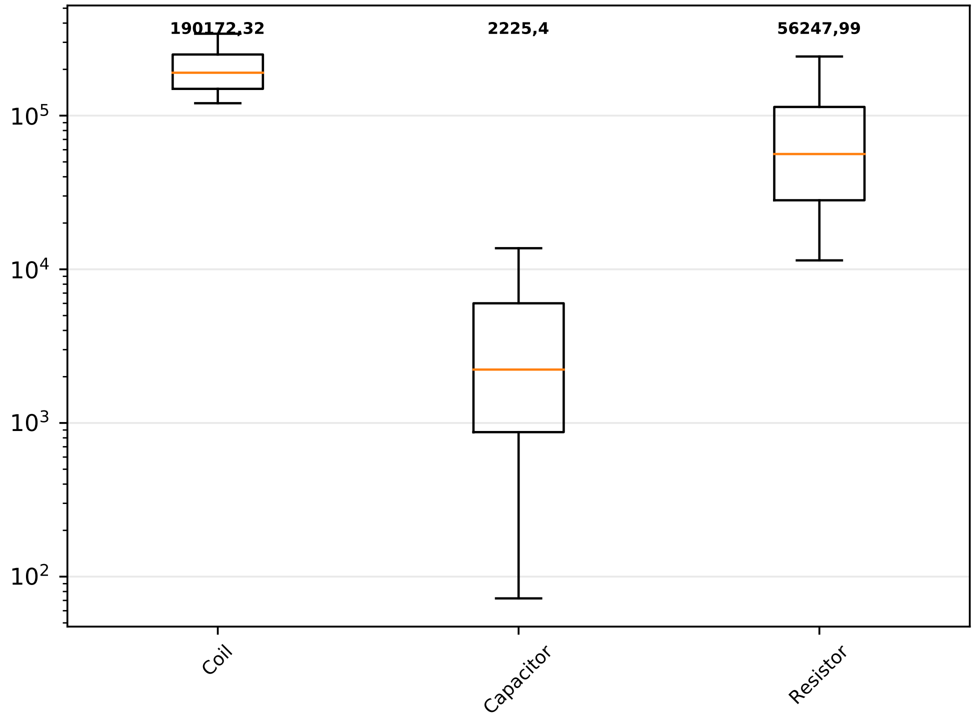


Figure 1.19: MTTF estimation, in years, of passive components.

lifetime results for different parameters, even though it only concerns a single specific reference. Due to the number of parameters in this model, it is necessary to reduce the number of input values to avoid long calculation times. Therefore, the tested temperatures range from -20 °C to 55 °C with steps of 5 °C instead of 1 °C. The age of the battery is linked to the duration of a discharge cycle, which is set between 10 s to 1 full day. The Depth of Discharge (DoD) varies from 0.01 (corresponding to 1 % discharge) to a full charge/discharge, which can be calculated from the SOC as $DoD_{MAX} = 1 - 2 * |SOC - 0.5|$.

Supercapacitors are simpler, as they are mainly affected by their voltage and the environment temperature and humidity. Even though no lifetime estimation model has emerged from the academic world, rough estimation models have been proposed by supercapacitor manufacturers, such as [113]. This model applies a voltage stress factor and a temperature factor to a base MTTF value of the supercapacitor, expressed as a couple $[L, T]$ where L is the number of hours before the supercapacitor is considered defective, and T the temperature used to characterize this MTTF. For this estimation, the temperature varies from -20 °C to 55 °C with 1 °C steps, and voltage stress ranging from 0.05 to 0.95 with 0.05 steps. These stress factors are applied to a range of $[L, T]$ tuple from commercial supercapacitor datasheets. The results of lifetime estimation for both lithium batteries and supercapacitors is shown in Fig. 1.20 on a logarithmic scale, with the median indicated on the graph. It can be noted that, for this estimation, the end-of-life of energy storage is assumed to correspond to an energy capacity reduced by 20 % compared to the nominal value of the storage. In many cases, the energy storage is still functional, although with degraded performance, and could still be used if the power management software takes into account the wear-out of the storage.

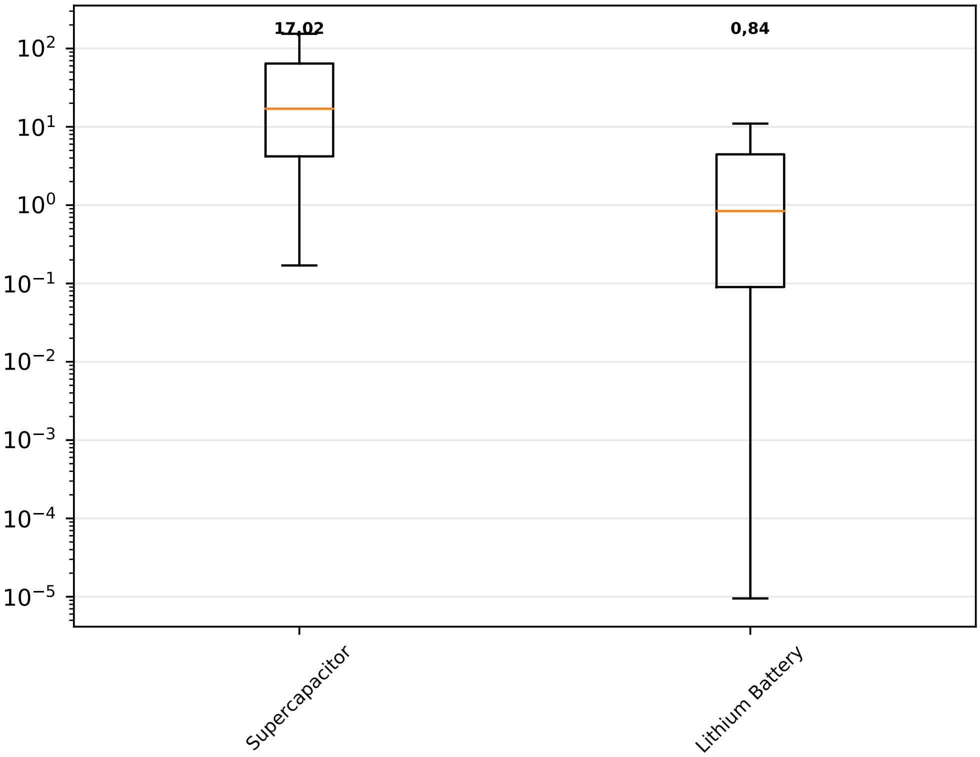


Figure 1.20: MTTF estimation, in years, of energy storage components.

Fig. 1.21 shows the MTTF of all components on the same graph, on a logarithmic Y-axis. It can be quickly observed that the energy storage is the weak point of energy harvesting devices. In particular, the use of lithium batteries severely impacts the lifetime of the node, due to the sensitivity of this component to temperature variations. For extreme use-cases, where the battery is subject to short (tens of seconds) deep charge/discharge cycles at extreme temperature, the wear-out of the battery is particularly low. However, this use-case rarely happens in real energy harvesting use-cases if the energy harvester and storage are properly dimensioned. This also shows the advantage of supercapacitors to design resilient energy harvesting systems.

This work also shows that, even if a power manager can be used to ensure a theoretically infinite lifetime to the node, the real lifetime of a node will still be limited in time by the wear-out of its components. Therefore, there is a risk for the lifetime of the node to be limited by poor hardware design choices instead of power management strategies. Thus, it is necessary, when designing an energy harvesting sensor node, to ensure the proper choice of component so that the wear-out of the system does not become the limiting factor. For use-cases which require an ultra-low cost, and thus lower quality components, it can be interesting to use the power manager as a mean to maximize the QoS of the node, instead of maximizing its lifetime.

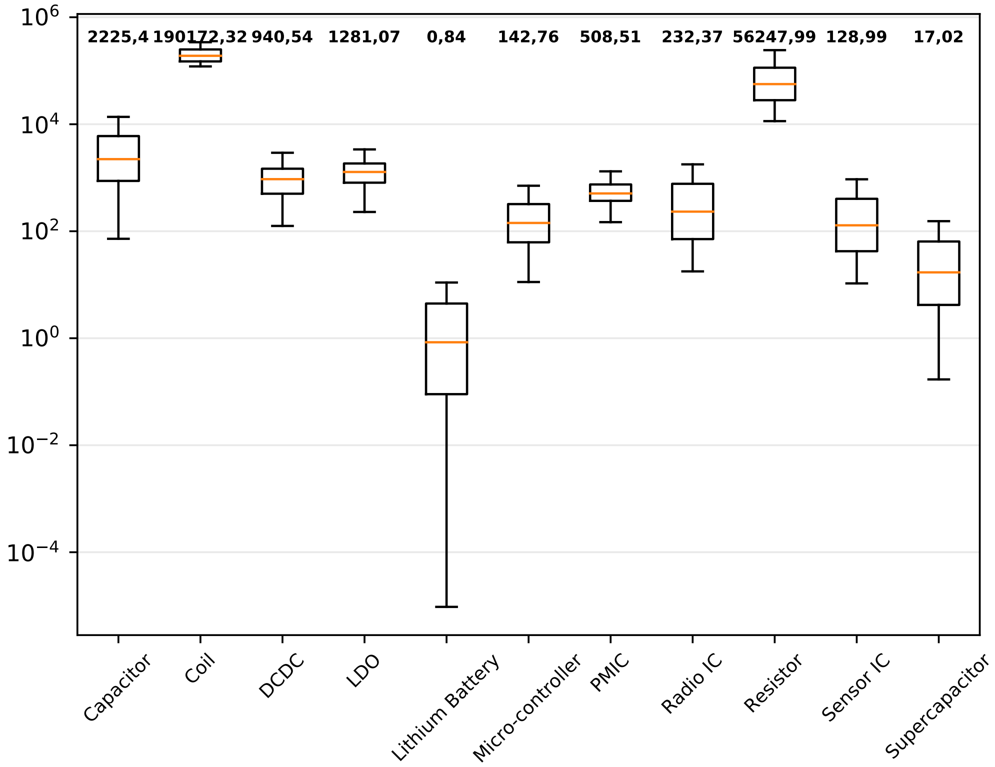


Figure 1.21: MTTF comparison for all components of an energy harvesting device.

1.7 Conclusion

In this chapter, we presented the different components used in the design of an energy harvesting system, and the associated trade-offs. Additionally, we proposed a lifetime assessment of the different energy harvesting system building blocks, based on the MIL-HDBK-217 specifications and state of the art energy storage models. To the best of our knowledge, this is the first lifetime analysis applied to energy harvesting WSN nodes with theoretically infinite lifetimes. Using this analysis, we show that the theoretically infinite WSN lifetime, advertised by the use of a power manager, is in reality limited by the lifetime of the components, and especially of the battery storage. Thus, we show that care should be taken to the energy storage choice in order to properly ensure a long WSN autonomy.

Chapter 2

Multi-Source Energy Harvesting

Energy harvesting has been an active area of interest in recent years, in both academic and industrial worlds, notably to deploy WSN in remote locations [114][115]. While most early energy harvesting architectures focus on the use of a single energy source, recent systems attempt to simultaneously harvest energy from multiple heterogeneous sources. This chapter presents state-of-the-art energy harvesting platforms, and our proposal for a low cost multi-source energy harvesting architecture.

2.1 State of the Art Energy Harvesting Platforms

The performance of these systems can be compared on many different aspects. Common metrics include the harvesting, power conversion and charging efficiencies of the system, but also the cold boot time of the platform, i.e. the time it takes for the platform to boot from fully depleted energy storage elements. In many use-cases, the cost, size and integration of the solution also play a significant role. In this section, we review the existing energy harvesting platforms and their characteristics.

2.1.1 Single Path Architecture

The most straightforward energy harvesting architecture is called the single-path architecture. In this architecture, the energy storage, which can be either a lithium battery or supercapacitor, acts as an energy buffer between the energy harvesters and the power supply of the node. All harvested energy charges the energy storage which powers the node. Fig. 2.1 shows a block diagram of such an architecture. This architecture is simple and straightforward to implement, while limiting its cost. However, this architecture increases the booting time of the platform. Indeed, the power supply of the node requires a minimum input voltage to operate, which will not be reached until the energy storage is sufficiently charged. A bigger energy storage will provide more autonomy but will increase the booting time, while a smaller energy storage will charge faster but decrease autonomy. This drawback can be worked around by charging the energy storage just before deploying the WSN.

During the recent years, multiple energy harvesting platforms which use this architecture have been proposed. Heliomote [11] is an example of such a platform, aiming to bring energy harvesting capabilities to an off-the-shelf sensor node (Mica2 motes). This platform is designed to harvest energy from solar panels. Energy

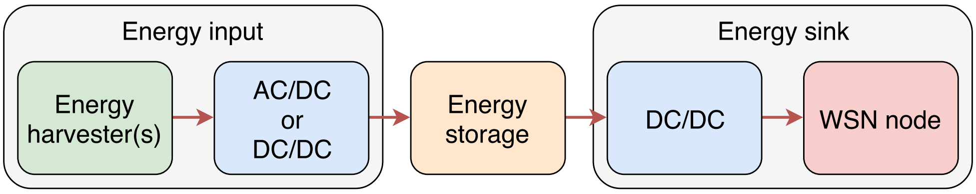


Figure 2.1: Single path architecture for energy harvesting node

is stored in two NiMH batteries, which voltage ranges from 2.2 V to 2.8 V. The batteries are rated for 500 discharge cycles only, which limits the lifetime of the platform. Moreover, this platform does not use a MPPT circuit. Instead, a diode is used to protect the 4.0 V solar panel from reverse current, and to keep the voltage of the solar panel between 2.9 V and 3.5 V, closer to the MPPT. This approach is inexpensive to implement but creates losses due to the voltage drop of the diode.

Zebranet [77] shows an application example of such a system. In this work, a WSN record GPS position data of long-term animal migrations. The nodes are powered by a solar array of 14 solar modules, where each module includes a boost converter and produces up to 7 mA at 5 V. An MPPT circuit based on a hysteresis comparator is implemented in each module. This method is inexpensive but lacks precision compared to state-of-the-art MPPT circuits. The use of piezoelectric harvesters and mechanical conversion techniques are also considered, but are dismissed due to mechanical and packaging considerations. The energy is stored in a 2 mAh lithium-ion battery with a voltage varying from 3.0 V to 4.2 V, which is able to power the node for about 3 days. Although the lithium-ion batteries are rated for 1000 charge cycles, which is higher than NiMH batteries, the lifetime of the system is still limited by its energy storage. Moreover, Zebranet does not provide any power management scheme outside of power consumption reduction, which makes it heavily reliant on a sunny weather.

Everlast [12] increases the system lifetime of the platform by replacing the battery by a supercapacitor. This solar powered platform uses a custom designed voltage regulator to efficiently charge the supercapacitor. Everlast uses energy pulses to charge the energy storage up to 400% faster than direct charging, where a current limited energy source is directly connected to the supercapacitor. Moreover, the power conversion circuit includes a MPPT function, which sweeps the I-V characteristic of the source to get the MPP. This significantly increases the conversion efficiency of the platform. However, the MPP implementation is done in software, and requires multiple components to stay active. Thus, the control circuitry of this platform consumes over 2 mW, which is too high for a variety of energy harvesters, especially in indoor harvesting systems.

In [116], Weddell et al. show the design and implementation of an energy harvesting system specifically designed for indoor solar panels. The system uses a buck-boost topology and integrates a MPPT circuit, which periodically samples the open circuit voltage of the solar panel to calculate its MPP voltage. The circuit reaches over 70% efficiency under SPICE simulations and shows a 30% improvement over diode-only system in practical tests. However, this system requires further adaptations to be usable with low voltage sources such as thermoelectric generators and alternating current sources such as wind turbines and piezoelectric elements. It is also subject

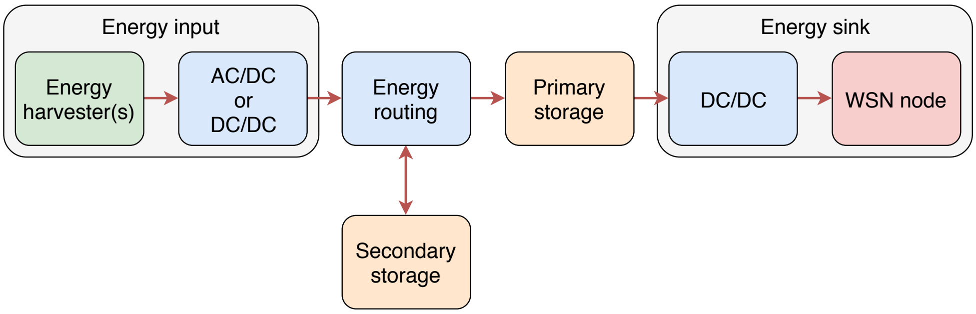


Figure 2.2: Dual path architecture for energy harvesting node

to the limitations of single path energy harvesting systems.

The cold boot problem has been recognized and some single path architecture implementations try to mitigate its effects. In TwinStar [117], Zhu et al. address the instability of the voltage regulator during cold boot. The addition of a Schmitt trigger ensures that sufficient energy has been harvested before starting the regulator. To limit the power consumption overhead of this solution, this boot-up circuit is powered by an independent solar panel which stores its energy in a 47 μF capacitor. However, this second solar panel increases the cost of the solution, and TwinStar does not accelerate the cold boot time. Solar Biscuit [118] defines multiple operating modes, in order to allow the node to boot with lower energy in a degraded mode. This software solution reduces the boot time but still requires significant energy to power the main micro-controller and radio module.

2.1.2 Dual Path Architecture

In order to tackle the limitations of single path architectures, dual path architectures have been developed. In a dual path architecture, two energy storages, called primary and secondary storage (PS and SS), are used instead of only one, as shown in Fig. 2.2. The PS usually has a small energy capacity and is used to power the node, while the SS has a higher capacity and is not directly connected to the node. The harvested energy is used to charge the PS in priority, and the SS is only charged with the surplus energy when the PS is fully charged. On the other hand, the SS is used to recharge the PS when environment energy is insufficient. The use of a small energy storage for the PS enables a fast charging, thus a fast cold boot from empty storages. The use of a bigger energy storage as the SS enables a long operation even when environment energy is lacking, without slowing the booting time. However, this architecture requires circuitry to route energy between the different elements, which adds to the cost and complexity of the energy harvesting solution. Due to this, a large number of current energy harvesting systems still uses a single path architecture.

Prometheus [119] is an example of an early dual path energy harvesting system. This platform uses a Li-Ion battery for long term energy storage and uses a supercapacitor as the PS in order to increase the lifetime of the system. By using it as an SS, the lithium battery is charged and discharged less often, increasing its lifetime. On the other hand, the PS is charged and discharged more often, but this does not affect the lifetime of the system as a supercapacitor is used. As these components have a higher leakage current than lithium-ion batteries, an energy model is proposed to define a trade-off between the supercapacitor characteristics and the power consumption of the node. However, this circuit does not integrate an MPPT circuit, which limits its harvesting efficiency. This platform still suffers from a long boot time, which is due to the high start-up voltage of the regulator.

DuraCap [120] directly addresses the cold booting problem and MPPT for different solar panels. The harvested energy is used to charge a small capacitor as the PS, which can quickly boot the node, while the SS is made of an array of supercapacitors. DuraCap uses a I-V curve based MPPT circuit, similarly to Everlast [12]. However, the control logic of the MPPT is implemented with four hardware analog comparators, instead of software ones. This reduces the power consumption overhead of the MPPT circuit and thus rises its efficiency. The MPPT circuit can be re-calibrated through digital potentiometers by the node when a new solar panel is plugged. EscaCap [13] extends this work by making the supercapacitor array reconfigurable, enabling the node to connect the supercapacitors in series or parallel. This can be used to improve the charging speed of the array, reduce its leakage energy, and use more energy from the supercapacitors when they have a low energy level.

In [121], Le et al. detail the design of a dual path energy harvesting system using off-the-shelf components, in order to extend the existing PowWow [122] platform. In this work, a capacitor of appropriate size is used for both PS and SS and the platform harvests energy from a thermoelectric generator. In [123], this work is extended to use either a solar panel or a thermoelectric generator as its energy harvesting source. Although this makes the platform more flexible, this rules out the use of a simple MPPT circuit, as the possible sources have different characteristics and a different MPP, which limits the harvesting efficiency of the platform. This extended work also describes a method to accurately determine the size of both PS and SS, which takes into account the leakage current of the supercapacitor and the power consumption of the node.

[28] shows a variation of the classical dual path architecture for an environmental measurement system which harvests its energy from a thermoelectric generator. This circuit is based on the off-the-shelf LTC3108 [74] IC, an ultra-low voltage boost converter which integrates power management circuitry which can be used to implement a dual path system. By default, the component first outputs a stable regulated voltage, which can be used with a PS, and charges a supercapacitor used as SS afterwards. The SS is then used to maintain the output voltage if the harvested energy is not sufficient. However, Dias et al. bypass this power management circuit to implement their own, where only the energy management circuit is powered by the PS. The rest of the node is powered by a second regulated output of the LTC3108 when harvested energy is sufficient, or switches to use the supercapacitor when there is not enough energy. Although this system is not a pure dual path architecture, as only a part of the node uses the PS, this hybrid approach reduces the energy management circuit power consumption overhead and increases the autonomy of the system, when no energy is harvested, to 266 h, compared to 136 h when using the LTC3108 dual path energy management circuit. One drawback of this system, however, is that no MPPT is used, which limits the efficiency of the harvesting system.

More recently, [124] shows an example of a state-of-the-art dual path energy harvesting system dedicated to solar energy harvesting. This system aims at minimizing the cold boot time under very low illumination level. It uses standard capacitors as primary energy buffer, backed by a backup supercapacitor as SS, and details the methodology used to properly size both primary and secondary storages. The implementation is based on off-the-shelf components, which makes it accessible. The novelty of this circuit lies in a day/night detector which is used to route energy between the different elements. The full node has a power consumption of less than 25 μW in sleep mode, which makes it last 81h in total darkness. However, this system does not include a MPPT circuit, limiting its efficiency. It also does not use any voltage regulator, which means the power supply of the node will not be stable over time. This can cause problems for some radio modules and components which require a certain voltage to ensure output power.

2.1.3 Multi-Source State of the Art Architectures

While a significant number of energy harvesting systems are dedicated to a single energy transducer, a recent trend is to design multi-source energy harvesting systems. This is especially useful to ensure the service continuity in harvesting environment, for example by combining complementary energy sources such as a wind turbine and a solar panel. Several systems, such as [125], switch from an energy source to another when the former source does not give energy (e.g. a solar panel during night time). In this thesis, the definition of multi-source energy harvesting system is limited to systems which can simultaneously harvest energy from multiple energy sources, thus increasing the total harvested energy. While some works [126] assume that the total harvested energy is simply the sum of energy from the different sources, the different energy sources used in IoT systems exhibit very different characteristics. This makes combining different energy sources non trivial. Multi-source energy harvesting systems can be based either on single-path or dual-path architectures, as they simulate a combined energy harvesting source and do not require completely new energy harvesting architecture. This enables designers to capitalize on previous works more easily.

An early multi-source management circuit is proposed in [127]. This system uses a solar panel and considers its battery as a second energy source. In this work, the power sources are not combined but connected to one or more matching power consumers, which can be the microelectronic components, sensors, wireless transmission module or battery charger of the node. The power sources are connected to power consumer subsystems through a switch array called Power Utility Maximizer (PUMA), that includes a switch unit for each consumer subsystem. The switch array can be dynamically reconfigured with control logic. Although the system manages to balance the power consumption between the battery and solar panel and could theoretically be used for multi-source energy harvesting system, it is complex to implement. Indeed, the complexity of the switch unit increases with the number of both energy sources and consumers. This makes it impractical to add energy sources after design, as it requires a redesign of the switch units. Moreover, the active circuitry of the PUMA switch induces a power consumption overhead, which is not quantified in the original publication.

The simplest way to combine energy sources is to connect them through diodes, as shown in Fig. 2.3. [128] is an implementation of this architecture, where an indoor

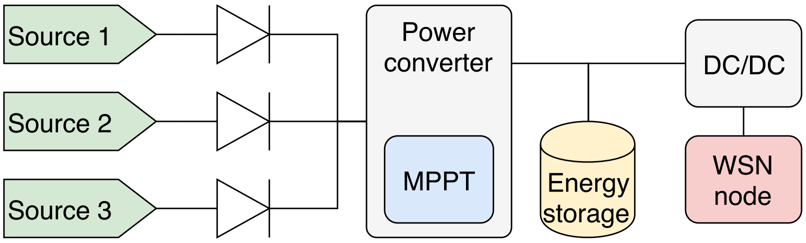


Figure 2.3: Diode ORing architecture, as presented in [128]

solar panel and a thermoelectric generator are connected through Schottky diodes to an energy converter. The use of a single energy converter aims at reducing the cost, form factor and design complexity of the system. The system is based on a single path architecture which uses a 0.1 F supercapacitor as storage, and targets deployments in office and factory settings, where indoor light is intermittent and machines constantly emit heat. This system uses an MPPT circuit with a fixed reference voltage which is cheap and power efficient. However, this implementation lacks flexibility, as using a different solar panel or thermoelectric generator would change the MPP, and would require a redesign of the MPPT adapted to the new MPP. This circuit does not track the MPP of each source individually but uses an MPP of the combined energy sources, which means that each source will not have its best efficiency. Moreover, the calculated efficiency of 90% does not take into account the 135 μW power consumption of MPPT control circuit, which is a significant power overhead. Additionally, the power losses in the diodes is not quantified. Indeed, the 0.2 V voltage drop of the diodes reduces the harvested energy.

Another approach of multi-source energy harvesting based on diodes is presented in [49]. This system uses both light and vibration energy to charge a supercapacitor. The solar panel is associated with an MPPT circuit using the fractional open circuit algorithm, while the piezoelectric generator is associated with a buck-boost regulator which acts as an impedance matching circuit. These circuits are fully implemented using analog components, in order to reduce power consumption compared to digitally controlled implementations. This enables both sources to harvest energy close to their MPP, thus maximizing their energy harvesting efficiency. However, the use of an impedance matching circuit for each source increases the cost of the solution. Moreover, this system suffers from the same drawback as other diode based multi-source energy harvesting system: the diode induces a voltage drop and thus power losses.

Ambimax [31] is a multi-source energy harvesting system which can combine a solar panel, a wind turbine, a thermoelectric generator and a piezoelectric generator. Each energy transducer is connected to a supercapacitor through a boost regulator with a custom MPPT circuit. This circuit is fully implemented with analog components and no digital control, which lowers its current consumption to less than 500 μA. All supercapacitors are connected together through diodes, as shown in Fig. 2.4, to form a Reservoir Capacitor Array (RCA). This RCA is used to power the node and charge a back-up LiPo battery. New energy harvesting sources can be simply added by connecting a new energy harvesting subsystem to the RCA with a diode. In this design, each supercapacitor has to be properly sized to its associated source in order to ensure that its terminal voltage is not always lower than the other

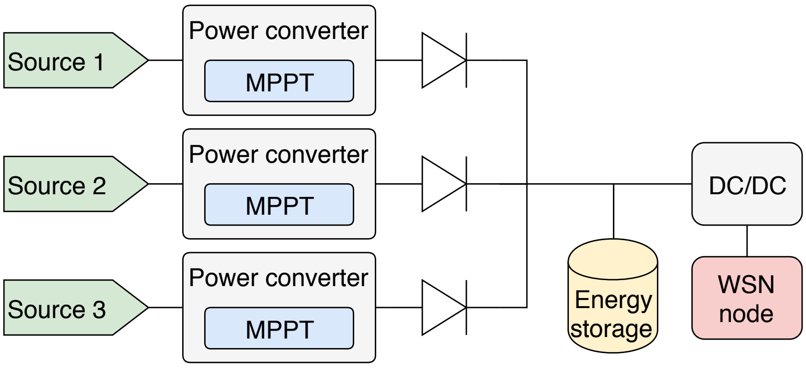


Figure 2.4: Diode ORing with multiple power converters [49][31]

supercapacitors voltage in the RCA, which would prevent it from delivering current. However, the use of diodes to connect the supercapacitor to the RCA creates some power losses, which reduce the efficiency of the multi-source combination circuit. Moreover, supercapacitors are expensive components, and the use of multiple supercapacitors significantly increases the cost of the system, which can reduce its interest for commercial applications.

Inspired by this system, [32] proposes a WSN node for agriculture monitoring, powered by a solar panel, a wind turbine and a small hydrogenerator. Each energy harvesting source is connected to a capacitor which is used as a charge reservoir. The voltage of this capacitor is monitored by a comparator with hysteresis. When this voltage rises above an upper threshold, a boost converter is powered on and discharges the charge reservoir into a NiMH battery. Once the charge reservoir voltage falls under a lower threshold, the boost converter is turned off and the battery is disconnected. These pulses can be counted to estimate the harvested energy. By setting the thresholds of the comparator near the MPP of the energy source, this system constantly operates the energy transducer around this point and thus maximizes harvesting efficiency. Also, charging with this circuit is done with energy pulses which increases the energy transmission. This system does not need diodes because the boost converters isolate the energy source from the battery when they are powered off. Current flowing back from an energy harvester to another is only possible when two or more converters are active at the same time. Thus, the system is designed so that the battery voltage is always lower than the converter output voltage, which ensures the current will always flow towards the battery. The main drawback of the MPPT scheme used, which relies on a hardware implemented comparator threshold and requires a redesign if another energy source is used.

Multiple other systems use a power converter for each energy source, in order to maximize the harvesting efficiency. More recently, CAPNET [129] reuses a similar architecture for an outdoor wireless node powered by light, wind and thermal gradient, while [130] uses this architecture with a solar panel and a wind turbine in a dual-path architecture. [131] proposes a standardized plug and play interface where each energy harvesting module has to regulate its output to a maximum of 4.5 V and prevent back-flow of current with a diode or switch. The goal of this interface is to enable the node to identify its different energy harvesting sources in order to better manage its energy resources and dynamically reconfigure itself when energy sources are added or removed.

In order to reduce the form factor and power consumption overhead of the multi-source energy harvesting system, it is possible to develop a dedicated Application Specific IC (ASIC) instead of a PCB level solution. For example, [132] is an ASIC which can harvest and combine energy from a solar panel, a thermoelectric generator, a piezoelectric generator and an electromagnetic induction harvester operating on the 13.56 MHz band. In this component, called MultiHarvesting Power Chip (MHPC), each energy harvesting source is connected to an appropriate power converter. Each power converter can be connected to its own local storage device or a single common storage device. The power converters implemented are LDOs, which are cheaper to implement and do not require external passive components, but are much less efficient than SMPS converters. Moreover, if reverse current protection is not implemented, current can flow back from the output of the converter to the energy source if the voltage difference between output and input is higher than the LDO transistor parasitic diode threshold. In this design, no mention of reverse current protection is made. Moreover, this IC does not integrate MPPT circuits, which limits the harvesting efficiency of the solution.

[133] is another example of integrated circuit for multi-source energy harvesting from a solar panel, a thermoelectric generator, a piezoelectric generator and a radio-frequency energy harvester over the 935 MHz band. Each energy harvesting source uses its own power conversion chain, which outputs power to its own capacitor. The four capacitors are connected in series in order to add their voltage, and the capacitor stack is connected to a supercapacitor which powers the load. In this work, the solar panel has a primordial importance as it is responsible for the startup of the complete IC. As such, the solar panel energy harvesting chain integrates an MPPT circuit based on the P&O algorithm presented in Section 1.5. However, the other energy harvesting sources do not use any MPPT circuit. Moreover, this IC has only been tested in simulations, and has not been fabricated. Thus, it has not been validated in real world conditions.

An alternative approach, shown in Fig. 2.5 is proposed in [134]. In this work, a multi-input energy harvesting power converter ASIC is designed to combine energy harvested from a solar panel, a thermoelectric generator and piezoelectric generator. The IC integrates a Hill Climbing MPPT tracking circuit for its photovoltaic input and adapts the duty cycle of the regulator to match the impedance of the thermoelectric and piezoelectric generators. This component adapts the inductor sharing design developed in [69] for heterogeneous energy sources. A switch matrix is used to connected one energy source at a time to the converter, which is reconfigured as a boost or buck-boost converter to match the energy transducer. Each energy source is connected to a capacitor, so that the harvested energy is not wasted when it is not connected to the power converter. The IC also implements a dual path architecture which output a regulated voltage to a primary storage and an unregulated power output to a secondary storage supercapacitor. The solution minimizes cost by reducing the number of required external components. However, manufacturing a component is expensive and the application must be produced in volume in order to be economically efficient. Moreover, the switching pattern between sources is hard-coded, which limits the flexibility of the solution when choosing energy sources.

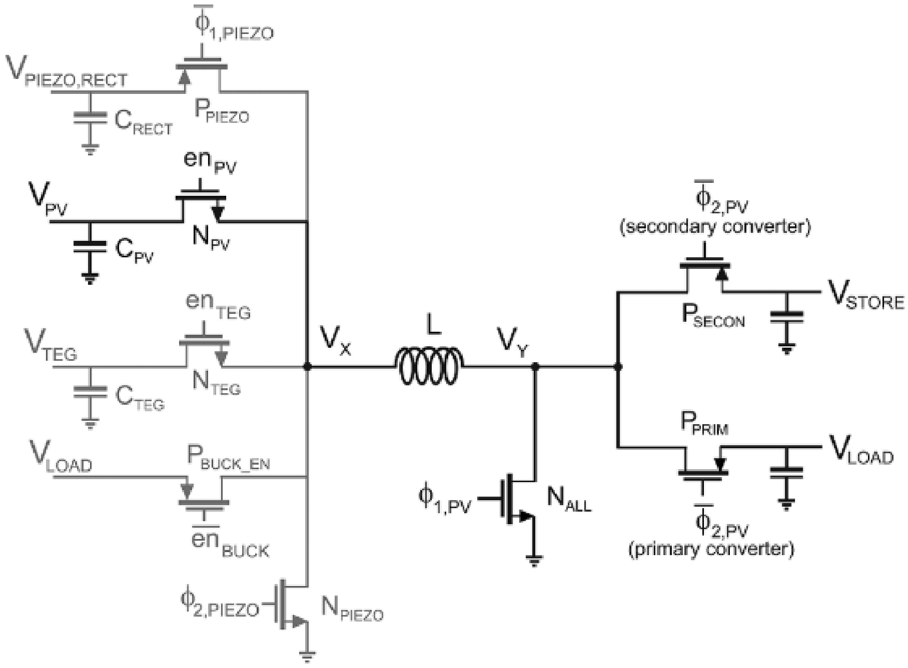


Figure 2.5: Shared inductor architecture presented in [134], configured for photovoltaics

2.2 A New Approach to Multi-Source Energy Harvesting

When designing an industrial solution, there is interest in developing a solution as generic and flexible as possible, in order to reduce Non-Recurring Engineering (NRE) costs as much as possible. However, generic designs often show less performance, and a compromise has to be made between flexibility and performance. In the context of energy harvesting, this translates into having to balance between the energy harvesting and power conversion efficiencies on one side, and the possibility to use any energy source, or even multiple sources on the other side. In this section, we present the design of a multi-source energy harvesting platform aimed for use with an existing commercial IoT node. This platform should be able to be adapted to new designs with minimal modifications while keeping a competitive efficiency. In order to ease its industrialization, the platform should rely on off-the shelf components and have a low cost.

2.2.1 Multi-Source Switching Architecture

A way to reduce the cost of a multi-source energy harvesting solution is to use a single power converter, instead of using one per energy source. Fig. 2.6 shows the block diagram of our proposed solution. In this system, a switch matrix circuit is used to select which energy source is connected to a single PMIC at a time. This component integrates a voltage converter, a MPPT circuit and a battery charging circuit in a single package. To avoid wasting harvested energy when an energy source is not connected to the PMIC, a capacitor is used as an energy buffer. When an

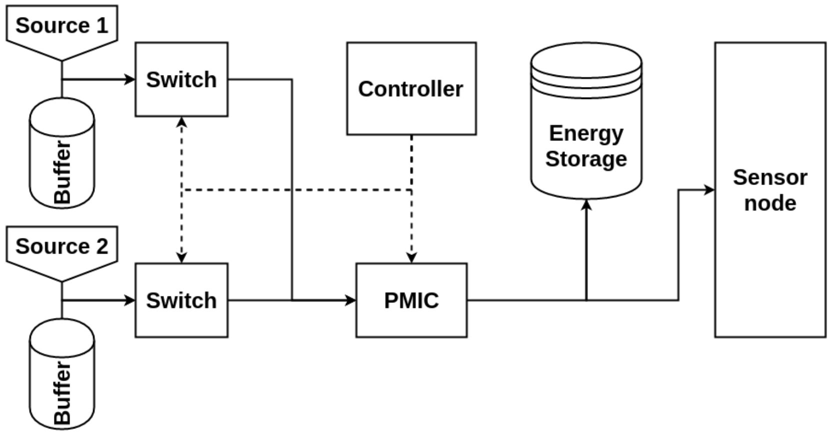


Figure 2.6: Multiple source switching system.

energy buffer is connected to the power converter, the latter draws as much current as it can from the buffer to charge the main energy storage and power the WSN node. When the energy buffer is emptied, its switch is opened and a new energy buffer is selected. Contrary to previous work such as [134], a programmable controller is used to decide which source should be connected to the PMIC at each time. This programmable solution can be used to dynamically adapt the switching policy to the different sources and new environment conditions.

This system uses a single path architecture. The power path contains two SMPS converters, which reduces the overall power conversion efficiency. The first is a buck-boost converter, embedded in the PMIC, used to transform the input voltage and charge the energy storage. The second is a buck converter present on the commercial WSN node platform, and is used to power the node from the battery. This architecture choice reduces the overall cost of the platform and allows to easily use our system with existing platforms, simply by replacing the node energy supply by the PMIC energy storage.

The PMIC integrates a power converter, a MPPT circuit and a battery charger. The same functions could be implemented separately with discrete components. The use of a generic component instead of a custom designed power converter has a negative impact on the conversion efficiency. However, using an off-the-shelf IC has advantages in terms of size and cost. In this work, the PMIC used is the SPV1050 [76] from STmicroelectronics. This PMIC integrates a converter which can be used as a boost or buck-boost, depending on the application. Moreover, the input stage can tolerate voltages from 0.15 V up to 18 V, which covers a large range of energy transducers, and has a low quiescent power consumption P_Q, as shown in Fig. 2.7, e.g. 6 μW or a 5 V input voltage.

Most commercial off-the-shelf energy harvesting PMIC use a Fractional Open Voltage Circuit as a cheap and low power MPPT implementation. However, this approach is incompatible with our switching system. Indeed this MPPT method relies on periodically opening the energy harvesting circuit to measure the open circuit voltage V_{OC} of the energy transducer. When the transducer is connected to an energy buffer, the impedance of the buffer prevents the energy transducer from reaching its V_{OC} during the sampling time, which gives an incorrect value for the

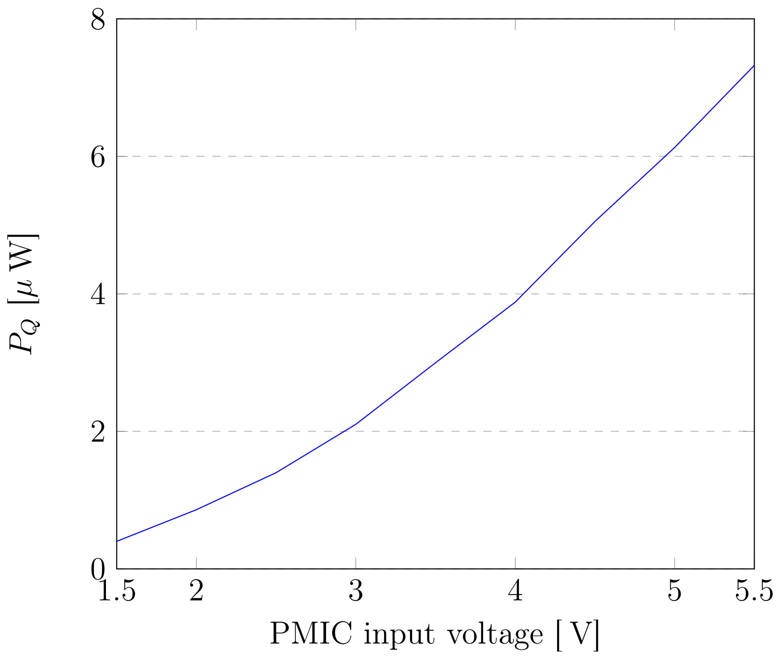


Figure 2.7: Quiescent consumption of the PMIC.

MPPT algorithm to work on. Moreover, the measured MPP corresponds to a single source and is not adapted to a second energy source, and will therefore reduce the harvested energy. A more advanced MPPT method, such as hill-climbing or P&O algorithm is required. However, the implementation of such algorithm requires additional hardware and software, which increases the size, cost and power consumption overhead of the platform. Thus we have chosen to not use any MPPT methods, and rely on switching policies to maximize the harvested energy.

The MPPT circuit can still be used to control the PMIC. Indeed, the MPP voltage V_{MPP} of the SPV1050 can be set externally to control the integrated regulator. If this external pin is set to an arbitrary reference voltage V_{MPP}^{set}, the converter integrated in the PMIC will adapt its switching frequency to keep the input voltage close to V_{MPP}^{set}. This can be used by the programmable controller to more efficiently manage the discharge threshold of the energy buffers.

2.2.2 Multi-Source Switching Implementation

The developed architecture includes multiple active elements, such as the switches and the controller, for which the power consumption overhead must be minimized as much as possible. The size of the energy buffers can also impact the efficiency of the system. The following subsection details some design choices regarding the sizing, design and impact on the system of the different elements.

2.2.2.1 Energy Buffer Sizing

The input energy buffers are implemented using capacitors. An accurate sizing of these capacitors is required, as an oversized capacitor would increase its leakage current [117], as well as the cost and size of the solution. On the other hand, an undersized capacitor would be too quickly charged by its energy harvester, leading to energy waste once it is full of energy. In the proposed architecture, the size of the input energy buffers depends on both voltage and current provided by the harvesting device, but it also depends on the switching process. More specifically, the number of harvesting devices, the switching period D_{SW} between two switching decisions and the decision algorithm have an impact on the duty cycle of the switch associated to the harvester, and thus on the time it is charging or discharging. The capacitance of the energy buffer can be derived from the energy source as follows.

Let C_{BUF} be the buffer capacitance and V_{BUF} its voltage, the current I_{BUF} is given by:

$$I_{BUF} = C_{BUF} \frac{d\,V_{BUF}}{dt}. \tag{2.1}$$

The use of a solar panel is considered, which can provide up to V_{OC} Volts, I_{MAX} mA and has its MPP at $V_{MPP} = K_{focv} * V_{OC}$. At initial conditions, energy buffer is supposed to be empty with a voltage down to V_{MPP}^{set}. During charging state, once the voltage V_{BUF} reaches V_{MPP}, the power provided by the panel starts to decline. The energy buffer is thus considered to be "charged" when its voltage rises above the V_{MPP} voltage of its harvester. The time D_{CHRG} required to charge the capacitor up to V_{MPP} is expressed by:

$$D_{CHRG} = C_{BUF} \frac{K_{focv} * V_{OC} - V_{MPP}^{set}}{I_{BUF}}. \tag{2.2}$$

As the current I_{BUF} depends on its voltage, the worst case is considered with $I_{BUF} = I_{MAX}$. Oversizing the current at I_{MAX} leads to a shorter charging time, and thus a bigger capacitance.

As seen earlier, the capacitance also depends on the switching process between multiple harvesters. Considering a decision algorithm with a periodic switch between the N sources, one buffer is connected to PMIC during D_{SW}, while being in charging state during $(N-1) \times D_{SW}$. Therefore, to avoid overcharging the energy buffers in such a scheme, D_{CHRG} should be greater than or equal to $(N-1) \times D_{SW}$. With this constraint, the minimal energy buffer capacitance C_{BUF} can be computed by:

$$C_{BUF} \geq \frac{D_{SW} \times (N-1) \times I_{MAX}}{K_{focv} \times V_{OC} - V_{MPP}^{set}}. \tag{2.3}$$

Finally, a safety margin is used to prevent capacitance variations, which can occur according to ambient temperature, input voltage or chosen technology, and which can change over time. This margin can be analytically derived from the capacitor specifications.

This expression has been validated through electrical simulations on LTspice [135], using the solar panel model shown in Fig. 1.1 as the energy source. For different capacitances and various illumination conditions, the charging time D_{CHRG}^{sim} from $V_{MPP}^{set} = 1.8V$ to V_{MPP} is simulated, and compared with the theoretical time D_{CHRG}^{calc} calculated using (2.2). Simulation results, shown in Table 2.1, show that our model deviates from simulation when illuminance is low. This difference leads to oversizing the energy buffers, slighly increasing the circuit cost and leakage currents. However, it enables a simplified and sped-up capacitor sizing for each system configuration.

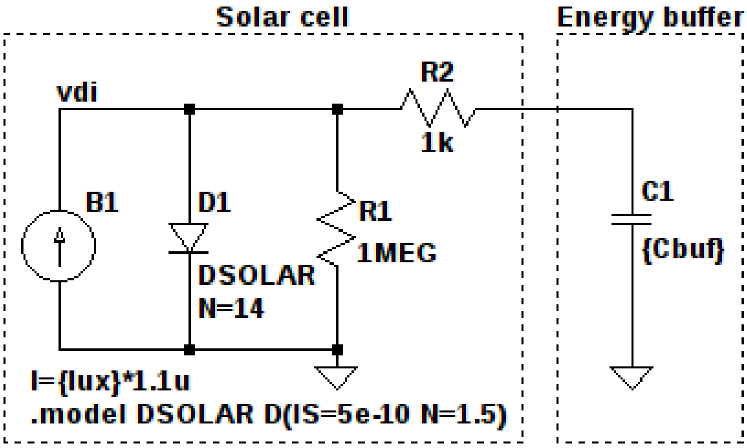


Figure 2.8: Simulation circuit for capacitor sizing, using a solar cell model.

Ill (lux)	C_{BUF} (μF)	D^{sim}_{CHRG} (s)	D^{calc}_{CHRG} (s)	ΔD_{CHRG} (%)
100	10	0.262	0.331	26.3
100	100	2.261	3.304	26.0
100	1000	26.214	32.996	25.9
200	10	0.143	0.174	21.6
200	100	1.431	1.755	22.7
200	1000	14.306	17.529	22.5
500	10	0.064	0.074	16.4
500	100	0.636	0.739	16.3
500	1000	6.357	7.374	16.0
1000	10	0.034	0.036	5.3
1000	100	0.342	0.361	5.6
1000	1000	3.418	3.641	6.5

Table 2.1: Comparison between D^{calc}_{CHRG} and D^{sim}_{CHRG}.

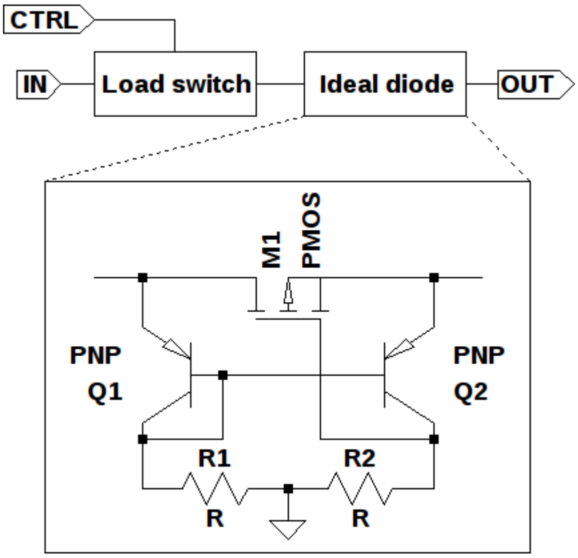


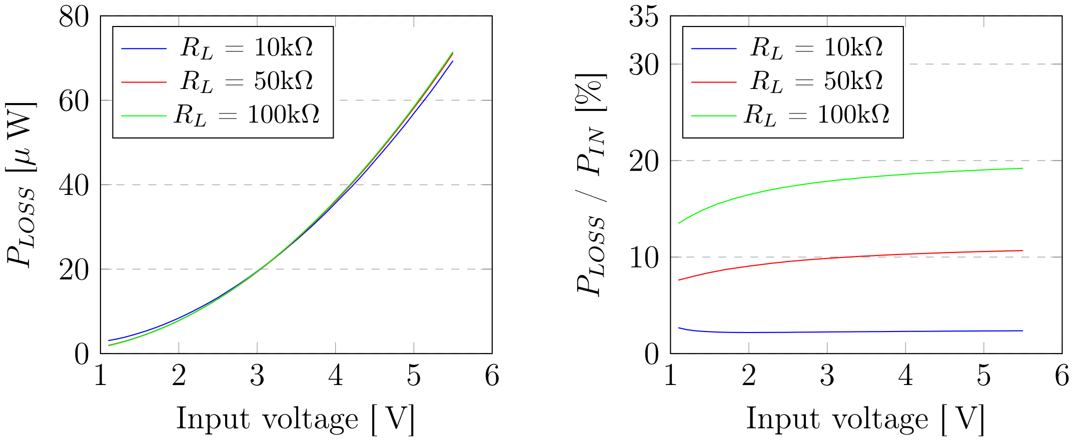
Figure 2.9: Load switch block diagram and schematic of the ideal diode circuit.

2.2.2.2 Switch Circuit Design

Switching between energy sources is a well-known issue. Power ORing [136] has been used for a long time in applications where power failures are unacceptable [137], such as datacenters. However, these systems are designed for power consumption of several kW, whereas our application uses only at most a few mW. Moreover, we want the harvested energy to flow from the energy buffers to the PMIC, and not to flow from an energy buffer to another. Therefore, the switches must have reverse current blocking capabilities.

In our system, shown in Fig. 2.9, each load switch is implemented with a SiP32431 [138]. Due to its input voltage limitations, this circuit can only be used for energy harvesting sources providing voltage between 1.1 V and 5.5 V. For different voltages, another load switch should be used. The load switch is composed of a PMOS transistor as the switching element and a NMOS transistor to control the gate of the former transistor. Like all MOS transistors, the PMOS structure creates a reverse body diode which can let current flow from the drain of the transistor to its gate, thus from the output of the load switch to its input. Some off-the shelf load switch components offer reverse blocking capabilities, but this feature is usually limited to an output voltage lower than the input voltage. This is not always the case in our system, if one energy source delivers a voltage sensibly higher than another, for example if both a solar panel and a thermoelectric generator are used.

Thus, the load switch is coupled with an ideal diode circuit to avoid reverse current flow. The full switch block diagram, which includes the ideal diode schematic, is shown in Fig. 2.9. The ideal diode circuit works as follows. When the output voltage V_{OUT} is lower than the input voltage V_{IN}, Q2 PNP transistor will be in blocking state and the Field-Effect Transistor (FET) gate voltage will be 0 V, making it passing. When V_{OUT} is higher than V_{IN}, Q2 is saturated, the gate voltage is close to V_{OUT} and the FET is blocked. The use of two resistors to polarize the PNP transistors creates an additional power consumption. This consumption can be reduced by increasing the resistor values, however the transient response of the



(a) P_{LOSS} over V_{IN} (b) P_{LOSS}/P_{IN} over V_{IN}

Figure 2.10: Power consumption overhead of a passing switch unit

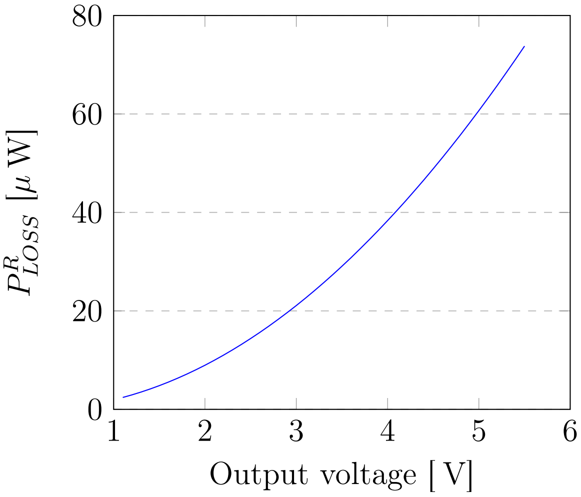


Figure 2.11: P^R_{LOSS} of a switch over V_{OUT}

circuit will also increase.

The power consumption overhead has been measured and is shown in Fig. 2.10 when the switch is closed and current is flowing from the energy buffer to the load. For this measurement, the load is a resistor R_L, which value varies between 10 kΩ, 50 kΩ and 100 kΩ to test different load currents. The power losses P_{LOSS} are shown as absolute values in Fig. 2.10a and relative to the input power P_{IN} in Fig. 2.10b. These measurements show that the power consumption of the switch cell is not negligible. However, these losses are constant and independent from the flowing current. This is because the losses come from the current in the resistors which polarize the Q1 and Q2 transistors. This has the advantage of providing a good efficiency when the current is sufficient.

However, when V_{OUT} is higher than V_{IN}, Q2 is saturated and current flows in R2. This means that the switch cell creates some leakage power losses P^R_{LOSS} on its output when it is blocking. These power losses have been measured and are shown in Fig. 2.11. These power losses are equivalent to those in passing mode, since the same resistors and transistors are used on both side on the circuit. The main drawback of these losses is that they draw current from the active energy harvesting source. It also means that care should be taken when adding too many harvesting sources, since the reverse power losses of each switch will be added to the other. In the case of a system with N energy sources, the reverse power losses are equal to $(N-1) \times P^{R}_{LOSS}(V_{OUT})$, where P^{R}_{LOSS} is expressed as a function of the circuit output voltage V_{OUT}. To mitigate these losses, we can set the V^{set}_{MPP} of the PMIC at a low voltage. The PMIC will draw most current until V_{OUT} is close to V^{set}_{MPP}, and the low V_{OUT} voltage will reduce the reverse power losses.

2.2.2.3 Switch Manager Implementation

In the system, the controller decides which switch should be closed at a time. Multiple decision algorithms can be implemented depending on the information that the controller could monitor in the system. For example, if the energy levels are known, the controller can allocate a time slot per source that is proportional to its maximum output power.

The only requirement for the controller is that the control signals must be compatible with the selected switches. The controller is an always-on component, thus its power consumption must be reduced to a bare minimum. Although dedicated analogue or digital circuits could lower this power consumption, the use of a programmable component allows a good trade-off between power consumption and flexibility.

Programmable digital logic (i.e. Field Programmable Gate Arrays (FPGAs) and Complex Logic Programmable Devices (CPLD)) are usually considered as a good compromise between a specialized and efficient circuit and a more general-purpose implementation. However, their use is not suitable for WSN nodes where an ultra low power consumption is required. Most FPGAs are designed for higher end system and their static power consumption, when inactive, range from hundreds of μW to tens of mW. The few references that target battery devices and have a lower static power consumption [139][140] often require a dedicated low voltage power rail, which increases the implementation cost. Moreover, designing with such components requires the use of specific languages such as Verilog or VHDL, for which it is difficult to find experimented developers.

Though generally considered a more general-purpose and less efficient alternative, micro-controllers can be efficiently used in ultra low power designs. Indeed, recent micro-controllers usually offer multiple power modes. By keeping the micro-controller in a low power mode and only waking it up when needed, the average power used by the micro-controller can be kept close to its sleep mode power consumption, i.e. few μW. Since the micro-controller functionality is implemented using software, its features can be easily modified, simply by loading a new firmware. Moreover, contrary to pure digital components like FPGAs and CPLDs, modern micro-controllers often include ADCs to interface with the analog world, while some references also include programmable analog blocks [141]. In our system, it allows a switching policy to get more information on the SOC of the energy buffers without adding external components.

Due to these considerations, we chose to use a MSP430FR5969 [142] as the switch-

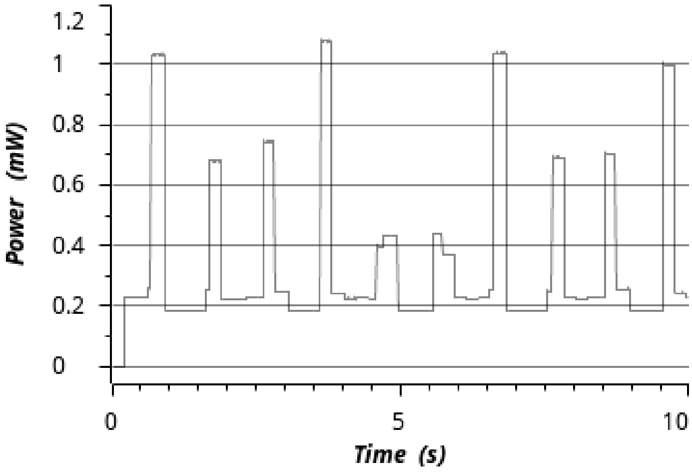
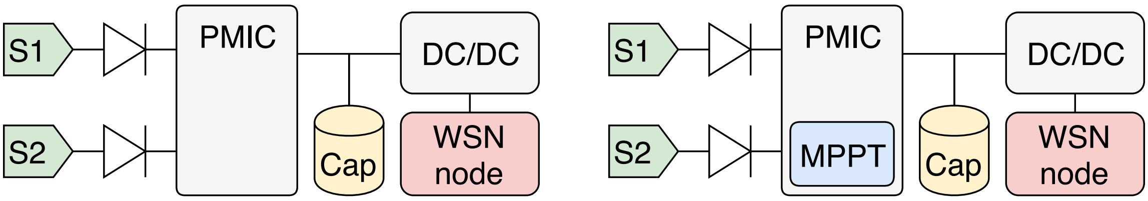


Figure 2.12: Power consumption of the controller.

ing controller. Its flexible clocking trees enable the use of many peripherals while being in low power mode. In our implementation, to avoid adding an external Digital-to-Analog Converter (DAC), an internal timer is used as a Pulse Width Modulation signal, averaged through a low-path RC filter to provide the PMIC its V_{MPP}^{set} voltage. V_{MPP}^{set} can be set down to 1.1 V at minimum, to keep the energy sources voltages in the SiP32421 voltage range. A TPS60210 [143] charge pump with ultra-low quiescent current is used to power the MSP430 with a constant 3.3 V voltage, attenuating the battery voltage variations.

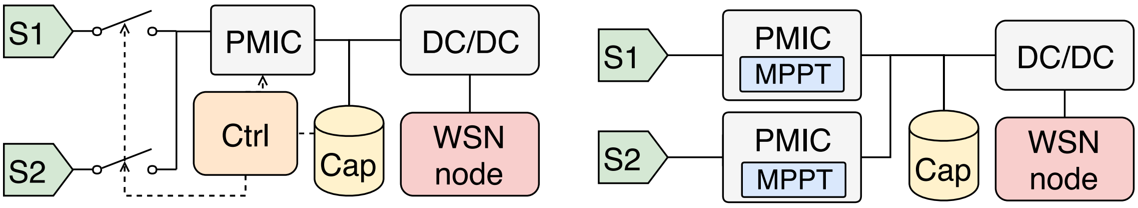
We use the MSP-EXP430FR5969 development platform, which provides an embedded energy measurement tool, enabling real-time measurement of the controller power consumption. In order to take into account the power drawn by the controller charge pump, we powered our development board from the charge pump, and powered this charge pump from the integrated debugger 3.6 V power supply. The charge pump is set in low-power mode, so that its quiescent current consumption is minimized. The SOC of the main energy storage is measured with an external ADC with high input impedance through a resistor divider.

To get an estimation of the controller power consumption, a simple decision algorithm is implemented, which alternatively closes the switches each $D_{SW} = 1\ s$. It also implements a simple power management algorithm, which monitors the main energy storage SOC. Fig. 2.12 illustrates the power consumption of the controller on a 10 second period. To get the average power consumption of the full circuit, a 5 minute measurement has been run. The mean power consumption was 334 μW for a current of 92 μA. The power losses induced by the controller are relatively high, and mainly come from the ADC measurements required by the power manager. If the switch system is implemented without the power manager, the power consumption of the controller decreases at 185 μW for a current of 51 μA.



(a) Diode ORing implementation without MPPT

(b) Diode ORing implementation with MPPT



(c) Switching circuit implementation

(d) Multi-converters implementation

Figure 2.13: Evaluated multi-source energy harvesting circuit architectures.

2.3 Multi-Source Energy Harvesting Evaluation

2.3.1 Experimental Set-up and Measurements

In order to compare our proposed architecture to existing solutions, we implemented the different existing architectures for laboratory experiments, using breadboards and evaluation boards. In order to minimize the differences due to implementation choices, the same components and circuits were reused as much as possible.

The realized prototype is functional and is able to self-start. Functionality of the controller has also been validated using a simple switch algorithm, which alternatively closes the switches each $D_{SW} = 1\ s$. Our system is used to power a wireless sensor developed by Wi6labs [144]. A simple power manager is implemented on the MSP430, which measures the main energy storage SOC and commands the sensor node to transmit a packet if the storage is charged. Therefore, the more energy is harvested, the lower the period between two transmissions D_{TX} will be. Voltage generators are used to emulate sources. The first source is set at 4.2 V, while the second is set between 1.5 V, 3.1 V or 3.7 V. Both sources are limited to 1 mA in order to simulate low power sources. However, these sources are still oversized compared to real energy sources. As the power provided by real sources would be lower, the power consumption overhead of the circuit would have a higher impact on the harvested energy.

Even if these energy sources are oversized, compared to low power provided by real energy sources, the use of deterministic sources allows an accurate characterization of the proposed system. Different measurements can be performed in a consistent set-up, without being affected by the environment. The input energy buffers are arbitrarily chosen and have respectively a 4700 μF and 1000 μF capacitance, while the energy storage is a capacitor array with a 34.7 mF total capacitance. The following cases, illustrated in Fig. 2.13a to Fig. 2.13d are evaluated:

- $S_{1/2}$: Source 1 or 2 only, with MPPT

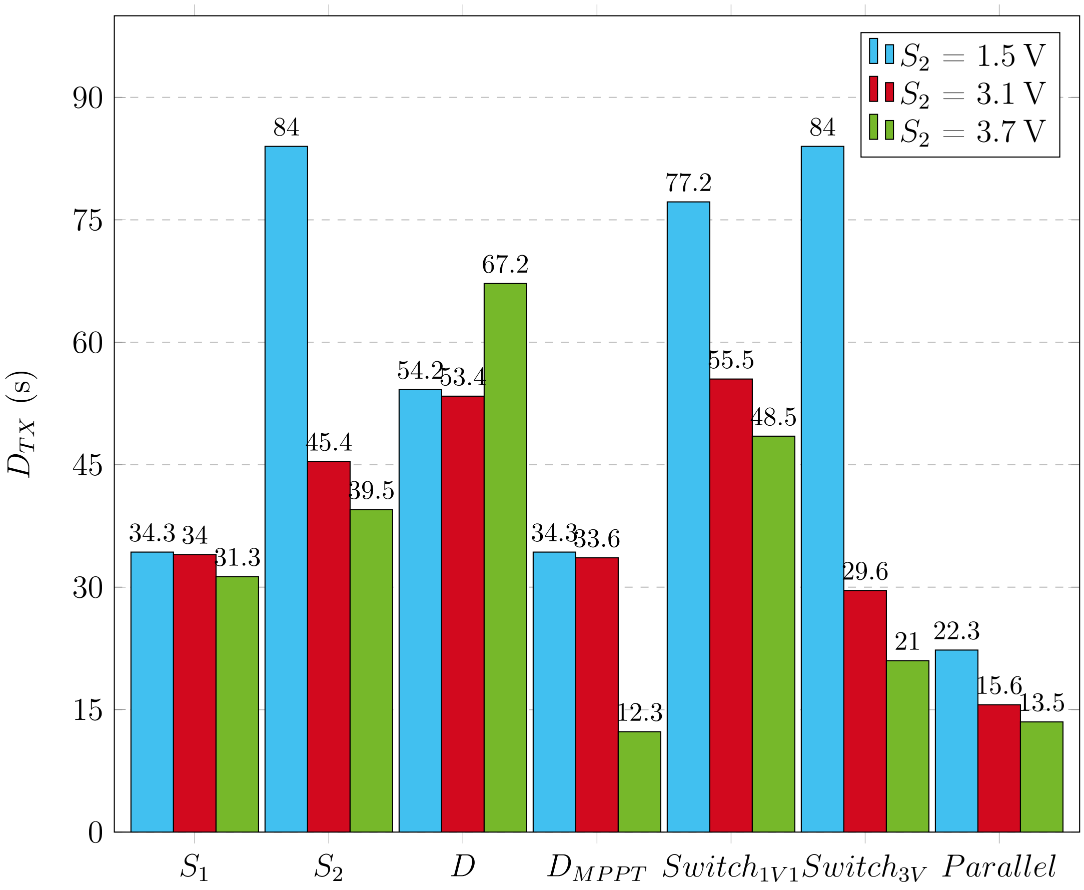


Figure 2.14: Period in s between two LoRa TX depending on case and S2 voltage.

- D: Source 1 and 2 connected to a single PMIC through ideal diode circuit, without MPPT (Fig. 2.13a)
- D_{MPPT}: Source 1 and 2 connected to a single PMIC through ideal diode circuit, with MPPT (Fig. 2.13b)
- $Switch_{1\,V1/3V}$: Periodic switch ($D_{SW} = 1s$) with PMIC V_{MPP}^{set} set to 1.1 V/3.0 V (Fig. 2.13c)
- $Parallel$: Parallel architecture - two PMIC directly connected to the battery, with MPPT (Fig. 2.13d)

For all situations, the period D_{TX} between two consecutive LoRa packet transmissions is measured ten times. The average value $\overline{D_{TX}}$ is then computed. These results are shown in Fig. 2.14. When more energy is harvested, the energy storage is charged faster, and the period between two messages is reduced. On the other hand, a less efficient energy harvesting system will induce a higher delay between messages.

2.3.2 Discussion

The situation D performs significantly worse than single-source situations, while D_{MPPT} performs better. This is due to the lack of any MPPT in situation D, and

Component		Switching Arch.		Parallel Arch.	
Name	Unit Price	Count	Price	Count	Cost
SPV1050	\$2.21	1	\$2.21	N	$\$N \times 2.21$
MSP430	\$0.70	1	\$0.70	0	\$0
TPS60210	\$1.47	1	\$1.47	0	\$0
SiP32431	\$0.28	N	$\$N \times 0.28$	0	\$0
Ideal diode	\$0.30	N	$\$N \times 0.30$	0	\$0
Total		1	$\$(4.38 + N \times 0.58)$	1	$\$N \times 2.21$

Table 2.2: Cost breakdown of the solution.

shows the impact of MPPT. However, D_{MPPT} is a naive implementation, and is not efficient if the two input voltages are too different. When the MPPT circuit measures V_{OC}, it measures the highest voltage in all sources, and sets its V_{MPP} accordingly. Thus, a source with a lower voltage operates far from its MPP, or does not provide power even if its voltage is smaller than the measured V_{MPP}.

Our solution is hindered by the lack of MPPT. By setting V_{MPP}^{set} too low, as in situation $Switch_{1V1}$, the sources operate far from their optimal power point, and provide less energy. Alternatively, rising V_{MPP}^{set} too high may render some sources useless. Indeed, when $S1$ is set at 4.2 V, rising V_{MPP}^{set} to 3 V brings the converter operating point close to the source MPP. But, if the second source voltage is smaller than V_{MPP}^{set}, no current will be drawn, and the source will not be used at all. However, if V_{MPP}^{set} is close to both sources MPP, such as situation $Switch_{3V}$ with $S2$ set a 3.7 V, our solution performs correctly. This demonstrates the potential of our solution, when the sources operate close to their MPP. The *Parallel* architecture does not suffer from this limitation, as a different MPPT circuit is used for each energy source.

The energy buffer size has also an impact on the system: an oversized buffer takes more time to charge, and the energy source takes a longer time to reach its V_{MPP}. In order to maximize harvesting efficiency, decision algorithms should be designed so that the sources operate near their MPP. Possible solutions include setting V_{MPP}^{set} from a DAC and adapt its value according to the selected source, or adding a full-featured MPPT circuit between a source and its energy buffer.

Cost was a primary concern when designing our system. Table 2.2 shows the cost associated with a state-of-the-art parallel architecture and our switching architecture. Only active components are taken in account. Prices are obtained from standard distributors for 100 pieces of each component. Ideal diode cost is estimated. N designates the number of energy sources in the system. Our system becomes cost effective for $N \geq 3$, which makes it interesting for systems using a large number of energy sources.

2.4 Industrial implementation

The previous work was used as a basis for the design of Wi6labs energy harvesting platform. In an industrial energy harvesting WSN node, the system is usually powered by a single energy harvesting source. Although we can easily identify use-cases

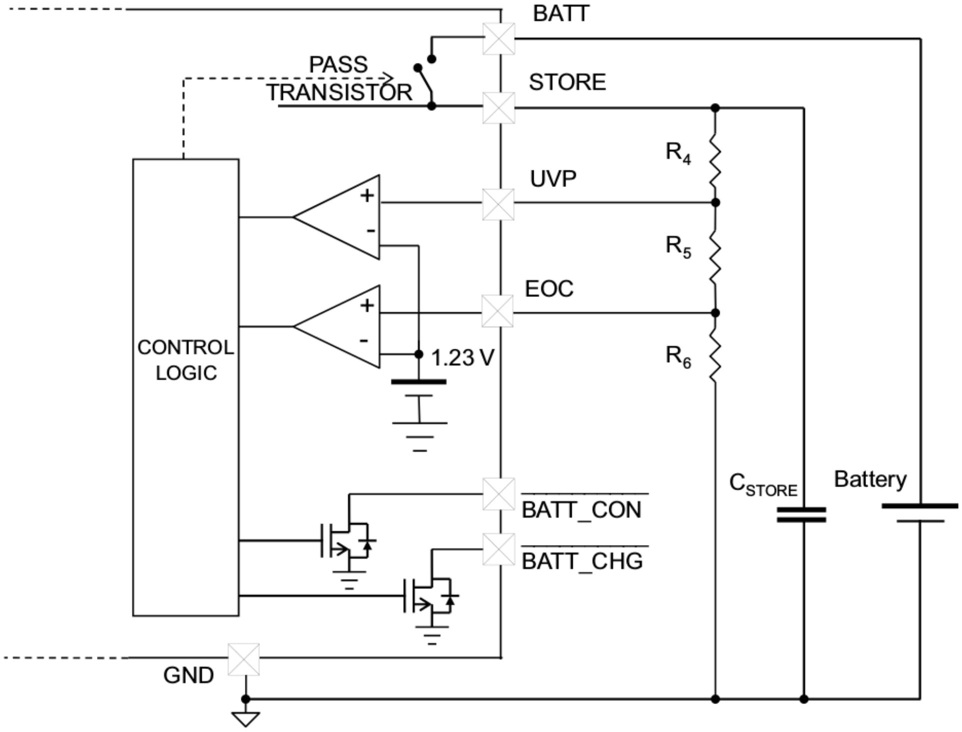


Figure 2.15: SPV1050 battery management circuit - from SPV1050 datasheet [76]

for combining two energy sources, the number of use-cases for 3 energy sources or more is limited. We previously highlighted that a parallel architecture implementation based on the SPV1050 is more cost efficient for a low number of energy sources, and has a better energy harvesting efficiency, especially for heterogeneous sources. Thus, we designed the Wi6labs platform based on this architecture, instead of our proposed source switching architecture.

Fig. 2.15 shows the battery management circuit of the SPV1050 PMIC. It implements a dual path architecture where it first charges a capacitor C_{STORE}. When its voltage V_{STORE} reaches an end-of-charge voltage V_{EOC}, detected with a comparator, a pass transistor is closed to discharge C_{STORE} in the system battery. If the battery is full, i.e. the pass transistor is closed and $V_{STORE} > V_{EOC} - EOC_{HYS}$ where EOC_{HYS} is the comparator hysteresis, the SMPS converter is stopped to avoid overcharging the battery. The pass transistor is opened to avoid discharging the battery when V_{STORE} falls under an under-voltage threshold V_{UVP}. This charging system can be used with both supercapacitors and lithium batteries, provided that V_{EOC} is set between 2.6 and 5.3 V and V_{UVP} is set between 2.2 and 3.6 V. If an energy storage requires end-of-charge and under-voltage protection thresholds outside of these limits, active circuitry would be required to manage the charge limits outside the PMIC.

This system can also be used to implement an energy harvesting system similar to the one proposed in [32]. Our implementation is represented in Fig. 2.16 for tree energy sources. The three SPV1050 use a 47 μF capacitor as C_{STORE}. Here, this capacitor is not used to store energy but to stabilize the output of the SMPS converter. All SPV1050 are connected to a single energy storage. A supercapacitor or lithium battery can be selected, by modifying 3 resistors to set the appropriate

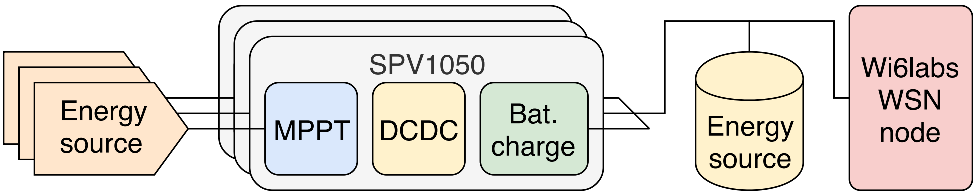


Figure 2.16: Architecture of the Wi6labs multi-source energy harvesting board.

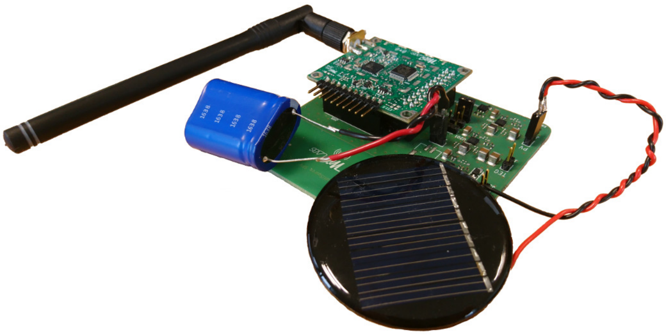


Figure 2.17: Wi6labs multi-source energy harvesting board.

V_{UVP} and V_{EOC} thresholds. When the energy storage voltage V_{BAT} is under V_{UVP} and the pass transistor closes ($V_{STORE} \geq V_{EOC}$), C_{STORE} quickly discharges itself in the energy storage, until V_{STORE} goes lower than V_{UVP}. Thus, the energy storage is charged with pulses until $V_{BAT} > V_{UVP}$, after which the energy storage is constantly powered by the SMPS regulator. However, the converter isolates the energy source from the energy storage, preventing current from feeding back to the source.

Our implementation offers a multi-source energy harvesting system with very low power consumption overhead, as the only active components are the SPV1050. The platform approach makes it possible to reuse the circuit for different energy sources and storage, by modifying solder straps and setting resistors. However, the circuit still has to be adapted, and energy sources and storages can not be interchanged in a plug-and-play manner. As the board is used as a basis to develop custom solutions for customer, this is not blocking in our use-case. The demonstration board, shown if Fig. 2.17, implements a simple single path architecture, which directly powers the WSN node. However, additional features, such as a dual-path architecture or starter circuit could easily be implemented without modifying the multi-source energy harvesting scheme.

2.5 Conclusion

In this chapter, we presented the different architectures which can be used to implement energy harvesting functionality. A new multi-source switching system has been proposed, implemented and evaluated against state-of-the-art architectures. This system is efficient for sources which have similar voltage, and is cost-efficient if three sources or more are used. However, this proposed system does not match Wi6labs requirements. Thus, we implemented a circuit inspired by previous work. This implementation shows a better energy harvesting efficiency and lower cost if less than three energy sources are used. Moreover, this system is designed as a platform, using only off-the-shelf components, and can be quickly adapted to fit a new product design. This platform has been fabricated on a PCB and is now used as the main energy harvesting development platform at Wi6labs.

Part II

Power Management for Long Range IoT Nodes

Chapter 3

Power Management for Energy Harvesting IoT nodes

In modern embedded systems, the role of the firmware includes hardware management tasks. In particular, the embedded software has to properly control the different components in order to manage the power consumed by the device. In energy harvesting systems, this role is particularly important as it enables the node to increase its battery life and avoid battery depletion. In this chapter, we present the different techniques used in modern embedded systems to reduce the power consumption and properly manage the harvested energy.

3.1 Reducing Power Consumption in IoT Nodes

3.1.1 Power Consumption in IoT Nodes

As the power dissipated in passive components is negligible, the power consumption of an IoT essentially comes from its active components. The large majority of modern components relies on Complementary Metal-Oxyde-Silicium (CMOS) logic, which enables a lower power consumption. The power consumption of CMOS components is the sum of a static power consumption P_S, a short-circuit power dissipation P_{SC} [145] and a dynamic power consumption P_{dyn}. The static power consumption is fixed and independent of the component workload. This consumption is due to the fabrication process of the component, and notably the small leakages of each transistor which constitutes the IC. Thus, the reduction of P_S is determined during the conception of the component, when its architecture and fabrication process is determined. The short-circuit power dissipation P_{SC}, illustrated in Fig. 3.1, is created during a gate transition, when a direct path is created between the power supply and the ground by a pair or PMOS and NMOS transistor. This power dissipation can be reduced by ensuring that all switching signals have sharp rising and falling edges, reducing the duration of this short-circuit. The dynamic power consumption P_{dyn} varies with the workload of the component. For each CMOS gate in the IC, the dynamic power consumption P_{dyn} can be calculated as (3.1), where C_L is the capacitance of the gate load, f_ν is the switching frequency of the gate and V_G is the voltage of the gate. As most gates are not always switching, the expression can be written as (3.2) where α is a coefficient between 0 and 1 which represents the duty cycle of the gate. In an

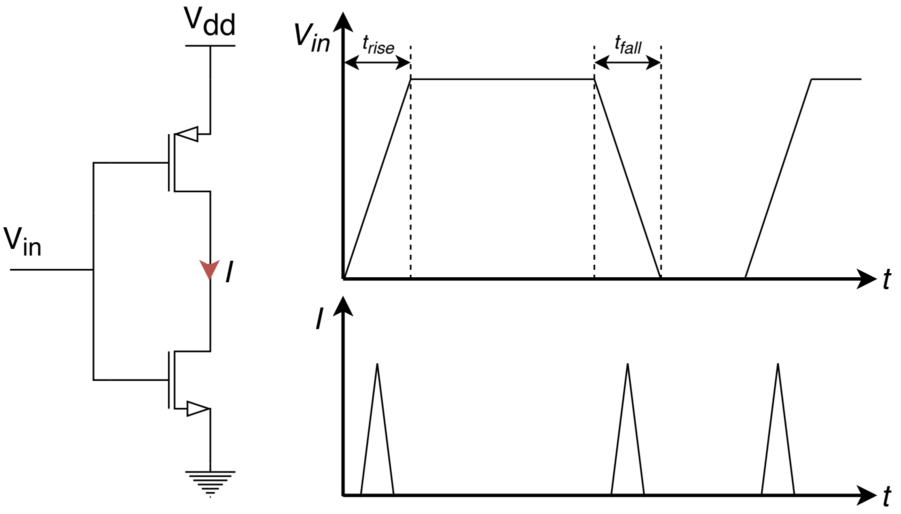


Figure 3.1: Short circuit power consumption in a CMOS inverter.

IC, C_L is determined by the architecture and fabrication process of the component, and thus cannot be reduced once the component is fabricated.

$$P_{dyn} = C_L \times f_\nu \times V_G^2 \tag{3.1}$$

$$P_{dyn} = \alpha \times C_L \times f_\nu \times V_G^2 \tag{3.2}$$

In usual server and workstation computing systems, the majority of the power consumption comes from the CPU. In IoT nodes, the microprocessor core is integrated in a micro-controller, which also integrates RAM and ROM memories and an array of analog and digital peripherals to interface the CPU with external components. The microprocessors used in WSN nodes are small scale device, which do not integrate a Memory Management Unit and can not run usual Operating Systems (OSs) such as Linux [146] or BSD derivatives. Instead, the memory is directly accessed. Although the application can be run bare-metal, without any OS, a small OS such as Riot-OS [147] or Contiki [148], or a Real-Time Operating System (RTOS) such as FreeRTOS [149], is often used to ease the development process. As IoT use-cases become more complex, the use of more complex and featured OS such as NuttX [150] or uClinux [151] is also considered. A majority of micro-controllers embed a microprocessor from the ARM Cortex-M family, but other possibilities include AVR cores, PIC cores, MIPS cores, the MSP430 CPU core or the older Intel 8051 processor. Micro-controllers implement several low power modes, which enable the developer to disable different components of the micro-controller in order to lower the overall power consumption. Due to the variety of possible implementations, the power consumption of micro-controllers can vary from several μW in low-power mode to a maximum or nearly 1 W. As the micro-controller does not run at full power all the time, its mean power consumption is often far lower than the maximum. In WSN nodes especially, the micro-controller mainly runs control tasks, while the heavier data processing calculations are usually done by a central remote server.

The wireless communications of WSN nodes require the use of a radio communication IC, which integrates both transmission and reception chains. The reception chain integrates a low-noise amplifier to amplify the received signal and the transmission chain includes a power amplifier. Depending on the wireless communications protocol, the emission power can increase up to several hundreds mW. Some protocols also require the node to periodically emit control messages or to open reception windows to check for incoming messages. Due to this, the radio IC has to be periodically turned on and consumes power, which raises the mean power consumption of the WSN node. Due to the relatively high power consumption of the amplifiers used in both reception and transmission chains, and the relatively low power consumed by micro-controllers, wireless communication accounts for a significant part of the total node power consumption. It can be noted that the use of wired communications could significantly decrease the power consumption, but comes with significant constraints on the use-case of the node.

Due to the variety of use-cases for WSN, it is difficult to quantify a priori the power consumption of the sensing element. While some sensors, such as temperature or voltage sensors, are easy to integrate and consume little power, others, such as gas sensors, consume much more energy. Sensors usually integrate a transducer, which converts the sensed physical data to an electrical value (voltage or current), which goes through a signal conditioning circuit and is converted to a digital value with an ADC. Depending on the sensed data, the transducer can be a simple thermistor, or can be based on complex electro-chemical reactions. Moreover, the signal conditioning stage can be complex, including multiple filters and preamplification stages. Additionally, some sensitive sensors have to be calibrated before use, which consumes power. Although the impact of sensors can be minimal on the overall power consumption, it can also, in complex use-cases, be the most power consuming part of a sensor.

Finally, the quiescent power consumption of the power converters (SMPS and LDO) also has a significant impact on the system. This quiescent power consumption corresponds to the energy consumed by the active circuitry of the converter (feedback loop, comparators, voltage generator...) which is always powered regardless of the circuit activity. WSN nodes usually use small scale power converters, which have a low quiescent current consumption of several μA. Although a dozen μW power consumption is lower than the consumption of other active components by several orders of magnitude, this represents the majority of the power consumption of the device when all components are in a low power mode. Therefore, minimizing the quiescent current of the power converters reduces the mean power consumption of the node and has a significant impact on the power consumption of the platform.

Fig. 3.2, Fig. 3.3 and Fig. 3.4 show the result of a power consumption estimation obtained with Wisebatt [152] for a commercial WSN node, powered by a primary battery, over its whole lifetime. The node is based on Wi6labs LoRa platform, built around a STM32L0 micro-controller and a SX1272 RFIC. The device is used to periodically measure temperature and humidity values, and immediately transmit them over a LoRaWAN network. Since the device is in standby 99.99 % of the time, the power components account for a large majority of the spent energy, as shown in Fig. 3.2. If the SMPS converter power consumption is ignored (Fig. 3.3), the power consumption of communication and processing (micro-controller) tasks can

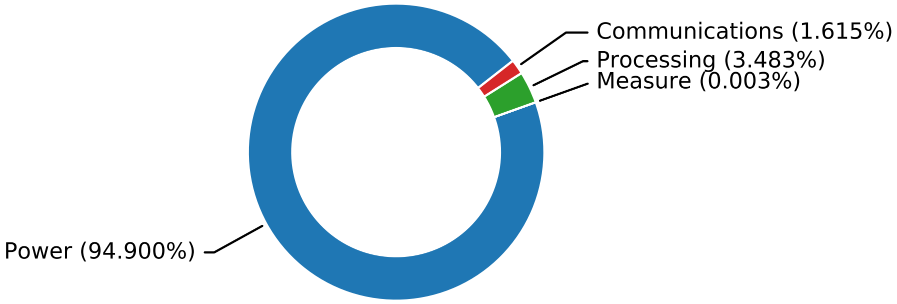


Figure 3.2: Energy consumption repartition per component type with main SMPS converter.

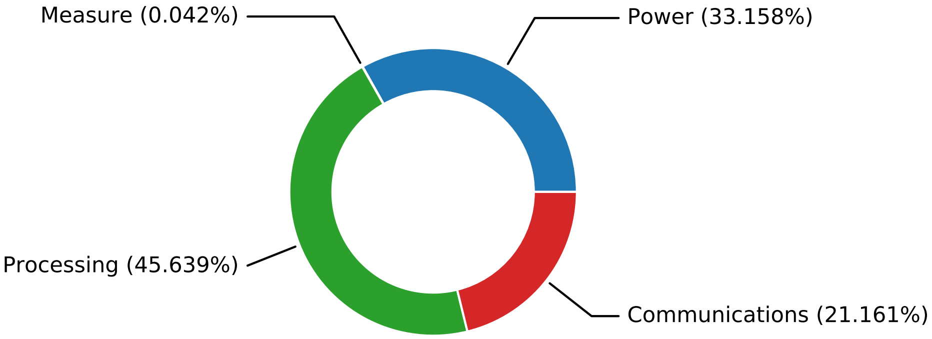


Figure 3.3: Energy consumption repartition per component type without main SMPS converter.

be observed, alongside the consumption of other power components, such as load switches, which stay powered all the time. Although this can not be seen in this graph, the radio-communications IC has the highest instantaneous power consumption. Finally, Fig. 3.4 shows the power consumption repartition per task. Despite the very low power consumption of components in standby mode, the sleeping power consumption accounts for 44.5 % of the energy consumption. Radio-communications tasks amount to a total of 35.6 % of the consumed energy. In particular, using a more efficient MAC protocol could reduce the time spent listening to the radio channel, and could significantly reduce the energy consumed by this task.

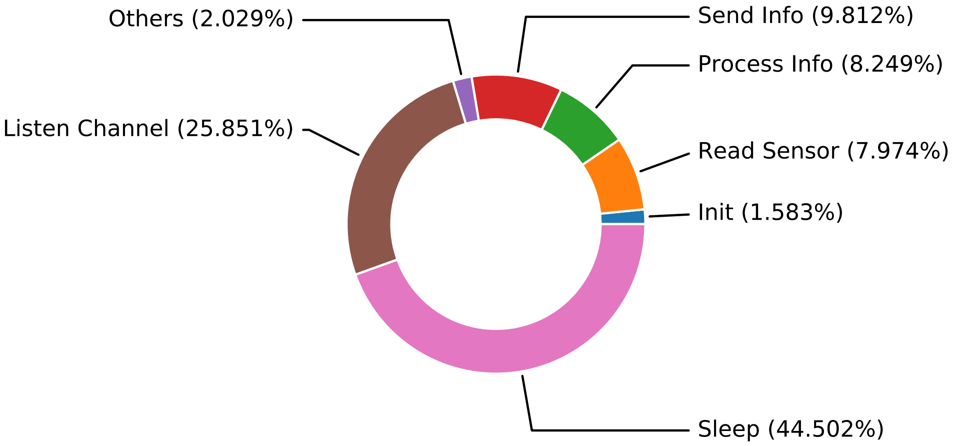


Figure 3.4: Power consumption repartition per task for a commercial WSN node.

3.1.2 Power Consumption Reduction

Multiple strategies have been deployed in order to reduce the power consumption of embedded systems. A first method is to reduce the power consumption of electronic components when they are active.

On the other hand, for many components, the frequency and voltage can be varied dynamically. This process is called Dynamic Voltage and Frequency Scaling (DVFS) [153]. In a typical use-case, the working frequency and supply voltage of a component will be reduced when the workload is low in order to save power, while they will be increased when needed to process a large workload. Fig. 3.5 shows the effect of varying the frequency and voltage of a component on the power and energy consumed for the same task execution. A lower colored area means that less energy is consumed. The voltage reduction has more impact than frequency reduction when reducing the dynamic power, since its squared value is used in P_{dyn} calculation (3.2). However, voltage cannot be reduced under the transistor switching threshold, which is determined by the silicon fabrication process. Moreover, reducing the voltage can limit the maximum switching frequency, as it reduces the power available to properly drive all gate signals with short enough rising and falling edges. Frequency variation has a more limited impact on the component power consumption but is more flexible. The maximum frequency is set by the design of the component, and corresponds to the inverse of the critical path, i.e. the longest path in a set of combinatorial logic of

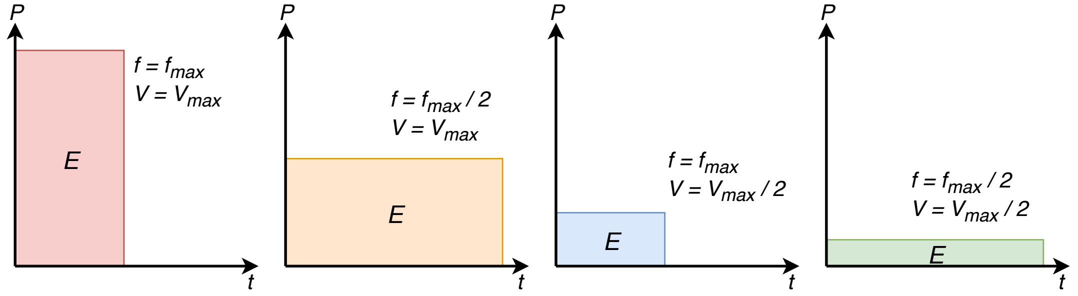


Figure 3.5: Energy consumed for a task, for different voltages and frequencies.

the chip. This critical path can be cut, up to certain limits, by inserting sequential logic (i.e. registers and flip-flops) as an intermediate step. While the frequency could theoretically be reduced down to 0 Hz, it is usually, in practice, limited by the capabilities of the embedded clock generators.

Although DVFS has been used for a long-time in a variety of system, from smartphones and laptops to datacenter servers, there is few reports of DVFS implementation on resource constrained WSN nodes. An explanation is that the implementation of DVFS strategies requires additional work and resources. The firmware of the node has to implement the functionality and regularly monitor the device activity in order to decide which is the most optimized state. Moreover, the components have to be able to support quick variations of frequency and voltage. In particular, voltage converters (LDO and SMPS) have to be designed to enable a programmable output voltage and support quick output voltage variations without losing stability, which increases the cost of the converter, and thus of the complete solution. Nevertheless, some solutions can be implemented by using programmable potentiometers in the converter feedback loop. Moreover, recent micro-controllers tend to implement different options to implement low power running modes.

But some recent electronic components such as micro-controllers, radio communication ICs and even some SMPS, also implement low-power modes. In these power states, the functionality of the component is severely degraded or completely stopped, but the power consumption drops drastically, down to a less than a μA, while SMPS converters, micro-controllers and radio-communications ICs draw several μW to tens of mW when in active mode. Therefore, it becomes interesting for the WSN node to execute its task as fast as possible to spend most of the time in these low-power modes. This power consumption reduction technique is known as duty-cycling. Since the power consumption in low power modes is several orders of magnitude lower than in active mode, it is more interesting to run tasks as quickly as possible, whereas DVFS is more useful when the component still consumes power in idle state (e.g. desktop processors). This ensures that the device stays in sleep mode for as long as possible, thus reducing the mean power consumption. Duty cycling is a very efficient power consumption reduction method for devices which only operate intermittently, which applies to a lot of IoT use-cases. [154], for example, presents a wake-up system which makes use of sleep modes to detect high frequency sound waves while consuming as little as 0.56 mW. In this example, the Micro-Controller Unit (MCU) is only active when measuring the received frequency, and sleeps while no high frequency wave is detected. However, some use-cases which require extremely low latencies or

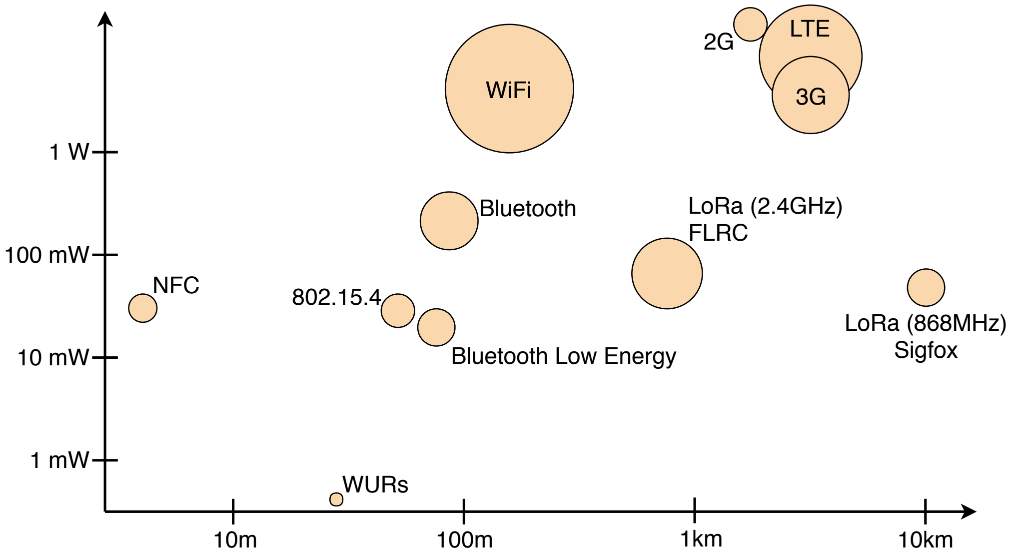


Figure 3.6: Comparison of commonly used protocols for IoT communications. A bigger circle indicates a higher data-rate.

continuous operation are not compatible with this method, thus a compromise has to be made between energy consumption and performance [155].

Radio-communications represent a significant part of the power consumption in an embedded system. This is notably due to the use of amplifiers in the radio chains. Low noise amplifiers are used when receiving the signal, and power amplifiers are used to amplify the emitted signal to up to 200 mW, depending on the communications protocol. Moreover, some protocols such as recent WiFi versions, incur heavy signal processing computations, which further increase the energy consumption of radio-communications [156]. Thus, it is interesting to reduce the communications time, in order to better duty-cycle the radio-communications tasks. To this effect, multiple low-power transmission protocols have been developed, as shown in Fig. 3.6. It can be observed that not all protocols can be used for all use-cases. For example, Bluetooth Low Energy (BLE) can only be used for short range, while LoRa [157] and SigFox [158] have been developed to enable long range communications with a limited power consumption. The choice of the communications protocols is tightly coupled with the end-user application, and is outside the scope of this work.

A large part of these communication protocols requires the WSN node to periodically emit a message to keep its synchronization with the network. Due to these mechanisms, WSN nodes have to periodically draw power to emit a beacon or open a receiving window. Multiple protocols [159][160][161] have been proposed in order to minimize the power required to synchronize two nodes. Another approach is to use Wake-Up Radio (WUR) [162][163] to completely remove the need for receiving windows. WURs are radio specifically designed to have an ultra low power consumption. The transmitting node emits a beacon addressed to the WUR. Upon reception of the beacon, the WUR wakes up the main radio from low power mode so that it can receive the message from the transmitting node. WURs power consumption can go down to a few μW, which is significantly lower than the mean power consumption compared to standard radio. Moreover, the use of an always-on WUR instead of a standard radio which only periodically opens a receiving window can lower the mean transmission latency. However, WURs are recent components and are not available off-the-shelf yet. More importantly, due to their low power consumption, they have a low sensitivity, often around -60 dBm to -50 dBm, which makes them unsuitable for long range transmissions on their own. To circumvent this problem, [164] proposes the use of both a WUR and LoRa radio, where WURs are used for short range communication in a cluster and LoRa is used to relay the gathered data to a central gateway. Moreover, no standard communication protocol have been designed to take advantage of WURs, and a custom protocol layer has to be developed, implemented and validated.

Without changing communication protocols, it is possible to reduce the radio communication power consumption by transmitting less data. A way to reduce the size of the transmission packet is to compress the data before sending it. Compression can be lossless, in which case the compressed data can be reconstructed to be an exact copy of the original, or lossy, in which case part of the original data is lost in the process. Lossy compression algorithms are typically used for images (e.g. JPEG), audio (e.g. mp3, Opus, Ogg...) and video (e.g. H.264, VP9, AV1...) files, where small details which can not be heard or seen by the human ear or eye are discarded to reduce the size of the file. Lossless compression is less efficient, but is suitable when no data losses are acceptable, typically for documents or raw measurement data. This type of compression exploits statistical redundancy and probabilistic model to create a dictionary which matches a repetitive string of data with a short compressed code. In the context of WSN nodes, this type of data compression has two significant drawbacks. First, in many use-cases, the size of data to send is limited to a few tens of bytes. Due to the small sample size of this data, there is usually little redundancy to be exploited by the algorithm. Secondly, the dictionary has to be sent along with the compressed message. The size of this dictionary further reduces the efficiency of the compression, especially when the data to compress is already small. Moreover, some communication protocols, such as LoRa and SigFox, have a very limited maximum payload size, where a dictionary and compressed message can not fit. Finally, running a compression algorithm can be complex for a micro-controller, which raises its power consumption.

For this kind of use-cases, alternative compression methods have to be used. The use of specific number representation can be used to optimize the number of bits required to represent each measurement. The use of fixed point representation enables a good compromise between precision and dynamic of the data. An alternative is the use of floating point representations on a reduced number of bits, such as half-precision floating-point format, as defined in the IEEE 754-2008 standard [165]. Based on the same principles, floating point values can be represented on even less bits [166][167] to further reduce the measurement size. Floating-point representations enable a higher dynamic fixed-point representations on a similar number of bits. However, the precision of the representation is not fixed, and high numbers close the the maximum representable value have to be rounded to the nearest representable value. On the contrary, fixed-point representations have a smaller dynamic but a fixed absolute precision 2^{-Q} where Q is the number of bits used to represent the fractional part. Thus, a compromise between the number of bits used to represent the measurements and the required precision. Fig. 3.7 and Table 3.1 respectively show

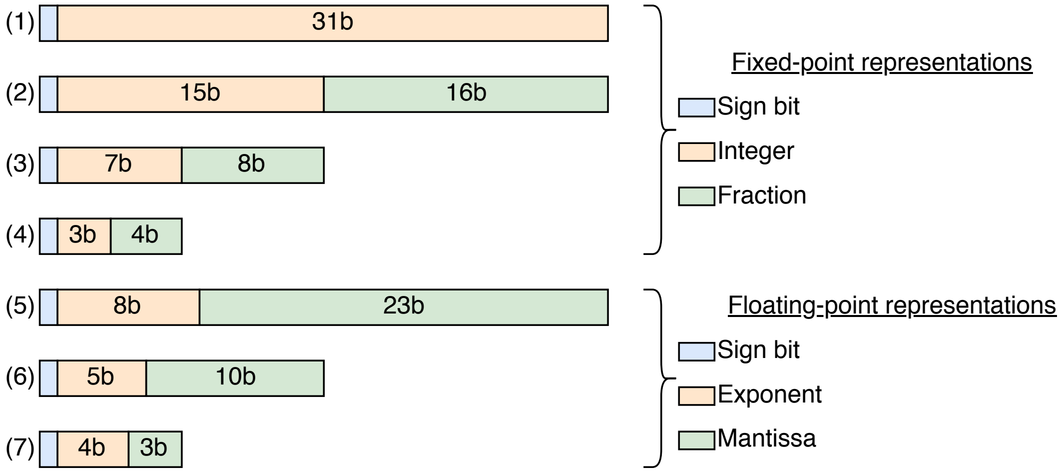


Figure 3.7: Examples of fixed-point and floating-point representations.

the structure and precision of different fixed-point and floating-point representation. As can be seen, floating-point representation can enable the representation of much smaller numbers. Moreover, by carefully choosing the exponent bias, floating-point representation can represent high numbers in much less bits, as shown in the last line of Table 3.1, although with a reduced precision.

	Dynamic	Minimum positive	Precision
(1) Signer integer	$-2^{31} \leq i \leq (2^{31} - 1)$	1	1
(2) Signed Q15.16	$-2^{15} \leq i \leq 2^{15} - 2^{-16}$	2^{-16}	2^{-16}
(3) Signed Q7.8	$-2^{7} \leq i \leq 2^{7} - 2^{-8}$	2^{-8}	2^{-8}
(4) Signed Q3.4	$-2^{3} \leq i \leq 2^{3} - 2^{-4}$	2^{-4}	2^{-4}
(5) Single precision floating-point	$-(2^{128} - 2^{104}) \leq i \leq 2^{128} - 2^{104}$	2^{-149}	Variable Minimum 2^{-149}
(6) Half-precision floating-point	$-(2^{16} - 2^{5}) \leq i \leq 2^{16} - 2^{5}$	2^{-24}	Variable Minimum 2^{-24}
(7) Minifloat (Exp. bias = -7)	$-(2^{8} - 2^{4}) \leq i \leq 2^{8} - 2^{4}$	2^{-9}	Variable Minimum 2^{-9}
(7) Minifloat (Exp. bias = 2)	$-(2^{17} - 2^{13}) \leq i \leq 2^{17} - 2^{13}$	1	Variable Minimum 1

Table 3.1: Dynamic and precision of multiple fixed-point and floating-point representations.

Another commonly used method for measurement compression when a single data is measured is to encode only the first measured value, and then encode the difference between a measurement and the previous one. The use of such a method can introduce redundancy to improve a usual lossless compression. Moreover, when the difference between measurements is sufficiently small, it is possible to encode these values in less bits than the full measurements. When the WSN node is sensing slow-moving values, such as temperature, the reduction in data volume can be significant. However, the use of a such a method must be known by the receiver, as a specific decoder must be implemented to reconstitute the different measurements.

3.2 Managing Harvested Energy

3.2.1 Power Manager in an Energy Harvesting System

Embedded systems are usually powered by batteries, which requires the user to recharge or change the battery when it is empty, depending if the battery is rechargeable or not. In this type of use-case, a power manager can be implemented to minimize the power consumption of the device. Typically, the power manager monitors the device activity to detect which part of the device is used and power off the unused parts, or apply DVFS policies. This power manager can be implemented on a dedicated micro-controller which manages the platform or as a software module which is part of the main firmware. By minimizing the power consumption, the power manager tries to increase the battery life time and the useful life of the device.

In an energy harvesting WSN node, the autonomy of the WSN node depends on the battery capacity, the power consumption of the electronic components, but also on the harvested energy. If the power manager is only used to minimize the power consumption, then the purpose of adding energy harvesting capabilities is only to increase the battery life of the device. However, it is possible to use a more intelligent power manager to adapt the QoS of the node according to its energy capabilities. This way, a node can reduce its QoS when it harvests little energy in order to avoid completely depleting its energy storage, and profit from plentiful harvested energy and charged energy storage to increase the QoS of its application.

Using such a power manager theoretically enables the node to avoid depleting its energy storage, preventing it from stopping its operations, but also to avoid completely charging the storage and wasting the harvested energy. Moreover, if the QoS of the node is correctly adapted to the harvested energy, we can define a point, called the Energy Neutral Operation (ENO) point, at which the harvested energy is equal to the consumed energy, as detailed by *Kansal et all* in [168]. Thus, the power manager can be designed to enable perpetual operation of the WSN node by constantly trying to operate around the ENO point. The consumed energy is reduced with the QoS when little energy is harvested, and is increased by improving the QoS when more energy is harvested.

In such a use-case, time is divided in time slots of a fixed duration. The power manager is periodically executed, at the end of each time slot. It adapts the QoS according to the residual energy E_R in the energy storage and the harvested energy E_H during the time slot. Multiple power managers using this concept have been proposed by the academic world [169][170][171]. These proposals target WSN which use short range ultra-low power communications, where the delay between messages D_{TX} is short, usually counted in seconds or minutes. Due to this short D_{TX}, at least a few messages are transmitted during each time slot. Thus, the time slot can be defined to a relatively short duration. As the time slot duration decreases, the number of power manager executions increases, making it easier for the WSN node power management algorithm to converge to the ENO point.

Finally the power manager can also ensure the proper boot of the platform. When a node boots up, it usually has to join a network. Depending on the wireless protocol used, joining the network requires one to several message exchanges, which consumes power. This is especially challenging in energy harvesting application, if the node has harvested enough energy to boot up but not enough to join the network. In this case, joining the network can deplete the battery enough to make the node shut down. Two solutions can be performed to mitigate this issue. A first solution is to use to hardware comparator with hysteresis to generate the node boot up signal. The hysteresis thresholds are determined to ensure that the node has enough energy to join the network after booting up. A second solution is to use the power manager to monitor E_R after boot, and only try to join the network if sufficient energy is harvested. This solution is more flexible and more resilient. Indeed, if the transmission conditions are bad, the node may have to try multiple times before joining the network, each time consuming more power. Using the power manager enables waiting until sufficient energy is harvested before attempting to rejoin the network, while the node would fully empty its energy storage in repeated join attempts if it only uses a hardware comparator.

3.2.2 Power Manager Structure

Although the power manager can include extra-functionalities such as boot management, the main feature of the power manager is still to control the QoS of the node so that the consumed energy E_C^{Total} is equal to the harvested energy E_H. Thus, the behavior of a power manager is analog to a control algorithm, which takes the WSN node energy capabilities as input and outputs the delay between two consecutive messages D_{TX}. This type of problem is frequent in other domains, such as industrial control, robotics and automation, and multiple solutions have already been proposed to optimize control systems speed, precision and stability. These solutions can also be applied to power manager design.

A common solution found in control systems is the use of a Proportional-Integral-Derivative (PID) controller. PID controller is a closed-looped system designed to continuously control a process to obey a command. The PID controller continuously computes the error between the command and the result, and processes this error through three components: a Proportional component which corresponds to the error multiplied by a factor $\mathcal{K}$, an Integral component which represents the accumulation of the past errors, and a Derivative component which represents the prediction of future errors. The outputs of each component are weighted, summed together and passed through an adaptation function to generate the controlled output. An application of such a control system applied to power manager is shown in [169]. In this work, the SOC of the supercapacitor is compared with a reference SOC the input of the PID controller, which generates D_{TX}. The reference SOC can be set by the designer to fit different use-cases: a high reference SOC will quickly increase D_{TX} in order to take more risks, while a lower reference SOC will take less risks but offers more performance in terms of transmitted messages. PID controllers are easy to implement, even on low performance micro-controllers, due to their low complexity. However, the proportional, integral and derivative component coefficients are hard to tune to optimal values to find the required compromise between control precision, speed and stability.

A recent solution to reduce the complexity of tuning the control system to the application is the use of self-learning technologies. In [172], the authors present RLman, a power manager which uses Reinforcement Learning (RL). RL algorithms

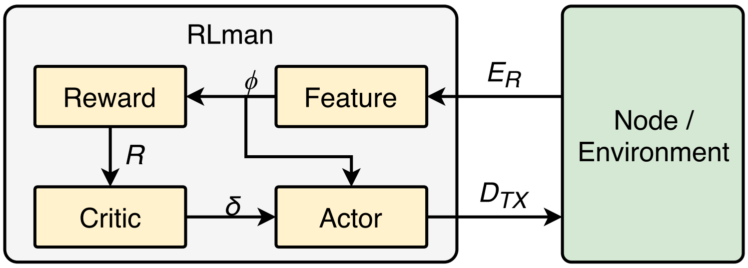


Figure 3.8: Structure of RLman algorithm.

try to maximize the cumulative *rewards* received by trying a set of actions in an environment. If an action is beneficial to the system in a specific environment, its *reward* will be higher, and will be favored by the algorithm as a better way to maximize the cumulative *reward*. Over the time, the system learns which actions maximize the *reward* in each environment. In the context of power management for energy harvesting WSN nodes, RLman uses E_R as an input and outputs the transmission frequency of the node. E_R is used to calculate a *feature*. The *feature* is associated to a *reward* sent to a *critic*, which represents an analyze of the previous decision. The *critic* and the *feature* are used by the *actor* to decide the next action. This structure is shown in Fig. 3.8. RLman is generic and can adapt itself to a large variety of energy sources and environments. It also has a low complexity, which enables implementation in the low resources of WSN node micro-controllers. However, it relies on high level theory and may be hard to tune for people foreign to RL theory. Moreover, it directly outputs the transmission frequency, which only applies to a use-case where the node sends its data after each measurement. For other use-cases, such as measurements without transmission and transmission of multiple aggregated data, the *feature* calculation step would have to be adapted, which could decrease the performance of the algorithm.

It is challenging to design a power manager which is efficient while still working with a wide variety of inputs for all use-cases. If order to ease these requirements, the power manager can be divided in two sub-functions, shown in Fig. 3.9. The first block is an Energy Budget Estimator (EBE). Based on the WSN node energy capabilities (i.e. E_R, E_H...), and its knowledge of the underlying platform (e.g. the energy storage characteristics...), the EBE delivers an Energy Budget E_B, which represents the amount of energy which can be consumed by the node until the next power management execution. The second block is the Energy Allocator (EA), which takes E_B as an input. Based on its knowledge of the use-case (e.g. power consumption of tasks, QoS requirements...), the EA decides how to spend E_B, i.e. which tasks to execute, and how much time to wait until the next power management algorithm execution. This separation is more flexible, as it enables easy modification of the power management scheme. For example, if the use-case evolves to add a new sensing tasks, only the EA has to be changed, while the EBE part of the power manager do not have to be modified. However, this separation also means that the power management policy is not fully optimized as a whole for the application.

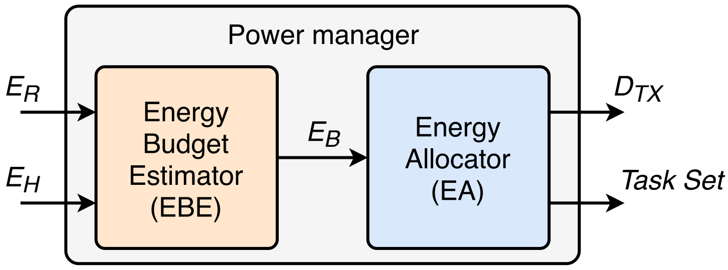


Figure 3.9: Power manager for energy-harvesting WSN node, divided in two functions.

3.3 Conclusions

In this chapter, we presented how a WSN node power consumption can be reduced to increase its autonomy. Furthermore, we showed how these methods are not sufficient in the case of energy harvesting WSN node. To enable a perpetual operation of such a system, a power manager is used to dynamically manage the power consumption of the node according to its energy capabilities, rather than simply reduce its power consumption. Such power managers implement an EBE function, which computes an energy budget E_B, and an EA function which allocates this E_B to the different task(s). Previous work has mainly focused on short range communications, while our platform uses LoRa communications, which is characterized by a long delay between transmissions and a higher instantaneous power consumption. Thus, the power manager must be adapted to fit these constraints. To the best of our knowledge, this work is the first to design a power manager for an energy harvesting LoRa WSN node. Chapter 4 details the adaptation of EBE algorithm for multi-source LoRa nodes, and chapter 5 shows how we can allocate energy in both single and multi-task systems.

Chapter 4

Energy budget estimators for Long Range IoT nodes

In order to properly manage the QoS of a device based on its power capabilities, a power manager must know how much energy it can allocate to each task. This amount of energy, called an energy budget and noted E_B, is estimated by the Energy Budget Estimator (EBE) algorithm. This algorithm bases its calculations on the different energy capabilities of the energy harvesting WSN node. The energy storage SOC is systematically used. The harvested energy is often required but is difficult to measure accurately, and is often estimated by a measurement of the variation of the SOC of the energy storage ΔE_R. This chapter presents the different types of energy budget estimation algorithms and describes some notable estimation methods. These algorithms are then adapted to a long range network (LoRaWAN) use-case, implemented and evaluated on the Wi6labs platform with the previously designed multi-source energy harvesting board. To the best of our knowledge, this is the first use of EBE algorithms on a LoRa WSN node.

4.1 Energy Budget Estimation Algorithms and Methods

4.1.1 Model-Based Energy Budget Estimation Algorithms

The first type of EBE algorithms is said to be “model-based”. This type of algorithm relies on a model of the energy source it is associated with. Using the characteristics of the energy source, the algorithm is able to predict the future harvested energy and to adapt the calculated energy budget E_B to past, present and future energy harvesting conditions. Such algorithms heavily rely on predictors to estimate the future harvested energy. Therefore, model-based algorithms combine two problems: how to accurately predict future amount of harvested energy and how to decide the best E_B using this parameter. As they use a model of an energy source to predict the future harvested energy, prediction algorithms are specific to the energy source they were designed for. Due to the progresses of energy harvesting technologies and use-cases, the design of energy harvesting prediction algorithms has been an active area of research during the past years, as detailed in the next paragraphs.

Although RF energy harvesting is not the most efficient energy source, it is easier to predict, as multiple works try to predict the future power of RF signal, in order to increase the quality of radio links. These work can be adapted to estimate the future power harvested with RF energy harvesting. For example in [173], *De Araujo et al.* model the link as a Markov chain and discover its parameters on the fly. This solution makes use of genetic machine learning algorithms to adapt itself to new radio conditions. This work is extended in [174] with the introduction of the notion of orientation in order to represent the tendency of the signal strength to increase or decrease. Machine learning is also used in [175] to predict the future link quality. In this work, link quality data is used to train a prediction model based on logistic regression, which enables a lower computational cost than alternatives such as Bayes classifiers and artificial neural networks.

The first energy predictor for solar sources for WSN has been proposed by *Kansal et al.* in [168], which introduced the concept of power management for energy harvesting WSN nodes. In this work, the predictor is based on an Exponentially Weighted Moving Average (EWMA) algorithm [176], with a window size of 24 hours divided in time slots of 30 minutes. The energy harvested during a slot is expected to be similar to the the energy harvested in the same slot of previous days. This assumption makes the algorithm sub-optimal for environments where the weather varies a lot. *Piorno et al.* [170] mitigate this issue with the Weather-Conditioned Moving Average (WCMA) algorithm, which is able to predict the output of a solar panel with a 10 % accuracy by taking into account both present and past weather measurements. [177] uses a phase displacement regulator to extend the WCMA algorithm and reduces its average error. A comparison of these different algorithms is given in [178], which shows the higher efficiency of the WCMA algorithm over EWMA and a neural network solution. However, WCMA requires more memory and computational power than the alternative solutions. [179] presents SunCast, an indoor solar energy harvesting predictor which learns predictable losses of luminosity due the the environment (e.g. trees, nearby buildings...). [180] proposes to estimate the harvested energy with EWMA when there are no clouds, and use meteorological models to scale this value when there are clouds. Finally, [181] uses Q-Learning, a reinforcement learning approach, to efficiently predict future harvested power based on past and present measurements.

Procarelli et al., in [34], proposes an energy predictor dedicated to forecast the availability of a wind energy source. The predictor uses a linear regression over a 30 s sliding window, using the last power measurement to predict the next one, and adapts the power converter settings to the harvested power. The Adaptive Response Rate Single Exponential Smoothing (ARRSES) is presented in [182]. This predictor is based on EWMA, but varies the smoothing factor proportionally with the fluctuations of input power, increasing its response time to weather variations. This approach is extended in [183] which considers longer forecasting periods.

Although most approaches are specific to a single energy source, some solutions can be adapted to fit multiple sources. [184] presents Pro-Energy, which predicts the future harvested energy by, once per time slot, comparing the energy harvested in the current day and saved energy harvesting profiles of past days. Pro-Energy is able to make short-term and medium-term predictions, and dynamically adapts its set of saved energy profiles, based on their age and similarities, to adapt itself to new conditions. [185] extends this work with the introduction of Pro-Energy VLT, which also dynamically adapts the lengths of time slots. [186] presents a study of both indoor light and thermoelectric energy harvesting in the context of Body area Networks (BAN), taking in account the impact of human activity. The system uses measurement to know the current state of the energy source and a Kalman filter to predict the next state of the source and how much energy it will provide. Finally, [187] proposes the use of Markovian models to simulate the behavior of piezoelectric and solar energy harvesting sources. The authors conclude that although both sources can be modeled by the same principles, the piezoelectric source is better modeled by a generalized Markovian model while the solar source is better modeled by a stationary Markovian model.

The prediction algorithm delivers the estimated future harvested energy to the model-based EBE, which then estimates the energy budget E_B. Multiple EBE algorithms have been designed to rely on energy prediction.

The first EBE is proposed in [168]. This EBE directly controls the duty cycle of the WSN node based on the difference between the estimated future harvested energy and the actual measurement energy input. Due to this, the EBE is directly impacted by the lack of precision of the EWMA predictor when the environment varies quickly. Moreover, the duty cycle is not balanced over the observation period, and thus is highly variable between periods of plentiful energy and periods where no energy can be harvested. It can be noted that this EBE does not take into account the residual energy E_R, which leads to power failures.

In [188], *Casgnetti et al.* present two EBE, named OL PM (Open Loop Power Manager) and CL PM (Closed Loop Power Manager) which both aim to reach ENO. The harvested energy E_H is used to estimate the recharge rate of the energy storage. OL PM calculates a duty cycle based on this recharge rate, while CL PM predicts the periods where the recharge rate is below a threshold, meaning not enough energy is harvested. In these so-called Zero Energy Intervals (ZEI), CL PM tries to save as much energy as possible. When enough energy is harvested, it uses the same policy as OL PM. A similar approach is taken in [6], where the power manager uses the EWMA predictor to decide the duty cycle during non ZEI periods and uses a specific NE PM (Negative Energy Power Manager) during ZEI, which focuses on avoiding power failures.

4.1.2 Model-Free Energy Budget Estimation Algorithms

Model-based EBE algorithms performance heavily relies on the quality of the used predictor. Accurate predictors tend to be computationally intensive, while algorithms targeted at constrained embedded environments tend to lack precision, especially with variable energy sources. Moreover, model-based EBE algorithms tend to apply a model to a particular type of energy source (e.g. pseudo-periodic model for a solar panel), which can not always be efficiently used with a different energy source. Therefore, there is a need for EBE algorithms which do not rely on a prediction algorithm.

The alternative to model-based algorithms is referred as "model-free" algorithms. This kind of algorithms takes only the platform energy capabilities (e.g. E_R, E_H, ΔE_R...) as inputs to compute E_B, and does not require an estimation of future

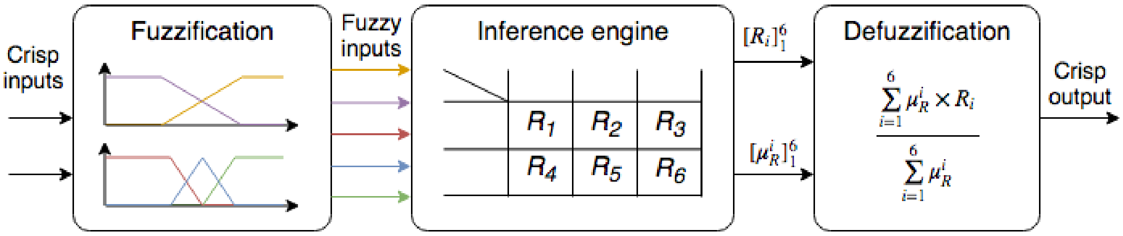


Figure 4.1: Structure of a fuzzy logic controller.

harvested energy.This type of algorithm is close to control systems, where a control loop tries to reach the optimal value of a process variable according to different inputs and constraints. In the case of EBE algorithms, the process variable is the energy budget E_B. Control systems have been widely implemented and studied, from home appliances to industrial and scientific systems. Thus, control theory and automation are active research areas, and a number of existing solutions, such as PID controllers [169], can be adapted to design model-free EBE algorithms.

Vigorito et al. proposed the first model-free EBE algorithm in [189] in 2007. The system targets ENO, explicitly focusing on generality and adaptivity, while keeping a good computational efficiency. The algorithm uses a linear-quadratic tracker (LQ-tracker) to solve the problem, using only the energy storage SOC as input. In order to get as close to ENO operation as possible, LQ-tracker tries to minimize the average squared difference between the current SOC and the initial SOC. When this difference is null, the system has consumed exactly as much energy as it has harvested. The control law coefficients are estimated using the gradient descent algorithm. This algorithm is executed online, so that the WSN node can adapt its control law to the environment condition and improve LQ-tracker control law dynamically. Moreover, the system computes an exponentially weighted moving average of the previous outputs in order to smooth the output and reduce its variance. Even without using an energy predictor, LQ-tracker manages to outperform *Kansal et al.* [168] by increasing the mean duty cycle of the platform, reducing its variance. More importantly, the battery SOC never drops to 0 %, ensuring continuous operation of the WSN.

[171] proposes P-FREEN, a prediction free energy neutral power management strategy which takes into account the energy storage inefficiencies. This algorithm takes both E_R and E_H as inputs and bases its decisions on time slots. If, at the start of time slot k, E_R is too low, or if the energy harvested in the previous time slot is lower than its last value, then P-FREEN outputs the minimal energy budget E_B^{min}. Otherwise, E_B is computed as a function of harvested energy E_H in the last time slot and the energy storage efficiency, capped at a maximum value of E_B^{max}. P-FREEN manages to deliver better performance than [168]. However, it requires E_H, which is hard to measure. As multiple calculations are based on E_H, it is difficult to replace E_H with the SOC variation ΔE_R without losing precision. Moreover, as E_R is compared to a hard threshold, it may create variations in the delivered E_B for a small E_R variation, which increases the variances of the output and reduces its stability.

In [190], the WSN node is designed to perform two actions: a storage action and a measurement and transmission action. For each of these actions, a fuzzy logic based controller is designed to compute the duty cycle, taking the percentage of data in the

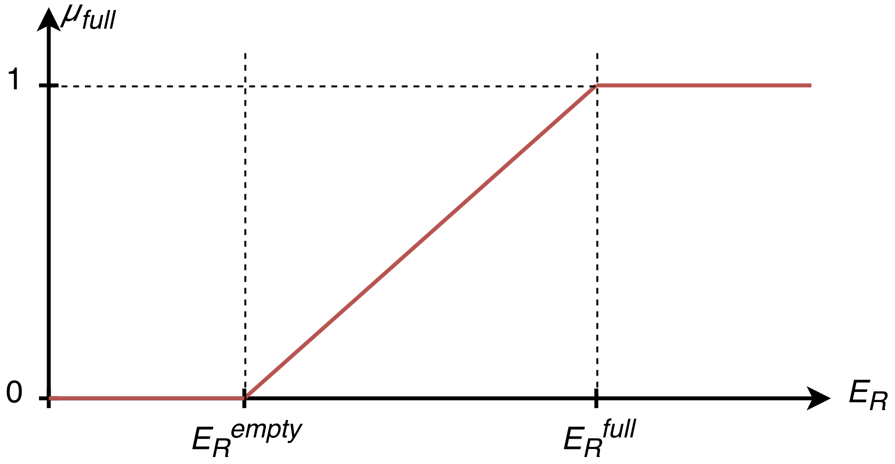


Figure 4.2: Graphical representation of the membership function (4.1).

data buffer and the 24 hour moving average battery SOC as inputs. Fuzzy logic uses 3 steps, described in Fig. 4.1. The fuzzification step transforms crisp input values in a set of fuzzy values through a set of membership functions, which is a representation of how much an input corresponds to a statement. For example, a fuzzy value μ_{full} could be equal to 1 if the energy storage SOC has reached a certain voltage threshold E_R^{full} and is considered full, 0 if the storage voltage is below a threshold E_R^{empty}, and a value between 0 and 1 when the voltage is between the two thresholds. Equation (4.1) shows such a membership function, and Fig. 4.2 shows its graphical representation. The fuzzy inputs are then processed by a rule-based inference engine, which makes each fuzzy input combination correspond to a certain rule and computes its activation value $\mathcal{A}_R^i$. Finally the defuzzification process computes a mean value of all rules weighted by their corresponding fuzzy inputs, to deliver the control output. Fuzzy logic can be seen as an extension of boolean logic, where there exist values between *true* and *false*. In the considered work, the output of the fuzzy controller is discretized to avoid heavy computations. However, storing the discretized fuzzy output consumes a lot of memory resources. Moreover, the fuzzy rule-set is tuned by trial-and-error for each specific source, and is hard to generalize to a large variety of applications.

$$\mu_{full}(E_R) = \begin{cases} 0 & \text{if } E_R \leq E_R^{empty} \\ \frac{-E_R + E_R^{full}}{E_R^{full} - E_R^{empty}} & \text{if } E_R^{empty} < E_R < E_R^{full} \\ 1 & \text{if } E_R \geq E_R^{full}. \end{cases} \tag{4.1}$$

Also based on fuzzy logic, [191] proposes Fuzzyman, a power manager which computes an energy budget E_B based on the residual energy in storage E_R and the harvested energy E_H. The membership functions and the rule-set are designed to be as simple as possible to avoid heavy computations, which removes the need to discretize the output of the fuzzy controller and free memory resources. Moreover, the proposed rule-set is generic enough to be independent of the used energy source. Fuzzyman significantly reduces the WSN node down time compared to [171]. However, it is hard to accurately measure E_H without consuming much power, which makes the implementation of Fuzzyman difficult in a real WSN node. In [192], the authors correct this point by proposing a Fuzzyman rule-set which takes E_R and its variation ΔE_R instead of E_H. For both Fuzzyman versions, tuning and optimization of the EBE is difficult. Indeed, although the proposed rule-sets are generic enough to be used in multiple contexts, design of fuzzy logic rule-set is complicated and must often be optimized through trial and error, which often breaks the generality of the solution.

Finally, another approach is proposed in [193]. In this work, *Hsu et al.* propose the use of Q-learning, a reinforcement learning algorithm, to compute the optimal power management policy for the WSN node in its environments. The system takes into account the battery SOC, the deviation from energy neutral operation and the deviation from the required QoS. Although the system increases the mean battery SOC, thus reducing the probability of down time, the obtained duty cycle is similar to [168]. Moreover, the algorithm requires the battery SOC E_R, the harvested energy E_H and the consumed energy E_C to operate. As the latter two values are difficult to efficiently measure with a reduced power consumption, the implementation of this power management scheme is difficult in the context of a real WSN deployment.

4.2 Application to Long Range IoT Nodes

4.2.1 Long Range Energy Budget Estimation Requirements

In this thesis application, the use of long range radio transmission technologies is considered. The delay between messages D_{TX} usually ranges from minutes to hours, but can even reach more than a day between two transmissions. This contrasts with state of the art proposals, where the delay between messages ranges from seconds to minutes, and the wait before the next power manager execution can be set to an hour or less. Thus, the time slot duration has to be set to a long time in order to ensure that at least one transmission is made during the time slot. This makes it much harder for the control algorithm to converge to the ENO point. Moreover, since the delay between two power manager executions is longer, the system can not quickly react to change in the environment and harvested energy. This creates a risk of depleting the energy storage and stopping operation if the power manager still provides the WSN node a high QoS while the harvested energy has decreased. To circumvent this problem, the power manager is executed after each transmission, so that it can quickly adapt to changes in the environment between two messages.

State-of-the-art EBE try to reach the ENO point as quickly as possible. In these works, one of the objective is to use the harvested energy as efficiently as possible. Thus, energy harvested when the energy storage is fully charged is considered lost, as it can not be used by the device. Power management algorithms try to avoid this situation by increasing the QoS of the node when the energy storage is sufficiently charged. In an industrial context however, a fully charged energy storage is not necessarily a problem. In such applications, QoS limits can be required by a customer. In this use-case, once the maximal QoS is reached, having more energy does not improve the performance of the WSN node. Thus, a smaller energy storage can be chosen to reduce the cost of the platorm.

Frequency band	Max. power	Max. duty cycle	Max. bandwidth	Note
433.05 to 434.79MHz	1 mW ERP -13 dBm/10 kHz if bandwidth $\geq$ 250kHz	—	—	♢
	10 mW ERP	10 %	—	
434.04 to 434.79MHz	10 mW ERP	100 %	25 kHz	
863 to 865MHz	25 mW ERP	0.1 %	—	♢♡
	10 mW ERP	100 %	—	△
865 to 868MHz	25 mW ERP	1 %	—	♢♡
868 to 868.6MHz	25 mW ERP	1 %	—	♠♡
868.7 to 869.2MHz	25 mW ERP	0.1 %	—	
868.4 to 869.65MHz	500 mW ERP	10 %	—	♠♡
	25 mW ERP	0.1 %	—	♢♡
869.7 to 870MHz	5 mW ERP	—	—	♢
869.7 to 870MHz	25 mW ERP	1 %	—	♢♡
2400 to 2483.5MHz	10 mW EIRP	—	—	—
5725 to 5875MHz	25 mW EIRP	—	—	—

♢ Analogue audio and video applications are excluded, except voice transmission if interferences mitigation techniques are implemented.
♠ Analogue video applications are excluded.
♡ Duty cycle can be ignored if channel access and interferences mitigation techniques (e.g. listen before talk, adaptive frequency hopping...) described in [194] are implemented.
△ Only applies to wireless audio and multimedia devices.

Table 4.1: Subset of legislation defined by [195] for common IoT use-cases and frequency bands. Recent evolutions (2018 and later) are not shown as the new rules have not been implemented to this documentation publication date. ERP = Effective Radiated Power.

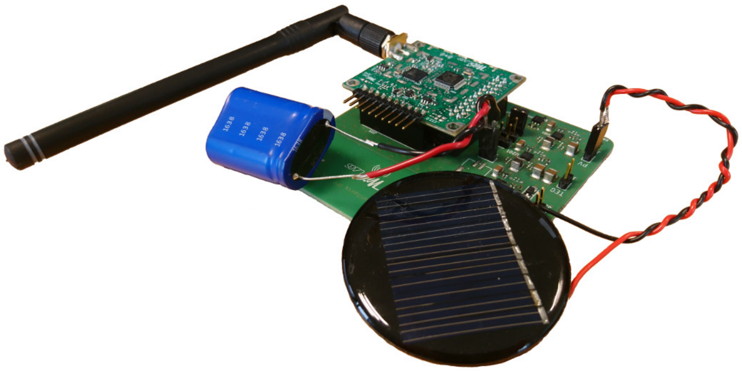


Figure 4.3: WSN node platform used for the experiments, powered by a solar panel.

Moreover, it can be noted that many wireless communication protocols operate in the Industrial, Scientific and Medical (ISM) frequency bands, especially in the 433 MHz, 868 MHz, 915 MHz, and 2.45 GHz bands. These frequency bands are free to use without any license requirements, but are regulated in order to reduce electromagnetic interferences between the multiple devices and transmissions. Especially, the legislation of sub-GHz bands (433 MHz, 868 MHz and 915 MHz) limits the duty cycle of the transmissions, as shown in Table 4.1. Depending on the region, the frequency band and the duration of the transmission, the WSN node must wait a minimum delay D_{TX}^{min} before emitting a new message. This imposes an upper QoS limit on the device, which has to be taken into account by the power manager.

Although those use-cases differ from the target of state-of-the-art model-free algorithms, the fundamental problem stays similar: delivering an optimal E_B under constraints, according to the node energy characteristics and its harvesting capabilities. Thus, there is an interest in evaluating the performance of these algorithms under those constraints, in order to quantify their performance for long range WSN. Moreover, most of these algorithms have only been evaluated in a simulation context. Therefore, a real-world evaluation of these algorithms is required, to validate the assumptions taken for the simulations.

4.2.2 Real-World Implementation of Energy Budget Estimators

A software library containing state of the art EBE algorithms has been developed in C, and implemented on a real world WSN node platform, shown in Fig. 4.3. Since the micro-controller has no Floating-Point Unit, the library was implemented using a portable fixed-point arithmetics library [196]. This portability enables the compilation of the library for multiple platforms, from a WSN node to a standard desktop processor. This enables the validation of the EBE library on a computer before being implemented on a target node and speeds up development time. Moreover, the library can be integrated as a software module in a simulator, which then executes the same code as the one on the WSN node. This limits the potential differences between simulations and real-world usage.

In this section, the focus is made on the impact of the E_B estimation step over the global efficiency of the power manager. Therefore, algorithms in which the energy estimation and allocation are tightly coupled, such as the PID controller [169] and RLman [172] are discarded. Moreover, in order to reduce its cost, the considered hardware platform does not embed components to measure the harvested and consumed energy E_H and E_C. Therefore, only the algorithms which use E_R and/or ΔE_R are implemented: LQ-tracker [189], Fuzzyman [191] and P-FREEN [171]. However, ΔE_R is a much less accurate measurement than E_H. This severely reduces the stability of P-FREEN, which relies heavily on the measured E_H. Thus, P-FREEN is not considered in this study. LQ-tracker can be implemented using only E_R as input. Fuzzyman in its original design also requires E_H, but has already been adapted to use ΔE_R. In this work, we consequently use a custom rule-set that uses ΔE_R as input instead of E_H, and uses the last energy budget $E_B\left[k-1\right]$ to determine the new E_B. The rules are detailed in Table 1 and only depends on the application and platform parameters and can be reused for a variety of applications and energy sources.

		E_R	
		Empty	Full
ΔE_R	< 0	E_B^{min}	$f(E_B\left[k-1\right])$
	$= 0$	$f(E_B\left[k-1\right])$	$E_B\left[k-1\right]$
	> 0	$E_B\left[k-1\right]$	E_B^{max}

Table 4.2: Fuzzyman custom rule-set.

In Table 1, $f(E_{B[k-1]})$ is a function defined by:

$$f(E_B\left[k-1\right]) = max\left(E_B^{min}, \frac{E_R - E_R^{empty}}{E_R^{full} - E_R^{empty}}.E_B\left[k-1\right]\right). \tag{4.2}$$

This study assumes that the WSN node periodically executes a single task, which enables benchmarking the different EBE algorithms with a simplified energy allocation scheme. In most use-cases, this task is composed of one or more sensor measurements, an eventual data processing, and the transmission of the processed measurement over the network. The energy consumption of this single task is supposed to be constant and known a priori, and is denoted E_C^{mono}.

The QoS is set in the library by defining a time base $\mathcal{D}$, usually set to 1 hour, and the minimal and maximal delay between two transmissions D_{TX}^{min} and D_{TX}^{max}. The minimal and maximal energy budget E_B^{min} and E_B^{max} for this application are then computed as (4.3) and (4.4), respectively.

$$E_B^{min} = \frac{\mathcal{D} \times E_C^{mono}}{D_{TX}^{max}} \tag{4.3}$$

$$E_B^{max} = \frac{\mathcal{D} \times E_C^{mono}}{D_{TX}^{min}} \tag{4.4}$$

Thus, to set a maximal or minimal QoS of υ ($\upsilon \in \mathbb{R}_{\geq 0}$) transmissions per time base $\mathcal{D}$ for example, D_{TX}^{min} is simply set to D/υ. The length of the time base $\mathcal{D}$ can be adapted to keep the computation within the fixed point format limits for a vast varying range of D_{TX}^{min} and D_{TX}^{max}. These parameters are dependent on the end-application and are customizable. D_{TX} (in seconds) is calculated from E_B and E_C^{mono} as:

$$D = \frac{\mathcal{D} \times E_C^{mono}}{E_B}. \tag{4.5}$$

The library also needs information about the underlying hardware energy capabilities. We denote V_{BAT}, V_{BAT}^{min} and V_{BAT}^{max} the current, minimal and maximal battery voltage. V_{BAT}^{min} is the voltage for which the platform stops functioning, or can be set with a margin before the battery is fully depleted. When the energy storage is a super-capacitor, its capacitance is defined. Based on these parameters, the current, minimal and maximal residual energy in storage E_R, E_R^{empty} and E_R^{full} are calculated.

Besides, we implemented two naive EBE algorithms: Linear_T and Linear_E. Linear_E calculates an energy budget between E_B^{min} and E_B^{max} as a pro-rata of V_{BAT} between V_{BAT}^{min} and V_{BAT}^{max}:

$$E_B = E_B^{min} + (E_B^{max} - E_B^{min}) \times \left(\frac{V_{BAT} - V_{BAT}^{min}}{V_{BAT}^{max} - V_{BAT}^{min}} \right). \tag{4.6}$$

Linear_T calculates a fixed delay between D_{TX}^{min} and D_{TX}^{max} as a proportion of V_{BAT} between V_{BAT}^{min} and V_{BAT}^{max}. This delay is then converted to an energy budget based on E_C^{mono}. The full equation is given by:

$$E_B = \mathcal{D} \times \frac{E_C^{mono}}{D_{TX}^{max} - (D_{TX}^{max} - D_{TX}^{min}) \times \left(\frac{V_{BAT} - V_{BAT}^{min}}{V_{BAT}^{max} - V_{BAT}^{min}} \right)}. \tag{4.7}$$

These two algorithms are used as a performance baseline to evaluate the advantages of more advanced control algorithms. Their advantage is their simplicity, as no settings are required beyond the common platform settings. Finally, to avoid sharp E_B variations due to temporary change in energy harvesting conditions, all algorithms outputs are smoothed by a low pass filter with a parameter $\gamma = 0.9$:

$$E_{B filtered}\left[k\right] = \gamma \times E_B + (1 - \gamma) \times E_{B filtered}\left[k - 1\right]. \tag{4.8}$$

4.2.3 Simulation Comparison of Energy Budget Estimators for Long Range IoT Nodes

All the selected algorithms can be tuned in order to improve their performance. The library has been designed to be portable, which enables the algorithms to be tuned in a simulation framework using the same code and same results as for the platform. Indeed, it takes more time to tune an algorithm with a measurement campaign than with simulation. This is particularly true for algorithms with many settings, such as Fuzzyman. In this subsection, we compare the performance of four algorithms: Fuzzyman, LQ-Tracker, Linear_T and Linear_E.

Each EBE algorithm has its tuning capabilities. Linear_T and Linear_E only need the common platform parameters, V_{BAT}^{min} and V_{BAT}^{max}. However, these thresholds can be adjusted to increase the linear algorithms performance. By respectively increasing and reducing the V_{BAT}^{min} and V_{BAT}^{max}, the energy storage is considered to be fully depleted or charged more quickly. This can be used to better conserve energy, by settings a high V_{BAT}^{min}. On the other hand, performance can be increased by reducing V_{BAT}^{max}, which makes the EBE output higher E_B values for the same E_R and deliver E_B^{max} even if the storage is not really fully charged.

LQ-tracker requires more parameters: it takes as input an energy target level E_{TRGT}, a step-size μ, a feature vector Φ and a parameter vector Θ. E_{TRGT} value is dependent on the application. μ is set by default to 0.001 and impacts how fast LQ-tracker converges. Φ and Θ default values can be modified to immediately fit the application. However, this is not necessary as the algorithm is iterative. Φ values are calculated at each iteration of the algorithm, while Θ is modified in order to converge towards an optimal value corresponding to the hardware platform and environment conditions. Increasing the parameter μ will enable a quicker convergence of the algorithm but may reduce its stability. E_{TRGT} is an important parameter, as it sets the residual energy storage that LQ-tracker will try to reach. A higher E_{TRGT} will make the algorithm more conservative, as it will tend to save more energy to reach its target, while a low E_{TRGT} will let it deliver higher E_B values but reduce the mean E_R. LQ-tracker also applies an exponentially weighted moving average filter (4.9) with a coefficient $\alpha \in [0, 1]$ to smooth the control output at time t u_t into a smoothed output $\overline{u}_t$. The higher α is set, the fewer past values will be used for smoothing. Finally, a coefficient $\beta \in [0, 1]$ is used to compute a moving average between short term and long term control. A small β with reduce variance but adapt more slowly to energy harvesting variations, while a high β will make the algorithm more reactive at the cost of increased E_B variance.

$$\overline{u}_t = \overline{u}_{t-1} + \alpha \left(u_t - \overline{u}_{t-1} \right) \tag{4.9}$$

Fuzzyman is the hardest one to tune, as it can be entirely redesigned to fit the application. E_R is processed through membership functions to compute whether the storage is full, empty or in-between. ΔE_R is processed to calculate if it is positive, negative or approximately equal to 0. The thresholds of these membership functions (E_{thr}^{full}, E_{thr}^{empty}, ΔE_{thr}^{pos}, ΔE_{thr}^{neg}) can be set by the user to define the EBE algorithm energy policy. Moreover, the complete rule-set can be redesigned to provide better performance. In this case, a compromise has to be made between performance, stability of the algorithm, but also the computational efficiency of each rule.

For our simulations, the delay between two messages is set between 15 minutes to 3 hours. The energy consumed by a Long Range Wide Area Network (LoRaWAN) transmission is measured to 140mJ with a SF7 spreading factor. We set V_{BAT}^{min} and V_{BAT}^{max} respectively to 3.7 V and 4.1 V, as for our real energy storage. The system was tested with a solar energy source, using EnHANTS [197] measurement data. The simulated solar panel is 3.2 cm × 4.2 cm with an efficiency of 10:%, like the one used in experimentation. The simulation results, without tuning, are shown in Table 4.5. The mean value $\bar{x}$ and standard deviation σ of V_{BAT}, E_B and D_{TX} are presented. Moreover, the number of transmissions Tx is counted. When the node has to transmit a message but V_{BAT} is under a threshold arbirarily fixed to 3.4 V, the transmission is aborted and is considered failed. The number of failed transmissions is denoted Tx_{fail}.

It can be noticed that all algorithms have some failed transmissions, meaning some transmissions were aborted due to V_{BAT} being lower than V_{BAT}^{min}. Most of them occur around a lack of data in the solar traces, which is kept to simulate a long period

Algorithm	Setting	Description	Value
Linear E	V_{BAT}^{min}	Low battery limit	$3.75\,V$
Linear T	V_{BAT}^{max}	Full battery limit	$3.85\,V$
LQ-tracker	μ	Step-size	0.005
	E_{TRGT}	Target SOC	36 % ($\sim V_{BAT} = 3.85\,V$)
	Φ	Initial feature vector	$[1, 0, -E_{TRGT}]^T$
	Θ	Initial parameter vector	$[2, -1, 1]^T$
	α	Control output smoothing coefficient	0.5
	β	Long-term/short-term decision coefficient	0.9
Fuzzyman	E_{thr}^{full}	"Battery full" threshold for membership function	$1.03 \times E_R^{empty}$
	E_{thr}^{empty}	"Battery empty" threshold for membership function	$0.85 \times E_R^{full}$
	ΔE_{thr}^{pos}	Positive ΔE_R threshold for membership function	0.35
	ΔE_{thr}^{neg}	Negative ΔE_R threshold for membership function	-0.5

Table 4.3: Settings used for EBE algorithms in this work.

		E_R	
		Empty	Full
ΔE_R	< 0	E_B^{min}	$0.95 \times E_{B[k-1]}$
	$= 0$	$0.95 \times E_{B[k-1]}$	$1.5 \times E_{B[k-1]}$
	> 0	$1.5 \times E_{B[k-1]}$	E_B^{max}

Table 4.4: Fuzzyman new ruleset.

without energy. Two algorithms, Fuzzyman and Linear_T, are more conservative, which can be seen as they keep a higher and more stable V_{BAT}, and thus emit less messages than the other two. E_B of Fuzzyman is high and unstable, which shows that it sometimes tries to deliver an E_B too high and can not stabilize. Linear_E and LQ-tracker, on the other hand, take more risks, showing a lower mean V_{BAT}, and are rewarded with a higher number of transmissions.

All algorithms parameters have been tuned in order to improve their performance as much as possible and maximize the number of transmissions. Table 4.3 shows the list of used settings for this work experiments. For Linear_E and Linear_T, V_{BAT}^{min} is set to 3.75 instead of 3.7 V, and V_{BAT}^{max} to 3.85 V instead of 4.1 V. This enables both algorithms to be more conservative when the energy storage is nearly depleted and take more risks when it is sufficiently replenished. The step-size μ of LQ-tracker is increased to 0.005, and Φ and Θ are left to their default values, fixed in [189]. A new rule-set is designed for Fuzzyman, shown in Table 4.4, while the fuzzification thresholds E_{thr}^{empty}, E_{thr}^{full}, ΔE_{thr}^{pos} and ΔE_{thr}^{neg} were respectively set to $1.03 \times E_R^{empty}$, $0.85 \times E_R^{full}$, 0.35 and -0.5. The simulation was run again with the new settings, and Table 4.6 shows the results.

	Fuzzyman		Linear E		Linear T		LQ-tracker	
	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ
V_{BAT} (V)	4.060	0.058	4.010	0.103	4.056	0.070	3.988	0.107
E_B (J)	0.454	0.163	0.446	0.125	0.364	0.155	0.529	0.107
D_{TX} (min)	29	36	24	23	33	27	21	28
Tx	18201		22624		16452		25814	
Tx_{fail}	7		31		13		31	

Table 4.5: Simulation before algorithm tuning.

	Fuzzyman		Linear E		Linear T		LQ-tracker	
	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ
V_{BAT} (V)	3.978	0.116	3.985	0.109	3.996	0.100	3.988	0.107
E_B (J)	0.530	0.100	0.528	0.107	0.526	0.117	0.528	0.107
D_{TX} (min)	20	27	21	27	21	27	21	28
Tx	26217		25947		25284		25800	
Tx_{fail}	36		33		28		31	

Table 4.6: Simulation after algorithm tuning.

After optimization, there is little difference between the performance of the different EBE algorithms. In the considered use-case, the energy budget E_B is limited between a minimal value E_B^{min} and a maximal value E_B^{max}. Thus all EBE algorithms have a similar behavior in limit cases where energy storage is sufficiently charged or empty. As all algorithms are tuned to maximize throughput, they deliver $E_B = E_B^{max}$ for an extended range of input values, as shown in Fig. 4.4, which shows the evolution of E_B as a function of the state-of-charge.

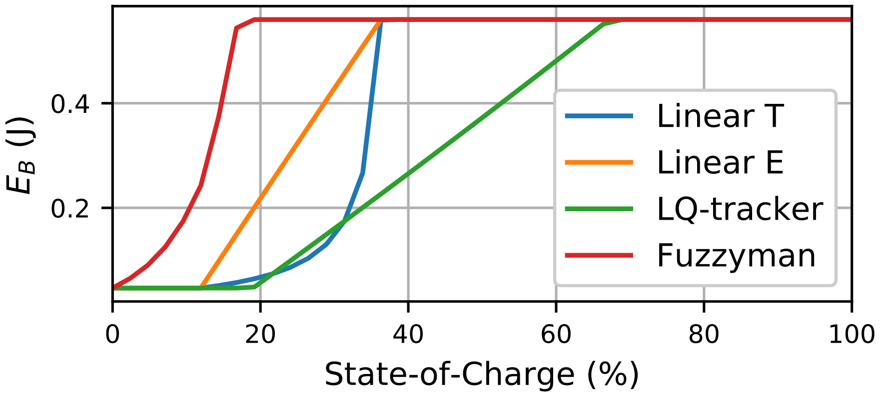


Figure 4.4: E_B as a function of the State-of-Charge.

This graph gives a quick insight of how the different algorithms behave at low and high states of charge. Indeed, it illustrates how much risks are taken by each algorithm for different SOC. However, this visualization has its limits when applied to algorithms which use more data inputs than only V_{BAT}, such as Fuzzyman, since it does not show the other inputs. Since all algorithms try to maximize E_B, they take less precautions and are more likely to fail if the environment conditions are bad for an extended time period. In the evaluated algorithms, LQ-tracker seems to be the most conservative, starting to reduce the delivered E_B when the state of charge goes under 70 %. The Linear algorithms and Fuzzyman are more aggressive, delivering maximum E_B when the state of charge is respectively over 35 % and 17 %.

4.2.4 Real-World EBE measurements

In order to confirm the simulation results, we set up an experiment on real WSN nodes to run the same comparison. During the experiments, each hardware device is powered by a solar panel which can provide up to 5 V and 40 mA from direct sunlight. The experiment was run for seven days in an office settings. Each algorithm is run on a different device. All sensors were located next to a north oriented window, and ran in parallel to reduce differences in ambient energy. After each transmission, the energy budget and delay to the next transmission are calculated and sent through an Universal Asynchronous Receiver-Transmitter (UART) link, along with V_{BAT} and a count of successful transmissions. This UART transmission is logged by a computer, parsed and analyzed to produce the graphs shown in Fig. 4.5.

The experimental results confirm the tendencies observed with the simulation results. The differences between both, especially regaring the number of failed transmissions, can be explained by the difference between the energy traces profile used in simulation and the real harvested energy profile. The number of successful transmissions is approximately the same for all algorithms, within a 15 % margin. Only one transmission failed during the fifth night for Fuzzyman. Although this performance difference is notable, it is sufficiently small to be overlooked in some industrial applications, and may be reduced over a longer experiment, with more weather variations. Thus, even though Linear_T and Fuzzyman are somewhat less efficient algorithms under these conditions, their use can still be justified by their other qualities, i.e. the simplicity of Linear_T and the customization possibilities of Fuzzyman. This shows that the algorithms are quantitatively equivalent when they are properly tuned, and confirms the results obtained through simulation. Qualitative differences can however be observed between the different EBE implementations.

As expected, LQ-tracker, by being more conservative, keeps its V_{BAT} higher than the other algorithms, which makes it more able to survive long periods without energy. On the contrary, Fuzzyman delivers a competitive number of transmissions but keeps its V_{BAT} much closer to the failure threshold, making it more sensible to lack of energy input, as seen with the failed transmission. As Fuzzyman is highly tunable, this could be optimized by a carefully designed rule-set, to provide characteristics closer to LQ-tracker.

Moreover, Fuzzyman and Linear_T show a high variance in the delivered E_B and delay D_{TX}, which can be explained by the quick decrease of E_B when V_{BAT} goes under a certain threshold, defined by the tuning of the algorithm. This quick E_B decrease translates into a much longer delay D_{TX}, and decreases the performance of the algorithm, despite the aggressive tuning. On the other hand, Linear_E and

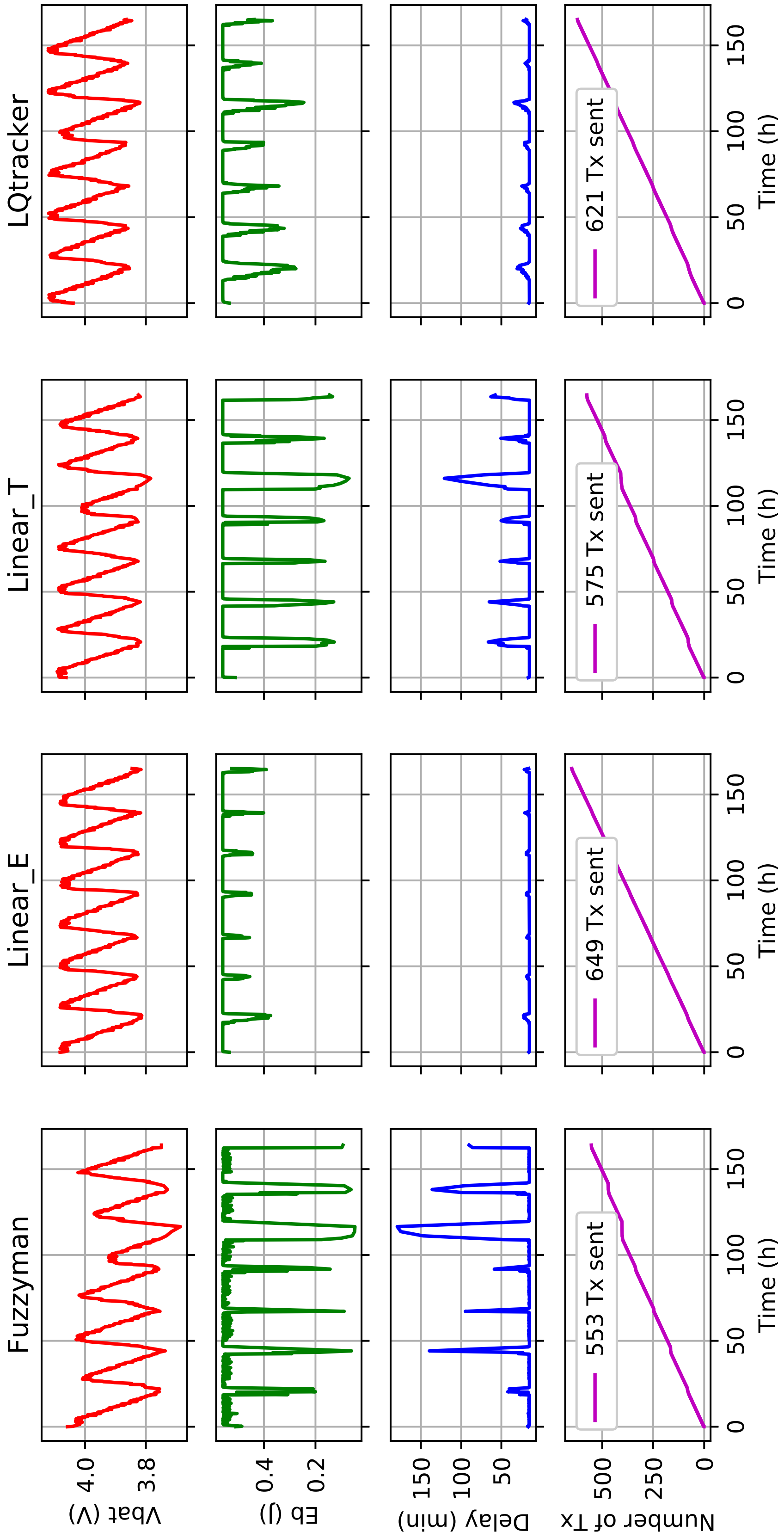


Figure 4.5: EBE algorithms comparison over seven days.

LQ-tracker reduce E_B more progressively, which translates to a lower mean D_{TX} and improved performance. This behavior also depends on how D_{TX} is calculated from E_B, and could be varied using a different energy allocation method.

Finally, it can be observed that all algorithms show a sharp rise of E_B when the energy storage gets replenished. As all the evaluated algorithms are model-free, they have no means to predict an approximate future energy state of charge. In our use-case, this is typically the case in the morning: the calculated D_{TX} is long at the end of the night, when the energy storage starts getting replenished. When this long delay D_{TX} is elapsed, the storage has been charging for a long time, and the EBE can deliver a higher E_B. This effect would be less important with a model-based EBE algorithm, as it would predict that energy was going to be harvested.

4.3 Conclusions

In this chapter, we focused on how an energy harvesting WSN node can estimate how much energy it can consume over a time period. State-of-the-art algorithms are presented, for both model-based estimators, which rely on prediction algorithms, and model-free EBE which draw inspiration from control theory solutions. This works presents the constraints of E_B estimation for long range energy harvesting WSN node, and shows a performance comparison of EBE algorithms in this context. Furthermore, the algorithms are implemented in a portable software library and executed on real-world WSN nodes. Experimentation confirms the results obtained through simulation. Both results show that the maximum performance which can be reached is nearly equivalent for all tested algorithms when the QoS is capped to a maximum value and the energy allocation policy is the same. However, the amount of fine-tuning required to reach this performance level varies between the different algorithms. Thus, the choice of algorithm is a question of ease of implementation and fine-tuning, more than a question of performance.

Chapter 5

Energy allocation for Energy Harvesting WSN Nodes

In addition to the EBE algorithm, energy harvesting power managers must embed an Energy Allocator (EA) step. In this context, EA focuses on how to best use the previously computed energy budget E_B to achieve the WSN node QoS requirements. Depending on the use-case and the node, this step can vary as the QoS requirements, the tasks of the node and/or the way QoS is measured differ. The typical tasks of a WSN node include measurement tasks, computing tasks to process the data and transmission tasks. These tasks can be defined by different properties, such as their priority, duration, energy cost and/or period. The use of a separate EBE algorithm isolates the EA from the energy capabilities of the platform. Thus, the EA can focus on the behavior and tasks of the node, enabling simple algorithm designs.

5.1 Related work

In some ways, EA can be seen as a task scheduling problem. In OSs where multiple tasks can be executed, a scheduler is used to decide which task is executed by the processor at a moment. In many general use-cases (e.g. general purpose computers, non-critical appliances...), these tasks have very few requirements, and a long delay to their execution has a low impact. In some other use-cases, called *hard real-time* systems, these delays have a high impact and it is important that tasks are executed before their deadline. In such systems, specific OSs called RTOSs are used to ensure that all tasks are correctly and timely executed. RTOSs make use of dedicated scheduling policies such as Rate Monotonic Scheduling (RMS) [198] and Deadline Monotonic Scheduling (DMS) [199] when tasks have a fixed priority based on their period, or Earliest Deadline First (EDF) [200] and its variants [201] when tasks have a dynamic priority based on their urgency. These scheduling policies are typically used when a missed deadline for a task has a high impact on the security of the system (e.g. plane control system...), of its user (e.g. modern car breaking system...), or when the task requires a low latency (e.g. audio/video applications...). However, these systems only take into account the deadline and/or the frequency of a task, with no concern for power consumption.

In order to better understand the following related works, we define a few terms which are commonly used in the field. Task schedulers can be implemented as *on-line* algorithms, which are executed directly on the WSN node, or *off-line* algorithms where the scheduling policy is computed before-hand on another more powerful machine. Off-line algorithms can be much more complex, as more computing power is available, but can not be used when new tasks with varying characteristics are dynamically created. On the other hand, on-line algorithms are more flexible but must be simpler, as less computing resources are available. The tasks to be scheduled can be characterized by their priority, their periodicity or their deadline. Tasks take a certain *length* or *execution time* to complete, which can be fixed or variable. In that case, the worst case execution time ($WCET$) can be defined. In the context of energy aware task schedulers, tasks characteristics also include the power consumed during their execution. Based on this, the energy consumed for a task execution can be computed. When a task has a variable length, the worst case execution energy ($WCEE$) is computed. The list of tasks which can be executed at some point by the task scheduler is defined as the *task set*. When the tasks are periodic, the *hyper period* of the task set can be computed as the least common multiple of the task periods.

Real-time scheduling for energy harvesting systems is a recent field of research. Some early works [202][203][204] focused on the use of DVFS to reduce the power consumption of the node according to its energy capabilities, with little modifications of existing real-time scheduling algorithms. On the other hand, an introduction to energy aware task schedulers has been published in 2012 in [205], which provides an overview of the complexity of real-time energy harvesting systems and existing solutions. This complexity is further detailed in [206]. In this work, the authors present a preemptive fixed priority scheduling policy where tasks are executed as late as possible (PFP_{ALAP}). The goal of this policy is to let the WSN node harvest energy to recharge its battery for as long as possible. This work also shows that the optimality of energy harvesting scheduling policies is much harder to prove and can go against intuition. Specifically, PFP_{ALAP} is evaluated against a similar preemptive fixed priority policy but where tasks are executed as soon as possible (PFP_{ASAP}). Although PFP_{ALAP} is supposed to let the WSN node recharge its battery for a longer time, the authors show an example task set where PFP_{ALAP} fully depletes the energy storage, while PFP_{ASAP} does not. To work-around this problem, the article shows an off-line method to compute the minimal size of the energy storage which ensures the task set schedulability, based on the expected harvested power and the task set hyper period.

In [207], the same authors focus on PFP_{ASAP} to prove its optimality, provided that all tasks consume the same amount of energy at each execution time unit and the harvested power is constant. Intuitively however, these conditions are not met in real-world conditions. PFP_{ASAP} is extensively evaluated against PFP_{ALAP} and PFP_{ST}, where tasks are executed As Soon As Possible (ASAP) when there is energy, and As Late As Possible (ALAP) while $E_R \leq E_R^{full}$ to recharge the battery when there is no more energy. PFP_{ASAP} is shown to have a low overhead and failure rate, but also lowers the average battery level. PFP_{ASAP} is extended in [208] to take into account the non-ideal behavior of supercapacitor energy storages. Also in [209], PFP_{ASAP} is extended with an harvested energy E_H predictor to create FPC_{ASAP} (As Soon As Possible Clairvoyant Fixed Priority), with the aim of extending the optimality of PSP_{ASAP}. However, this addition significantly increases the complexity of the algorithm, which makes its implementation more complicated for resource constrained WSN nodes.

Another early approach is presented in [210] with the Lazy Scheduling Algorithm (LSA). This algorithm extends the classical EDF policy by adapting its behavior to the battery level E_R. Tasks are usually executed as ALAP. However, when E_R reaches E_R^{full}, the task with the earliest deadline is executed ASAP. Thus, LSA can be seen as a switch between an ASAP policy when the energy storage is fully charged and an ALAP policy otherwise. Although basic, LSA is a significant improvement over classical real-time schedulers used in energy harvesting context. However, it requires a knowledge of E_H, which is hard to measure. In [211], *Audet et al.* introduce the concept of virtual tasks for task scheduling. Each virtual task represents a physical task, but smooths the energy required over time for a task execution down to the average energy usage ($STAM$ - Smooth To Average Method), or to a value calculated to create a virtual task set with 100 % utilization (i.e. the CPU is always executing a task of the virtual task set). The scheduling algorithm is applied to the virtual task set, and all virtual tasks are then replaced by their corresponding physical tasks. This method has been used on EDF, ALAP et LSA policies, and significantly decreases the probability of task deadline violation.

Algorithm 5.1: Earliest Deadline with energy guarantee (EDeg) algorithm

```
initialization;
while always do
    while tasks are pending do
        while $E_R > E_R^{empty}$ AND $Slack.energy(t) > 0$ do
            execute task;
        end
        while $E_R < E_R^{full}$ AND $Slack.time(t) > 0$ do
            wait;
        end
    end
    while no tasks are pending do
        wait;
    end
end
```

Another family of energy aware task schedulers, based on EDF algorithm and called Earliest Deadline with energy guarantee (EDeg), in presented in [212]. This algorithm, shown in Algorithm. 5.1 does not make any supposition on the nature and dynamics of the energy source, which makes it easily adaptable for multiple use-cases. The algorithm makes use of the slack time of the tasks, defined as the maximum time a task can be delayed on its activation to complete before its deadline and calculated with the function $Slack.time(t)$. This work also defines the slack energy of a task τ_i at time t as the energy surplus which can be used from t until the activation of τ_i and still guarantees that the task fulfills its energy and timing requirements, calculated with the function $Slack.energy(t)$. By extension, the slack energy of the task set is the maximum energy that can be consumed, starting at t, while still satisfying all timing constraints. This algorithm efficiently improves the performance of EDF algorithm for energy constrained WSN nodes. The complexity of the slack time and energy calculation is limited to $O(K \times Dd_{max}/d_{min})$ where K is the number of tasks, Dd_{max} is the longest deadline and d_{min} in the shortest period. Although this complexity is limited, the computation of slack time and energy may still create a significant computation overhead, especially for a large number of tasks with strong constraints. [213] extends this method to RMS and DMS algorithms, using the slack time to make the node idle to recharge its energy storage.

Another approach is the use of advanced algorithms to determine the optimal scheduling policy to take into account energy variations. [214] presents such an algorithm, where the system is modeled as a Markov Decision Process (MDP). In order to optimize scheduling, tasks are divided in subtasks, some being able to run in parallel with each other. At each state, a particular action, defined as a set of subtasks, is chosen and executed. Utility functions are used to compute the rewards associated with each actions. Like in reinforcement learning systems, the MDP learns the best actions (i.e. the best policy) to use at each state. However, this algorithm is computationally complex, as it grows exponentially with the number of subtasks. Thus, the authors implement a suboptimal version of the algorithm with a complexity of $O(M \times K)$ where M is the number of subtask queues and K is the number of tasks.

Some other works focus on task scheduling for specific applications. [215], for example, presents a periodic task scheduling framework for multi-core embedded systems with energy harvesting. The system maps new tasks to the processor cores with the lowest utilization. The system also includes DVFS and task slack management capabilities. However, it requires an energy predictor, which increases the required computations. [216] presents an algorithm which takes into account mesh networking, where a node must be able to route a packet to another node. This work introduces a priority policy where sensing tasks are executed first, their result stored in a buffer, and this buffer is only transmitted when the energy storage is sufficiently charged. However, this policy is hard to implement on long range networks, where the transmission buffer is usually small and limited to a few tens of bytes. A different approach is taken in [217], where a complete RTOS is designed with energy awareness as a requirement. The tasks can reserve resources such as CPU time, network bandwidth and energy. The OS enforces theses reservations, guaranteeing that the tasks meet their QoS and scheduling requirements. Like many new OS initiatives, however, the system supports very few platforms.

It can be noted that the previously cited solutions focus on real-time systems for which missing a deadline is not acceptable. However, in many applications, the non-execution of a task has moderate consequences. In a metering application, a missing measurement does not have a significant impact, and may even be corrected with an interpolation of the surrounding measurements. While the energy allocation problem for non real-time systems has not been as much explored as for real-time systems, some previous research exists. In [218], a system is presented which aims at finding optimal energy allocation to reach a battery energy target. The authors use Karush-Kuhn-Tucker (KKT) conditions [219] to compute the optimal energy allocation for each hour in a 24 hours horizon, based on an expected E_H value, and adapt the results when the measured E_H does not correspond to the expected value. This error correction system is shown to have a constant time complexity, and the full algorithm is shown to have little power consumption overhead.

In this thesis, the EA system has to divide an energy budget between one or multiple tasks. This energy distribution must be fair in regards with the priorities of the task. As the studied tasks do not have any real time constraints, the EA does not have to ensure the timely execution of all tasks. Instead, the EA is aimed at maximizing the QoS of the WSN node while preventing battery depletion. This approach has been previously explored in [220], but only considers the case of single task WSN node. This thesis extends the study to multi-task WSN nodes.

5.2 Single-task Energy Allocation

5.2.1 Single-Task IoT Nodes

This work first addresses the case of WSN nodes which only perform one task. Although this task can be composed of multiple subtasks, it is assumed that the whole task is executed as a whole. An example of such a system can be a WSN node which measures a value, processes it and transmits it immediately over the network. This approach is typically used for measurement reporting applications.

In term of energy allocation, this approach is the simplest one. Indeed, in this use-case, the role of the energy allocator is simply to translate the energy budget E_B into an inversely proportional delay between two messages D_{TX}. In other words, the only requirement of the energy allocation is to compute a descending function $D_{TX} = f(E_B)$. The function can be designed so that its shape fits the application, e.g. it takes more risks by computing a smaller D_{TX} for a large range of E_B values, or is more conservative and deliver $D_{TX} = D_{TX}^{max}$ for a range of small E_B values.

In this work, two functions for single task energy allocation are presented and compared. Both deliver a delay D_{TX} fixed between the minimal and maximal values D_{TX}^{min} and D_{TX}^{max}, which are fixed as parameters by the system designer. The first function computes D_{TX} between D_{TX}^{min} and D_{TX}^{max} on a pro-rata basis of E_B between E_B^{min} and E_B^{max}. Thus, the function $D_{TX} = f(E_B)$ is given by (5.1).

$$D_{TX} = D_{TX}^{max} - \frac{E_B - E_B^{min}}{E_B^{max} - E_B^{min}} \times \left(D_{TX}^{max} - D_{TX}^{min}\right) \tag{5.1}$$

The second function generalizes (4.3) and (4.4), which respectively compute E_B^{min} and E_B^{max} as a function of E_C^{mono}, $\mathcal{D}$, D_{TX}^{min} and D_{TX}^{max}. The formula can be generalized as (5.2).

$$E_B = \frac{\mathcal{D} \times E_C^{mono}}{D_{TX}} \tag{5.2}$$

The delay D_{TX} as a function of E_B can then be extracted as (5.3).

$$D_{TX} = \frac{\mathcal{D} \times E_C^{mono}}{E_B} \tag{5.3}$$

In a previous work [220], the authors obtain through a different reasoning an equivalent formula, shown in (5.4), where P_S is the power consumption in sleep

EBE		LQ-Tracker		Linear E	
EA		Inverse	Ramp	Inverse	Ramp
$V_{BAT}(V)$	Mean	3.962	4.002	3.995	4.057
	Std. deviation	0.075	0.064	0.089	0.060
$E_B(J)$	Mean	0.477	0.527	0.427	0.479
	Std. deviation	0.136	0.084	0.134	0.134
$D_{TX}(min)$	Mean	22.3	25.5	21.8	32.1
	Std. deviation	21.6	26.7	10.9	23.2
Transmitted messages		2362	2130	2527	1683

Table 5.1: EA algorithm comparison for different EBE over 40 days.

mode and d_T is the duration of the task. It can be noted that both this formula and (5.3) are similar if P_S is considered negligible and equal to 0 μW.

$$D_{TX} = \frac{E_C^{mono} - d_T \times P_S}{\frac{E_B}{\mathcal{D}} - P_S} \tag{5.4}$$

This approximation holds in the case of long range transmission as E_C^{mono} is generally higher than in traditional mesh networked WSN nodes. Indeed, in our use case, E_C^{mono} is equal to approximately 140 mJ, P_S is less than 80 μW, depending on the board used with our platform, and d_T is less than 10 s, depending on the network transmission parameters. Thus, the product $\tau_T \times P$ is always less than about 0.8 mJ, which is significantly lower than E_C^{mono}. On the denominator side, $\mathcal{D}$ is set to 60 minutes, and $E_B^{min} = (\mathcal{D} \times E_C^{mono}) / D_{TX}^{max} \approx 4.7\, mJ$, which makes the ratio $E_B/\mathcal{D}$ at least equal to 778 μW. Thus, in our use-case, we always have $E_B/\mathcal{D} \gg P_S$. So, as the P_S contribution is small in both sides of the fraction, it can be safely considered as equal to 0. The practical impact of this decision will be that the computed D_{TX} will be slightly lower, thus increasing the QoS but slightly increasing the risk of energy storage depletion.

5.2.2 Experimental validation

Both these energy allocation functions were implemented, tested and compared. As for EBE algorithms, these functions were implemented as a portable C library to help validation and speed up the development. The same sensors and energy sources as the EBE comparison in Chapter 4 were used. Out of four deployed sensors, two used LQ-Tracker and two used Linear_E as the EBE solution, in order to check the results consistency for different used EBE. For each of these two nodes, one used the *ramp* formula presented in (5.1) and the other used the *inverse* formula presented in (5.3). All WSN sensor nodes ran in the same location for approximately 40 days, sending logs of each transmission and energy capabilities over an UART link to a computer. The results of this experiment are traced in Fig. 5.1 and performance for all EBE+EA couples are shown in Table 5.1.

A first observation is that the choice of EA function has an important impact on the performance of the solution. In this case, the use of the *ramp* function implements a more conservative policy than the *inverse* function. This translates in a higher delay D_{TX} despite a higher energy budget E_B, which enables the node

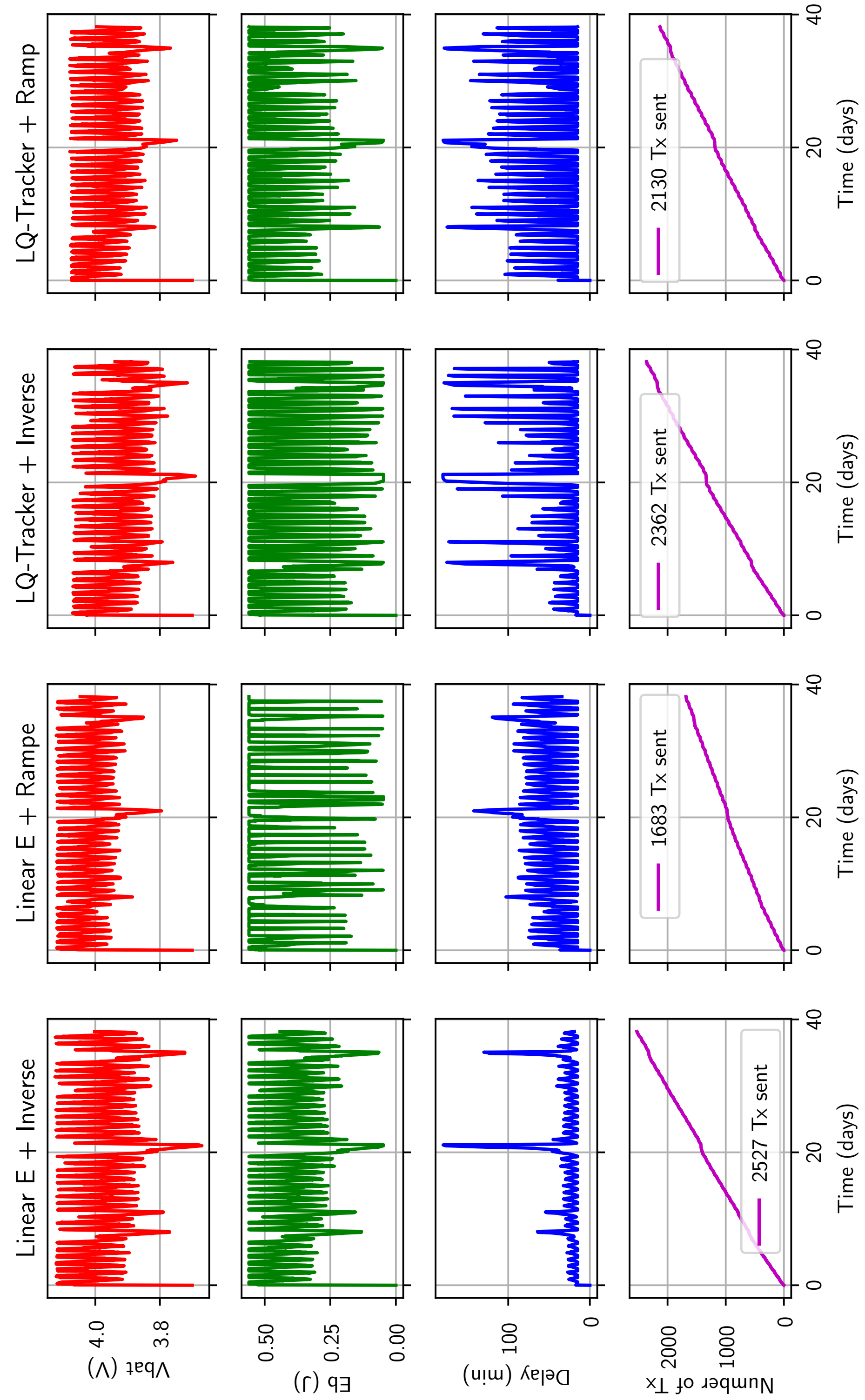


Figure 5.1: EA algorithm comparison for different EBE over 40 days.

Data aggregation		Disabled		Enabled	
EA		Inverse	Ramp	Inverse	Ramp
$V_{BAT}(V)$	Mean	3.937	3.947	3.973	3.977
	Std. deviation	0.097	0.088	0.083	0.065
$E_B(J)$	Mean	0.395	0.418	0.484	0.509
	Std. deviation	0.188	0.184	0.124	0.102
$D_{TX}(min)$	Mean	40.3	60.7	107.4	156.9
Messages transmitted		1004	668	377	258
Data transmitted		2008	1336	3770	2580

Table 5.2: Comparison of QoS with and without data aggregation.

to keep more energy in storage, as shown by the increase in the mean V_{BAT} value, but also decreases the QoS as less messages are transmitted. This experiment shows that, for single task WSN nodes, the choice of the $D_{TX} = f(E_B)$ EA function is a compromise between performance, in terms of QoS, and energy safety.

It can be noted that, when the EA function is a ramp, the performance of EBE algorithms differs by approximately 20 %, which is slightly more than the previously observed results in Chapter 4. One explanation is that the Linear_E EBE algorithm shows much more variations on its E_B output than LQ-Tracker, as shown with the difference in standard deviation. Due to this, Linear_E decreases much more quickly its E_B output when V_{BAT} gets lower. Due to this, the conservative *ramp* function tends to deliver higher D_{TX} more often, which increases the mean D_{TX} and decreases the QoS.

5.2.2.1 Aggregation: Increasing QoS for Single-Task Sensors

In the previous experiment, QoS is measured as the number of transmitted messages over a period of time. However, an alternative is to consider the number of transmitted measurements instead, and transmit multiple measurements per transmission. This scheme reduces the mean energy consumed each time the node wakes up, as the transmission is not always carried out. As less energy is consumed, the energy in storage E_R is higher, and the number of performed measurements may be higher than if it is directly transmitted.

In order to test this hypothesis, the previous experiment has been adapted to add data aggregation capabilities. The sensor performs both temperature and humidity measurements, which both have to be transmitted. Thus, at least two measurements are transmitted in each message. Using Wi6labs platform and proprietary WAP (Wi6labs Applicative Protocol) protocol, up to 11 measurements can be transmitted per LoRa frame, each encoded as a 5 bytes packet, including metadata. Thus, in our use-case, up to 5 sensor measurements (temperature + humidity) can be transmitted per frame.

When data aggregation is disabled, temperature and humidity measurements are formatted as WAP packets and directly transmitted. Thus the transmission chain is only powered for a 10 bytes transmission. On the other hand, when data aggregation is enabled, the WAP packets are stored in a buffer. This 50 bytes buffer is only transmitted when it is full, i.e. when 5 consecutive temperature + humidity measurements have been performed and stored. After each action (measurements or measurements + transmission), a new E_B and D_{TX} is computed, using the same code as in previous experiments. However, the new calculations are only logged after a transmission. Thus, as less transmissions are performed when data aggregation is enabled, less V_{BAT}, E_B and D_{TX} values are logged, which reduce the precisions of statistics over these values. The experiment has been run for 4 weeks on 4 nodes. All use LQ-Tracker as the EBE algorithm. Two use the *ramp* function and the two others use the *inverse* EA function. For each pair of WSN node using the same EA function, one is run with data aggregation enabled while the other has it disabled. Results are presented in Table 5.2.

As expected, this system reduces the number of transmitted frames but increases the number of transmitted data packets. In our system, this effect is particularly visible, as the temperature and humidity sensors use much less energy than a radio transmission. Thus, transmitting data only once out of five node wake-up saves a lot of energy, which lets the system wakes up more often to make more measurements. In particular, it can be observed that the number of transmitted frames is not simply divided by J when J data are aggregated. Indeed, as less energy is used each time the node wakes up, the delay between consecutive task executions is lower, which reduces the delay between transmissions.

These results show that such a system is an efficient way to increase the number of sensed data for an energy constraint WSN node. However, it is not fit for use-cases which requires reactivity through frequent data upload, as the number of transmissions is severely reduced, thus increasing the latency. Moreover, in LoRa systems, the data is often timestamped when received by the gateway. Using a data aggregation system, the delay between the different measurements is not always equal. Thus, the data would have to be timestamped before its transmission, which would add additional overhead on the packet, as the timestamp would also have to be sent. Also, the node would be required to implement time synchronisation logic, which requires additional resources. An alternative would be to estimate timestamps on the decoder side, using an interpolator. This method would avoid the overhead on the transmitted frame and required resources, but is less accurate as it does not compute the exact values. Thus, this type of system is not fit for all use-case.

5.3 Multi-task Energy Allocation

5.3.1 Multi-Task IoT Nodes

The case of multi-task is significantly more complicated than single task WSN nodes. Multi-task use-cases can consider WSN with a single sensor, but which de-correlates the sensing, processing and transmission tasks, or nodes which measures multiple physical values with different QoS requirements for each different sensing task. An example of such a use-case is the Super Citizen Smart Sensor (SCSS) project [221], where deployed WSN nodes are capable of measuring noise level, gas concentration for air quality, and temperature and humidity. All different sensing tasks can require a different periodicity: temperature and humidity tend to change slowly, while the noise level can vary very quickly. Moreover, the power consumption of each of these tasks is different. Thus, an efficient energy allocation algorithm is required.

In this work, a task τ_i is defined by the process it executes, its energy consumption $E_C^{\tau_i}$, its priority ρ_i, and its minimal and maximal number of executions between two transmissions, respectively denoted ϕ_i^{min} and ϕ_i^{max}, which enable to set the required QoS of the task. A high priority task has a high ρ_i. A task which execution is not always required can have ϕ_i^{min} set to 0. The transmission task is particular, as it is always executed at the end of a time slot. This can be modeled by setting both ϕ_i^{min} and ϕ_i^{max} to 1. This task model only adds QoS requirements and energy awareness to processes. Real-time capabilities can be integrated either by linking energy aware tasks to real-time processes, which embed information about their deadline and/or period, or by adding this information to the task model.

As the system is not composed of a single task anymore, the computation of E_B^{min} and E_B^{max} has to be adapted. E_B^{min} corresponds to the case where all tasks are executed the minimal number of times ϕ_i^{min} with a delay $D_{TX} = D_{TX}^{max}$ between two transmissions, and thus can be expressed as (5.5). Inversely, E_B^{max} corresponds to the case where each task τ_i is executed ϕ_i^{max} times with D_{TX}^{min} between two transmissions, as expressed in (5.6).

$$E_B^{min} = \frac{\mathcal{D} \times \sum_{i=1}^{K} (\phi_i^{min} \times E_C^{\tau_i})}{D_{TX}^{max}} \tag{5.5}$$

$$E_B^{max} = \frac{\mathcal{D} \times \sum_{i=1}^{K} (\phi_i^{max} \times E_C^{\tau_i})}{D_{TX}^{min}} \tag{5.6}$$

More than different QoS requirements, the different tasks can also have different types of use-cases. We denote three major types of use-cases which can be easily applied to most applications:

- *Reporting*: the task has a fixed QoS, as the goal is to measure a value with an accurate period between two measurements. For example in the previously presented use-case, a meteorological report based on temperature and humidity could be requested every two hours.

- *Alarm*: the task is triggered by an interrupt, which launches an optional processing task and a transmission to send an alarm. An example of such a task would be a microphone which detects an abnormal noise level, processes the samples to detect its source and transmit an alarm if a potential danger (e.g. a car crash noise) is detected.

- *QoS maximization*: the task is run as much as possible between the ϕ_i^{min} and ϕ_i^{max} limits. This is typically used to maximize the efficiency of a measurement campaign, for example to maximize the number of air quality measurements.

Although this work focuses on tasks which target QoS maximization, it is interesting to keep the possibility of adding other types of task in the energy allocation system, so that it can be extended for future use.

The strategy for data transmission can also differ, depending on the use-case. A measured data can be transmitted on the network directly after its measurement. Although this strategy reduces latency, defined as the delay between a measurement and its transmission, to a minimum, it also consumes more power, as a larger number of transmissions is attempted. Thus, this reduces the QoS for all tasks. If all tasks must be executed only once per transmission slot, the system can wake-up, execute all sensing and processing tasks, transmit the relevant data and go in sleep mode. In this case, the system is analog to a mono-task system, where a meta task composed of all sensing/processing tasks and the transmission task is periodically executed. Finally, the transmission task can send aggregated data from multiple sensing and processing tasks.

This last case is explored in the following subsections, with a QoS maximization target of all tasks. The goal of the energy allocation step is to maximize the consumed energy E_C^{Total}, given by (5.7) with the limit $E_C^{Total} \leq E_B$, where ϕ_i is the number of execution of a task between two transmissions, computed by the energy allocator, and K is the total number of tasks.

$$\begin{gathered} \max_{\phi_i} \left(E_C^{Total} = \sum_{i=1}^{K} (\phi_i \times E_C^{\tau_i}) \right) \\ s.t.\;\; E_C^{Total} \leq E_B \end{gathered} \tag{5.7}$$

5.3.2 Naive Energy Allocation Algorithm

A first approach is to find the set $\{\phi_1, \phi_2, ..., \phi_K\}$ which maximizes E_C^{Total}, expressed in (5.7), while still keeping $E_C^{Total} \leq E_B$. An algorithm could be used to search for all possible combinations of ϕ_i which respect $E_C^{Total} \leq E_B$, compute E_C^{Total} for all combinations and return the combination which maximizes E_C^{Total}. However, this algorithm would not take into account the priority of the different tasks nor the QoS requirements, and would require additional logic to check if the QoS and priority requirements are respected. Moreover, the complexity of this algorithm grows quickly with the number of tasks and the difference between E_B and the respective $E_C^{\tau_i}$, as a larger number of combinations becomes possible. This complexity requires both memory and computing resources, and is thus not fit for a WSN node.

Algorithm 5.2: Naive energy allocation algorithm.

```
input  : A list of the tasks to be scheduled, sorted by descending priority.
initialization;
for $i \leftarrow 1$ to $K$ do
    $\phi_i \leftarrow 0$
end
$E_i^{min} = min\,(E_C^{\tau_i})_{i=1}^{K}$;
while $E_B \geq E_i^{min}$ do
    for $i \leftarrow 1$ to $K$ do
        $\phi_i \leftarrow \phi_i + allocate\,(\tau_i, E_B)$;
        $E_B \leftarrow E_B - \phi_i \times E_C^{\tau_i}$;
    end
end
```

A simpler algorithm is to initialize all ϕ_i to 0, and to increment it for each task τ_i starting with the highest priority ρ_i. This algorithm can be run until there is not sufficient energy budget E_B to allocate energy to the task with minimal energy consumption, i.e.:

$$E_B - E_C^{Total} < min\,(E_C^{\tau_i})_{i=1}^{K} \tag{5.8}$$

Such an algorithm can be described as Algorithm 5.2. In this algorithm, the function $allocate\,(\tau_i, E_B)$ can be adapted to define different energy allocation policies. For example, ϕ_i can be incremented by 1, ϕ_i^{min} or ϕ_i^{max} at each loop to ensure that all tasks are executed. The increment could also be calculated as $E_B/E_C^{\tau_i}$, possibly capped at ϕ_i^{min} or ϕ_i^{max}, to minimize the number of loop iterations. It can be noted that this algorithm does not ensure that the minimal QoS is reached for all tasks before allocating energy to tasks for which $\phi_i \geq \phi_i^{min}$. To work-around this limitation, the algorithm can simply be executed multiple times with multiple stop conditions. The full pseudo-code of this method in shown in Algorithm 5.3. $\phi MinReached()$ and $\phi MaxReached()$ are functions which return $True$ if $\phi_i \geq \phi_i^{min}$ and $\phi_i \geq \phi_i^{max}$ $\forall i \in [1; K]$, respectively, and $False$ otherwise.

Algorithm 5.3: Naive energy allocation algorithm with QoS.

```
input : A list of the tasks to be scheduled, sorted by descending priority.
initialization;
$\phi_{reached}^{min} \leftarrow False$;
$\phi_{reached}^{max} \leftarrow False$;
for $i \leftarrow i$ to $K$ do
    $\phi_i \leftarrow 0$
end
$E_i^{min} = min\,(E_C^{\tau_i})_{i=1}^{K}$;
while ($E_B \geq E_i^{min}$ $OR$ $\phi_{reached}^{min} = False$) do
    for $i \leftarrow 1$ to $K$ do
        $\phi_i \leftarrow \phi_i + allocate\,(\tau_i, E_B)$;
        $E_B \leftarrow E_B - allocate\,(\tau_i, E_B) \times E_C^{\tau_i}$;
    end
    $\phi_{reached}^{min} \leftarrow \phi MinReached\,(TaskSet)$
end
while ($E_B \geq E_i^{min}$ $OR$ $\phi_{reached}^{max} = False$) do
    for $i \leftarrow 1$ to $K$ do
        $\phi_i \leftarrow \phi_i + allocate\,(\tau_i, E_B)$;
        $E_B \leftarrow E_B - allocate\,(\tau_i, E_B) \times E_C^{\tau_i}$;
    end
    $\phi_{reached}^{max} \leftarrow \phi MaxReached\,(TaskSet)$
end
while $E_B \geq E_i^{min}$ do
    for $i \leftarrow 1$ to $K$ do
        $\phi_i \leftarrow \phi_i + allocate\,(\tau_i, E_B)$;
        $E_B \leftarrow E_B - allocate\,(\tau_i, E_B) \times E_C^{\tau_i}$;
    end
end
```

Once the number of execution ϕ_i has been computed for each task, the delay between each task execution can be executed as $D_{\tau_i} = D_{TX}/\phi_i$, where D_{TX} is the delay between two transmissions and the length of the current time slot. A single timer set to the Greatest Common Divisor (GCD) can be used to schedule all tasks, which eases the resource requirements for constrained platforms.

This algorithm is much more efficient than searching for all possible ϕ_i combinations, but its complexity can still grow quickly if the number of tasks or E_B is high. Moreover, this solution does not accurately take into account the priority ρ_i of the tasks. Thus, a lower priority task with a consumption E_0 lower than a higher priority task can be executed more times. For example, if we take two tasks τ_0 and τ_1 where $E_C^{\tau_0} > \chi \times E_C^{\tau_1}$ and $\rho_0 > \rho_1$, we can find E_B values in $]E_C^{\tau_0}; 2 \times E_C^{\tau_0}[$ for which $E_B - E_C^{\tau_0} > E_C^{\tau_1}$, possibly leading to multiple executions of the lower priority task. Additionally, as all ϕ_i computed by this method are integer, E_C^{Total} is not always equal to E_B. Indeed, up to E_i^{min} can be unallocated as there is not enough energy to be allocated, even for the task with the smallest consumption. Thus, although it provides a quick practical solution to the energy allocation problem for multi-task WSN nodes, this algorithmic solution is suboptimal.

5.3.3 Optimal Energy Allocation for Multi-task IoT Node

As a fully algorithmic energy allocator is suboptimal, there is a need to determine an optimal solution to this problem. To define this problem, we suppose the following hypothesis:

- All priorities ρ_i are normalized so that $\sum_{i=1}^{K} \rho_i = 1$
- Data is transmitted only once, at the end of each time slot

The allocation problem solution must be able to provide equity between the different tasks, to avoid a task with a much lower consumption $E_C^{\tau_i}$ or much higher priority ρ_i being the only one executed. One way to solve this problem while keeping fairness between tasks is to solve the cost function as a sum of logarithms [222], as defined in (5.9), where $\overrightarrow{x}$ is a solution vector $\{\phi_1, \phi_2, ..., \phi_K\}$:

$$f_0(\overrightarrow{x}) = -\sum_{i=1}^{K} (\rho_i \times ln\,(\phi_i)) \tag{5.9}$$

With the constraint defined by the function (5.10), where E_C^{TX} is the energy consumption of the transmission task:

$$\sum_{i=1}^{K} (\phi_i \times E_C^{\tau_i}) \leq E_B - E_C^{TX}. \tag{5.10}$$

Thus we can define the constraint function $g_i\,(\overrightarrow{x})$ as (5.11):

$$g_i(\overrightarrow{x}) = \Big[\sum_{i}^{K} (\phi_i \times E_C^{\tau_i})\Big] - (E_B - E_C^{TX}) \leq 0. \tag{5.11}$$

This is a linear optimization of a convex function problem, thus a local solution is also a global solution. The Lagrangian function for the problem can be expressed as (5.12):

$$L\left(\overrightarrow{x}, \overrightarrow{\lambda}\right) = f_0(\overrightarrow{x}) + \sum_{i=1}^{K} (\lambda_i \times g_i(\overrightarrow{x})) \tag{5.12}$$

λ_i are the Lagrange multipliers. We resolve this optimization problem using the KKT conditions [219]. This theorem states that for each point x which maximizes the $f_0(\overrightarrow{x})$ function, there exists a set of these multipliers that satisfies the following conditions and is an optimal solution:

- x maximizes $L\left(\overrightarrow{x}, \overrightarrow{\lambda}\right) \forall x$
- $\lambda_i \geq 0$, at least one λ is > 0
- $\lambda_i \times g_i(\overrightarrow{x}) = 0$

Thus, if we can find a local solution which satisfies the KKT conditions, this solution is proven to globally maximize the function, and is the optimal allocated energy value. To determine this local solution, we pose $i = K$. (5.12) becomes:

$$L\left(\overrightarrow{x}, \overrightarrow{\lambda}\right) = \rho_K \times ln(\phi_K) + \lambda_K \times \phi_K \times E_K - (E_B - E_C^{TX}) \tag{5.13}$$

This function of ϕ_K presents a local maximum when its derivative with respect to ϕ_K is equal to 0, expressed as (5.14):

$$\frac{\mathrm{d}L\left(\overrightarrow{x}, \overrightarrow{\lambda}\right)}{\mathrm{d}\phi_K} = \frac{-\rho_K}{\phi_K} + \lambda_K \times E_K = 0 \Leftrightarrow \phi_K = \frac{\rho_K}{\lambda_K \times E_K} \tag{5.14}$$

This gives us a local solution for which $\lambda_K > 0$, as $\phi_K \geq 0$, and thus respects the first and second KKT conditions. If we check the third KKT condition for $\lambda_i = \lambda_K \geq 0$, we have (5.15):

$$\lambda_K \times \sum_{i=1}^{K} (\phi_i \times E_C^{\tau_i}) - (E_B - E_C^{TX}) = 0 \Rightarrow \begin{cases} \lambda_K = 0 \\ \sum_{i=1}^{K} (\phi_i \times E_C^{\tau_i}) = (E_B - E_C^{TX}) \end{cases} \tag{5.15}$$

λ_K can not be equal to 0, otherwise ϕ_K would not exist. By integrating the local solution in (5.15), we obtain (5.16):

$$\lambda_K \times g_i(\overrightarrow{x}) = 0 \Leftrightarrow \sum_{i=1}^{K} \left(\frac{\rho_i}{\lambda_K \times E_C^{\tau_i}} \times E_C^{\tau_i} \right) = (E_B - E_C^{TX}) \Leftrightarrow \frac{1}{\lambda_K} \sum_{i=1}^{K} \rho_i = (E_B - E_C^{TX}) \tag{5.16}$$

As ρ_i values are normalized so that their sum is equal to 1, (5.16) can be simplified as:

$$\frac{1}{\lambda_K} = (E_B - E_C^{TX}) \Leftrightarrow \lambda_K = \frac{1}{E_B - E_C^{TX}} \tag{5.17}$$

Thus, if $E_B > E_C^{TX}$, (5.17) proves that we have a set of λ_i defined as $\lambda_i = \lambda_K = 1/(E_B - E_C^{TX}) > 0$, a solution which fulfills both the second and third condition, while the first condition is fulfilled by the fact that the function is convex. So, the solution $\phi_i = \rho_i/(\lambda_K \times E_C^{\tau_i})$ maximizes $f_0(\overrightarrow{x})$, provided that E_B enables at least one transmission. If $E_B < E_C^{TX}$, the problem does not apply, as the WSN node does not have enough energy to perform any action.

Hence, the optimal ϕ_i for each task can be computed as (5.18):

$$\phi_i = \frac{\rho_i \times \left(E_B - E_C^{TX}\right)}{E_C^{\tau_i}} \tag{5.18}$$

Although this result is simple to compute, the mathematical process behind its demonstration ensures its optimality. It can be noticed that the result of this optimal formula is not always an integer. It is possible to truncate each ϕ_i to obtain integer values which respect the constraint $E_C^{Total} \leq E_B - E_C^{TX}$. This enables computing the delays D_{τ_i} of each task and schedule them as with the previous algorithmic method. However, that breaks the optimality, as there will still be a difference between E_C^{Total} and $E_B - E_C^{TX}$.

Another possibility is to compute $D_{\tau_i} = D_{TX}/\phi_i$ while keeping the exact value of ϕ_i, even if it is not an integer. As it becomes much complicated to compute the GCD of the different ϕ_i, each task must be scheduled with its own timer. Although this increases computing requirements, most embedded OSs provide software timer libraries which simplify this implementation, and avoid requiring a large number of hardware timers. Each timer can be set to the new D_{τ_i} value when it is computed. As the tasks are run asynchronously from the transmission task and the time slot, the data to be transmitted has to be stored in a properly sized buffer. In the case of long range transmission, such as LoRaWAN and Sigfox networks, where only a few tens of bytes can be transmitted at once, the QoS of the tasks must be set carefully to avoid overflowing this transmission buffer and losing data.

Although this solution is optimal in regard to the priority ρ_i and energy consumption $E_C^{\tau_i}$ of the tasks, it does not take into account the minimal and maximal QoS of the tasks. Moreover, it does not compute the delay between two consecutive transmissions, as it is supposed that each time slot ends with a transmission. Thus, these computations have to be included in an algorithm which enforces the required QoS and computes an adapted D_{TX}.

Algorithm 5.4 proposes the integration of the optimal ϕ_i. It defines E_C^{max} as the maximum energy that can be consumed, i.e. the E_C^{Total} when all ϕ_i are equal to ϕ_i^{max} and $D_{TX} = D_{TX}^{min}$. If $E_B > E_C^{max}$, it enforces the maximal QoS rules, which lets the node save more energy for later usage. If $E_B < E_C^{max}$, it tries to maximize the QoS by allocating energy using the optimal formula in (5.18). This example favors the battery life over the QoS, as the minimal QoS settings are not enforced if not enough energy is available. On the other hand, it tries to maximize QoS as much as possible, ensuring that high priority tasks will be favored. It can be noticed that the QoS limits do not have to be aligned with the priority setting. For example, a high priority task may require only one execution per timeslot, while a less important sensing task may be executed multiple times. Due to this, the calculated ϕ_i has to be capped at ϕ_i^{max}. Although this breaks the optimality of the ϕ_i computation for the other tasks, it avoids spending energy on an unrequired task execution. The delay D_{τ_i} between two consecutive τ_i executions is then computed as the duration of the time slot, equal to the delay D_{TX} between two transmissions, divided by its optimal number of execution ϕ_i.

Algorithm 5.4: Optimal ϕ_i calculation integrated in practical solution.

```
input : A list of the tasks to be scheduled, sorted by descending priority.
initialization;
if $E_B > E_C^{max}$ then
    $E_B^{\%} \leftarrow \frac{E_B - E_C^{max}}{E_B^{max} - E_C^{max}}$;
    $D_{TX} \leftarrow D_{TX}^{min} + (D_{TX}^{max} - D_{TX}^{min}) \times \left(1 - E_B^{\%}\right)$;
    for $i \leftarrow 1$ to $K$ do
        $\phi_i \leftarrow \phi_i^{max}$ ;
        $D_{\tau_i} \leftarrow \frac{D_{TX}}{\phi_i}$;
    end
else
    $D_{TX} \leftarrow D_{TX}^{max}$;
    for $i \leftarrow 1$ to $K$ do
        $\phi_i \leftarrow max\left(\phi_i^{max}, \frac{\rho_i \times \left(E_B - E_C^{TX}\right)}{E_C^{\tau_i}}\right)$;
        $D_{\tau_i} \leftarrow \frac{D_{TX}}{\phi_i}$;
    end
end
```

5.3.4 Efficiency of Multi-task energy Allocation Methods

The algorithm presented in Algorithm. 5.4 has been implemented and evaluated on real world IoT nodes. The nodes firmware is based on Contiki, which provide lightweight software timer libraries. Thus, an asynchronous task execution scheme, where all tasks have their own timer, can be implemented with a low overhead. The delay for each timer is computed using the previous algorithm.

Sensing task	Energy consumption (J)	Minimal number of executions	Maximal number of execution
Temperature/ Humidity	0.098	1	2
Noise	0.209	0	4
Gas (CO^2)	0.172	1	3

Table 5.3: Characteristics of the tasks executed by the energy harvesting nodes. All priorities are set to the same value.

This multi-source energy allocation scheme has been implemented on two sensor nodes. Each node is equipped with a temperature/humidity sensor, a CO^2 sensor and a noise sensor. The power consumption characteristics and minimal/maximal QoS settings for each sensor is described in Table 5.3. In this experiment, all priorities are set to the same value. It is clear that a higher priority will lead to more task

Energy harvesting node		Multi-sources	Single source
V_{BAT} (V)	Mean	4.061	4.058
	Std. deviation	0.061	0.127
E_B (J)	Mean	5.682	5.064
	Std. deviation	1.095	2.123
D_{TX} (min)	Mean	31.99	48.38
	Std. deviation	34.32	61.08
Messages transmitted		672	446
Failed transmission		0	0

Table 5.4: Comparison of multi and single source energy harvesting systems.

executions. Setting an equal value for all tasks priority avoids this and highlights the effect of QoS requirements on the energy allocation.

The first sensor is powered by a 2 W solar panel, while the second sensor uses a multi-source energy harvesting board to combine two solar panels capable of up to 2 W and 3.5 W. Additionally, a voltage generator is used to simulate a thermoelectric generator with an open voltage of 800 mV and a short maximum current of 4 mA. As the MPPT uses this source at $V_{MPP} = 0.5 \times V_{OC}$, and the boost converter efficiency is about 40 %, this simulated source delivers a continuous power $0.8 \times 0.5 \times 0.004 \times 0.4 = 0.64$ mW, which is small enough to let the 7.5 F supercapacitor partially discharge. This experiment also validates the approach of multi-source energy harvesting to increase the QoS of a WSN node.

The experiment has been run for a total of 15 days. The simulated thermoelectric generator was cut on the second day due to a power failure. However, this enabled testing the efficiency of multi-source energy harvesting with similar types of energy sources, i.e. two solar panels. The simulated thermoelectric generator was reset at the end of the fifth day. During the campaign, the energy storage voltage V_{BAT}, the energy budget E_B, the delay between two transmissions D_{TX} and the number of successful transmissions are recorded. These measurements are shown in Fig. 5.2.

The measurements show that, as expected, the use of multiple energy harvesting sources has a positive impact on the QoS of the node. In particular, Table 5.4 show that the mean E_B is sensibly higher and the mean D_{TX} is lower when more energy harvesting sources are used. Due to this, the node with multiple sources is able to transmit 672 packets, compared to 446 packets when the node is powered by a single energy source. In both cases, no transmission has failed due to a low SOC.

It can be noticed that the type of energy sources used has a direct impact on the efficiency of the multi source energy harvesting system. From the second to the fifth day, the first node was powered by two solar panels, providing respectively 2 and 3.5 W, while the second node was powered only by a 2 W solar panel. During this period, the multi source energy harvesting node was only marginally more efficient than the single source energy harvesting node, with 141 transmitted packets against 133.

Indeed, the 2 W solar panel is sufficient to completely recharge the energy storage and maximize its QoS at the start of a new day. Thus, the additional solar panel only helps the node recharge its storage faster and its additional energy is not required during the day. On the contrary, when a thermoelectric generator is simulated during

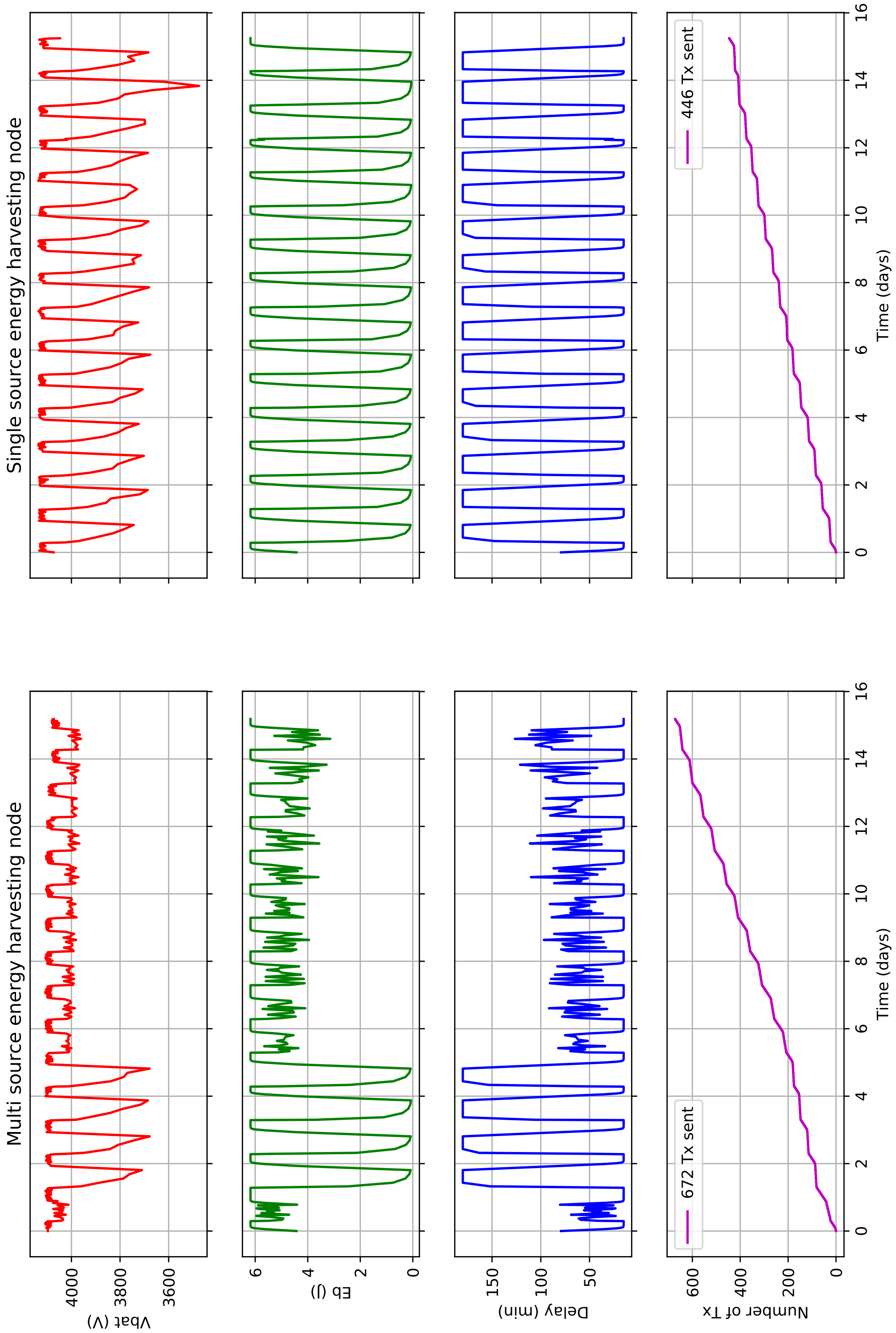


Figure 5.2: Comparison of multi and single source energy harvesting systems.

day and night, its energy enables the node to sensibly increase the QoS. In this case, the simulated energy source complements the solar panel, bringing energy when the solar panels do not harvest energy. This shows that the use of a second similar source has little interest in some use cases, while a complementary energy source would have a higher impact on the QoS of the node.

Additionally, the number of executions per time slot is counted for each task. Thus, the mean and total number of executions can be computed for each task. These results are presented in Table 5.5, while Fig. 5.3 shows the distribution of the measurements over time.

These results confirm that the addition of a second solar panel has much less effect on the node QoS than the use of the simulated TEG. Indeed, the simulated TEG is complementary to the solar panels, as it provides power at night, when the solar panels does not. This can be seen in the repartition of task execution, which is similar between the two nodes when the simulated thermoelectric generator is powered off. When it is powered on, however, the multi source energy harvesting node has enough energy to maximize the QoS of each task and reduce the delay D_{TX} between transmissions, while the other node still has to modulate the number of executions of each task to avoid depleting its battery. This shows that our energy allocation system works as expected, and enabled the multi source energy harvesting node to perform significantly more tasks than the single source one.

In some time slots, the number of executions of some tasks is higher than the specified maximum. This is due to the fact that each task manages its own timer. In some situations where multiple timers expire at a similar time, a task execution is already scheduled before its new number of executions is calculated. As the scheduler does not check if the task is already in the execution queue, it does not account for this task execution, which can lead to an additional task execution. A bug also happened around day 6 and 14, when the single source energy harvesting node respectively executed 12 and 31 noise sensing tasks, far more than specified. However, these bugs are linked to the scheduler and its implementation and do not prevent the proper application of the energy allocation policy.

Task executions		Multi sources	Single source
Temperature/ Humidity	Mean per time slot	1.978	1.871
	Std. deviation	0.199	0.482
	Total number of executions	1331	842
Noise	Mean per time slot	3.906	3.596
	Std. deviation	0.714	1.894
	Total number of executions	2629	1618
Gas	Mean per time slot	2.945	2.700
	Std. deviation	0.570	0.912
	Total number of executions	1982	1215

Table 5.5: Number of task executions for multi and single source energy harvesting systems.

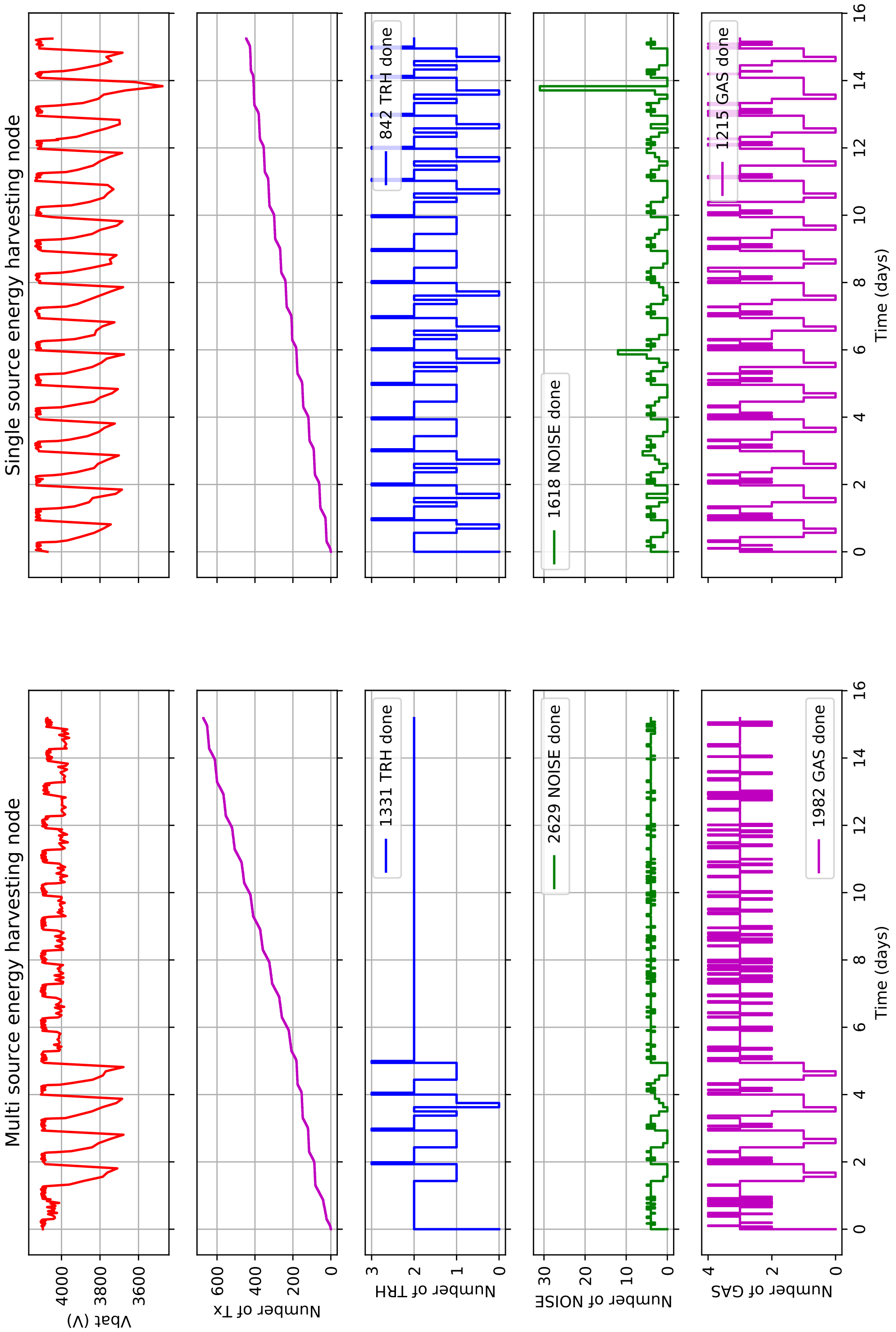


Figure 5.3: Number of task executions for multi and single source energy harvesting systems.

In this case, the use of additional energy sources enable the multi source energy harvesting node to keep executing its tasks at their maximum QoS. Due to this, the number of task execution per time slot is much more stable, as seen with the lower standard deviation of this value in Table 5.5. Thus, the use of additional energy sources has the unexpected effect of enabling a more predictable QoS.

5.4 Conclusions

In this chapter, we explored the problem of energy allocation for single and multi task energy harvesting systems. We showed that the state-of-the-art solutions focus on real time systems and thus are not adapted to our use-case. In particular, there is no existing solution to optimally divide an energy budget between multiple different tasks. We first detail how an energy budget can be allocated for single task WSN node, and demonstrate how data aggregation can improve the QoS of single task WSN node. The case of multi-task systems is then presented. An optimal energy allocation policy is mathematically derived, and is then adapted to fit a real world implementation. Experiments show the proper operation of our energy allocation system. Furthermore, this experiment also demonstrates that multi-source energy harvesting is useful to improve the QoS of the WSN node.

Conclusion & Perspectives

Conclusion

In the coming years, it is expected that billions of connected devices will be deployed, for applications ranging from industrial to residential contexts. A large part of these devices will be in charge of measurements, periodically measuring and transmitting physical values. Due to their vast variety of use cases, powering these billions of devices is a challenge. In particular, traditional solutions such as primary batteries do not scale well, as they would generate a huge quantity of chemical waste. Moreover, the cost of maintaining the network, by recharging and/or changing these batteries would be high, especially for use-cases where the nodes are difficult to access, and would impact the financial viability of some IoT deployments. In order to circumvent this problem, the use of energy harvesting technologies is considered, to let the device recharge itself. However, this solution also has its challenges, and requires the node to manage his own energy locally and efficiently.

In this thesis, energy harvesting capabilities were added to an already existing IoT platform which communicates using LoRa technology. In particular, we proposed a new energy harvesting architecture, aimed at lowering the implementation cost of such a solution. We compared this architecture with state of the art solutions, and showed that it reduces cost when a large number of sources are used but is less efficient when the sources level are too different. Based on this study, we designed a multi-source energy harvesting platform for Wi6labs sensor platform.

Although the produced board enabled the WSN node to efficiently harvest energy, it still required additional intelligence to efficiently manage its energy. A power management module was implemented in the firmware to fill this task. In this thesis, the power manager is divided into two sub-modules. The Energy Budget Estimator (EBE) sub-module is tasked with estimating the energy capabilities of the platform, and accordingly compute an energy budget which can be spent over a period of time. An Energy Allocator (EA) sub-module then takes this energy budget as input and adapts the behavior of the node to match the energy consumption. Thus, the WSN node was able to adapt its Quality of Service (QoS) to match his energy capabilities, enabling theoretically perpetual operation.

This thesis presents a comparison of state of the art EBE algorithms, both in simulation and implemented on Wi6labs platform. To our knowledge, this is the first implementation of such algorithms on a commercial WSN node platform. Moreover, this implementation adapts the execution of these algorithms to the specificities of long range radio networks, such as LoRaWAN, which include a high instantaneous power consumption and a large delay between transmissions. Our experiments showed that the choice of EBE algorithm has little impact over the performance of the WSN node, once all algorithms have been tuned to maximize performance. Thus, this makes it possible to choose algorithms which are much simpler to implement and/or to tune.

The problem of energy allocation for energy harvesting WSN nodes was then explored. Most state of the art solutions tackle this problem through the prism of task scheduling, often in real-time context, and do not consider variable QoS. We first focused on energy allocation for single task WSN nodes, showing that this problem is equivalent to finding a suitable function to transform an energy budget into a delay. Two different policies were implemented, which showed the impact that this step has on the global performances of the device. Moreover, we demonstrated that successive measurements can be aggregated to minimize the number of transmissions and increase the number of transmitted measures, and the implementation challenges this solution presents. In this type of system, the use-case drives the compromises to be done between performances, battery depletion avoidance and implementation cost, and thus the match between different possible solutions and requirements.

Finally, this thesis extends the precedent studies to multi-task WSN nodes, i.e. systems which have to perform multiple types of measurements with different QoS requirements. A fully algorithmic solution which solves this problem was presented, along with the difficulties of implementing it on a resource constrained WSN node. An optimal solution to the problem was then mathematically derived, and was integrated in a practical application. This solution was implemented and tested to show how it increases the QoS of a WSN node when more energy is available to harvest.

To conclude, this thesis outlines the different challenges of designing and implementing efficient energy harvesting solutions on real-world WSN platforms, and proposes solutions which tackle these challenges. The various requirements of IoT applications in terms of energy and applications make it complicated to design a generic solution which can be adopted as-is for all use-cases. In the end, the adopted solution should be tuned and adapted to fit the application requirements.

Perspectives

Although energy harvesting for WSN nodes has been an active research field during the recent years, multiple problems still have to be addressed in the future.

5.4.1 Multi-source energy harvesting integration

Though the academic world has proposed multiple multi-source energy harvesting circuits, very few industrial implementations have followed. Moreover, these systems are often a compromise between energy harvesting efficiency and genericity, where the most efficient systems are tightly coupled to a specific type of energy source. Though only a few passive components have to be modified to adapt our solution, it is not completely plug and play. Thus, there is an interest in developing circuits which can take multiple undetermined sources as input. Moreover, in order to lower the implementation cost of such solutions, there is a need for commercial off-the-shelf ICs which implement this type of solution. As of the publication of this document, a first commercial implementation has recently been announced [223] but not released.

5.4.2 Energy source detection in a multi-source context

As multi-source energy harvesting adds multiple heterogeneous sources, the EBE algorithm has to be model-free, and can not depend on the model of a particular source. However, model-based EBE algorithms can be very efficient when sources are predictable, such as a solar panel in an outdoor context. It could be interesting to be able to detect which energy sources are connected in a multi-source energy harvesting system, in order to use the most efficient EBE algorithm for each energy source.

5.4.3 Self-learning EBE algorithm

The performance of EBE algorithm can be drastically modified with different settings. Thus, proper tuning of the algorithm is essential to maximize the performance of the node. Using solutions which learn and adapt their settings over the time could greatly simplify the implementation of such solutions. RLman [172] is an example of such a solution, but only considers a single-task system, and can be hard to implement for multi-task systems. Thus, there is a need for a similar solution which decouples the EBE and EA steps.

5.4.4 Performance based QoS

In this work, the EA varies the QoS of a task on a WSN node by varying its frequency, and assumes the task is always executed with the same performance. However, the energy consumption of some tasks can be varied by adapting their performance, e.g. decreasing the energy consumption of a sensing task by reducing its measurement precision or adapting the transmission parameters. For example, [224] details how the LoRa modulation settings (i.e. bandwidth, spreading factor...) affect both the data throughput and energy consumption of the transmitter. Integrating this knowledge in the EA would enable in to take into account the network conditions and accordingly adapt its energy allocation policy.

5.4.5 Design tool for energy harvesting WSN node with power management

The usual approach to an energy harvesting WSN node design is to design the sensor for a type of energy source and storage, and the nergy source is only specified in the end to ensure it meets the application requirements. Often, the selected energy source is bigger than required to ensure that the system will always have a sufficient energy reserve. It could be interesting to develop tools which use the power manager characteristics to determine how much energy should be harvested to fulfill given QoS requirements under expected environmental conditions. Such a tool could be use to help properly size the different elements of the design.

Acronyms

ADC Analog to Digital Converter.

ALAP As Late As Possible.

ASAP As Soon As Possible.

CPU Central Processing Unit.

DAC Digital-to-Analog Converter.

DMS Deadline Monotonic Scheduling.

DVFS Dynamic Voltage and Frequency Scaling.

EA Energy Allocator.

EBE Energy Budget Estimator.

EDF Earliest Deadline First.

EEPROM Electrically Erasable Programmable Read-Only Memory.

EIRP Equivalent Isotropic Radiated Power.

ENO Energy Neutral Operation.

ERP Effective Radiated Power.

EWMA Exponentially Weighted Moving Average.

FPGA Field Programmable Gate Array.

IC Integrated Component.

IoT Internet of Things.

KKT Karush-Kuhn-Tucker.

LDO Low DropOut.

Li-Ion Lithium-ion.

LiPo Lithium-polymer.

LoRa Long Range.

LoRaWAN Long Range Wide Area Network.

MAC Media Access Control.

MCU Micro-Controller Unit.

MPP Maximum Power Point.

MPPT Maximum Power Point Tracking.

MTBF Mean Time Between Failures.

MTTF Mean Time To Failure.

NiMH Nickel-Metal Hybrid.

NMOS N-type Metal-Oxide-Semiconductor.

OS Operating System.

P&O Perturb & Observe.

PCB Printed Circuit Board.

PID Proportional-Integral-Derivative.

PMIC Power Management Integrated Component.

PMOS P-type Metal-Oxide-Semiconductor.

PS Primary Storage.

QoS Quality of Service.

RAM Random Access Memory.

RF Radio Frequency.

RMS Rate Monotonic Scheduling.

ROM Read-Only Memory.

RTOS Real-Time Operating System.

SMPS Switched Mode Power Supply.

SOC State of Charge.

SS Secondary Storage.

TEG ThermoElectric Generator.

UART Universal Asynchronous Receiver-Transmitter.

WSN Wireless Sensor Network.

WUR Wake-Up Radio.

Nomenclature

Common

V_{BAT}	Energy storage voltage, in Volt (V). V_{BAT}^{min} and V_{BAT}^{max} indicate the values at which the energy storage is considered empty and full, respectively.
E_R	Residual energy left in storage, in Joule (J). E_R^{empty} and E_R^{full} indicate the values at which the energy storage is considered empty and full, respectively.
ΔE_R	Residual energy variation between two consecutive time slots, in Joule (J).
E_H	Harvested energy, in Joule (J).
E_B	Energy budget for the current time slot, in Joule (J). E_B^{min} and E_B^{max} are the minimal and maximal E_B values that can be delivered, respectively.
E_C	Consumed energy, in Joule (J).
D_{TX}	Delay between two consecutive transmissions, in minutes (min). D_{TX}^{min} and D_{TX}^{max} are the minimal and maximal values, respectively.
V_{MPP}	Voltage of the MPP, in Volt (V).
Tx	Number of transmissions done.
Tx_{fail}	Number of failed transmissions.
$\bar{x}$	Mean value of x.
σ	Standard deviation.

Energy Harvesting Hardware

V_{IN}	Input voltage of a power converter or switch, in Volt (V).
V_{OUT}	Output voltage of a power converter or switch, in Volt (V).
$\mathcal{C}$	Capacity of a lithium based battery, in milliAmpere Hours (mAh).
C	Capacitance of a capacitor or supercapacitor, in Farads (F).
λ_b	Base failure rate of a component per 10^6 hours.
λ_p	Failure rate of a part per 10^6 hours.

π_i	Stress factor induced by parameter i.
L	Lifetime of a component, in hours.
V_{MPP}^{set}	Manually set V_{MPP}, in Volt (V).
V_{OC}	Open circuit voltage of an energy source, in Volt (V).
K_{focv}	Coefficient $\in [0;1]$ of V_{OC} at which the MPP is located.
N	Number of energy sources.

Energy Management

$\mathcal{D}$	Time base for EBE and EA computations. Set to 60 minutes.
E_{thr}^{full}	Fuzzy value which indicates how full the energy storage is. Used by Fuzzyman.
E_{thr}^{empty}	Fuzzy value which indicates how empty the energy storage is. Used by Fuzzyman.
ΔE_{thr}^{pos}	Fuzzy value which indicates if the energy storage is charging. Used by Fuzzyman.
ΔE_{thr}^{neg}	Fuzzy value which indicates if the energy storage is discharging. Used by Fuzzyman.
E_{TRGT}	Target E_R. Used by LQ-Tracker.
μ	Step size. Used by LQ-Tracker.
Φ	Feature vector. Used by LQ-Tracker.
Θ	Parameter vector. Used by LQ-Tracker.
u_t	Control output of LQ-Tracker.
α	Control output smoothing coefficient. Used by LQ-Tracker.
β	Long/short term filter coefficient. Used by LQ-Tracker.
γ	Low-pass filter coefficient. Used by all EBE.
E_C^{mono}	Energy consumption of the task, for a single task WSN node, in Joule (J).
K	Total number of tasks.
τ_i	Task i.
ρ_i	Priority of τ_i.
ϕ_i	Number of executions of τ_i. ϕ_i^{min} and ϕ_i^{max} indicate the minimal and maximal number of times τ_i should be executed.
D_{τ_i}	Delay between two consecutive τ_i executions, in minutes (min).

$E_C^{\tau_i}$	Energy consumption of the task τ_i.
$min(E_C^{\tau_i})$	Minimum $E_C^{\tau_i}$ in the task set of a WSN node.
E_C^{Total}	Energy consumption of the node, i.e. energy consumed when all tasks are executed ϕ_i times, in Joule (J).
E_C^{TX}	Energy consumption of the transmission task, in Joule (J).
E_C^{max}	Maximum energy consumption of the node, i.e. energy consumed when all tasks are executed ϕ_i^{max} times, in Joule (J).
$\overrightarrow{x}$	Solution vector of the EA optimization problem.
λ_i	Lagrange multipliers.
m	Lagrangian solution space size.

Bibliography

[1] R. Vullers, R. Schaijk, H. Visser, J. Penders, and C. Hoof, “Energy Harvesting for Autonomous Wireless Sensor Networks,” *IEEE Solid-State Circuits Magazine*, vol. 2, no. 2, pp. 29–38, 2010.

[2] Gartner Inc., “Gartner says 8.4 billion connected "things" will be in use in 2017, up 31 percent from 2016,” Feb. 2017. Accessed: Oct. 18, 2018.

[3] A. Golberg, H. D. Rabinowitch, and B. Rubinsky, “Zn/Cu-vegetative batteries, bioelectrical characterizations, and primary cost analyses,” *AIP Journal of Renewable and Sustainable Energy*, vol. 2, p. 033103, May. 2010.

[4] L. Catacuzzeno, F. Orfei, A. Di Michele, L. Sforna, F. Franciolini, L. Gammaitoni, “Energy harvesting from a bio cell.” `https://arxiv.org/pdf/1808.07000.pdf`. Accessed: 2018-10-29.

[5] M. Younis, M. Youssef, and K. Arisha, “Energy-aware routing in cluster-based sensor networks,” in *IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunications Systems*, pp. 129–136, 2002.

[6] T. N. Le, A. Pegatoquet, O. Berder, and O. Sentieys, “Energy-Efficient Power Manager and MAC Protocol for Multi-Hop Wireless Sensor Networks Powered by Periodic Energy Harvesting Sources,” *IEEE Sensors Journal*, vol. 15, pp. 7208–7220, Aug. 2015.

[7] P. Lee, M. Han, H.-P. Tan, and A. Valera, “An empirical study of harvesting-aware duty cycling in environmentally-powered wireless sensor networks,” in *IEEE International Conference on Communication Systems*, pp. 306–310, Nov. 2010.

[8] P.-D. Gleonec, J. Ardouin, M. Gautier, and O. Berder, “Architecture exploration of multi-source energy harvester for IoT nodes,” in *IEEE Online Conference on Green Communications (OnlineGreenComm)*, pp. 27–32, Nov. 2016.

[9] P.-D. Gleonec, J. Ardouin, M. Gautier, and O. Berder, “A Real-World Evaluation of Energy Budget Estimation Algorithms for Autonomous Long Range IoT Nodes,” in *IEEE International Conference on Telecommunications (ICT)*, pp. 561–565, Jun. 2018.

[10] National Renewable Energy Laboratory (NREL), “National Center for Photovoltaics (NCPV).” `https://www.nrel.gov/pv/national-center-for-photovoltaics.html`. Accessed: 2018-08-06.

[11] V. Raghunathan, A. Kansal, J. Hsu, J. Friedman, and M. Srivastava, "Design considerations for solar energy harvesting wireless embedded systems," in *IEEE International Symposium on Information Processing in Sensor Networks (ISPN)*, pp. 457–462, Apr. 2005.

[12] F. Simjee and P. Chou, "Efficient Charging of Supercapacitors for Extended Lifetime of Wireless Sensor Nodes," *IEEE Transactions on Power Electronics*, vol. 23, pp. 1526–1536, May. 2008.

[13] S. Kim and P. H. Chou, "Energy harvesting by sweeping voltage-escalated charging of a reconfigurable supercapacitor array," in *IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)*, pp. 235–240, IEEE, Aug. 2011.

[14] M. A. Green, Y. Hishikawa, W. Warta, E. D. Dunlop, D. H. Levi, J. Hohl-Ebinger, and A. W. Ho-Baillie, "Solar cell efficiency tables (version 50)," *Wiley Journal of Progress in Photovoltaics: Research and Applications*, vol. 25, pp. 668–676, Jul. 2017.

[15] J. F. Randall and J. Jacot, "The performance and modelling of 8 Photovoltaic materials under variable light intensity and spectra," in *World Renewable Energy Congress VII & Expo*, 2002.

[16] D. E. Carlson and C. R. Wronski, "Amorphous silicon solar cell," *AIP Applied Physics Letters*, vol. 28, pp. 671–673, Jun. 1976.

[17] M. A. Green, "Thin-film solar cells: review of materials, technologies and commercial status," *Springer Journal of Materials Science: Materials in Electronics*, vol. 18, pp. 15–19, Oct. 2007.

[18] R. Kamada, T. Yagioka, S. Adachi, A. Handa, K. F. Tai, T. Kato, and H. Sugimoto, "New world record Cu(In,Ga)(Se,S)2 thin film solar cell efficiency beyond 22%," in *IEEE Photovoltaic Specialist Conference (PVSC)*, pp. 1–5, Jun. 2017.

[19] infinityPV ApS, "InfinityPV - Organic solar cells." `https://infinitypv.com/technology/opv`. Accessed: 2018-08-06.

[20] X. Che, Y. Li, Y. Qu, and S. R. Forrest, "High fabrication yield organic tandem photovoltaics combining vacuum- and solution-processed subcells with 15% efficiency," *Nature Energy*, vol. 3, pp. 422–427, May. 2018.

[21] N. J. Guilar, T. J. Kleeburg, A. Chen, D. R. Yankelevich, and R. Amirtharajah, "Integrated solar energy harvesting and storage," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 17, no. 5, pp. 627–637, 2009.

[22] W. Rieutort-Louis, L. Huang, Y. Hu, J. Sanz-Robinson, S. Wagner, J. C. Sturm, and N. Verma, "A Complete Fully Thin-Film PV Harvesting and Power-Management System on Plastic With On-Sheet Battery Management and Wireless Power Delivery to Off-sheet Loads," *IEEE Journal of Photovoltaics*, vol. 4, pp. 432–439, Jan. 2014.

[23] S. Dalola, M. Ferrari, V. Ferrari, M. Guizzetti, D. Marioli, and A. Taroni, “Characterization of Thermoelectric Modules for Powering Autonomous Sensors,” *IEEE Transactions on Instrumentation and Measurement*, vol. 58, pp. 99–107, Jan. 2009.

[24] M. Rossi, L. Rizzon, M. Fait, R. Passerone, and D. Brunelli, “Energy Neutral Wireless Sensing for Server Farms Monitoring,” *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 4, pp. 324–334, Sep. 2014.

[25] J. P. Carmo, L. M. Goncalves, and J. H. Correia, “Thermoelectric microconverter for energy harvesting systems,” *IEEE Transactions on Industrial Electronics*, vol. 57, no. 3, pp. 861–867, 2010.

[26] S. Dalola, V. Ferrari, M. Guizzetti, D. Marioli, E. Sardini, M. Serpelloni, and A. Taroni, “Autonomous Sensor System With Power Harvesting for Telemetric Temperature Measurements of Pipes,” *IEEE Transactions on Instrumentation and Measurement*, vol. 58, pp. 1471–1478, May. 2009.

[27] A. Prijic, L. Vracar, D. Vuckovic, D. Milic, and Z. Prijic, “Thermal Energy Harvesting Wireless Sensor Node in Aluminum Core PCB Technology,” *IEEE Sensors Journal*, vol. 15, pp. 337–345, Jan. 2015.

[28] P. C. Dias, F. J. O. Morais, M. B. de Morais Franca, E. C. Ferreira, A. Cabot, and J. A. Siqueira Dias, “Autonomous Multisensor System Powered by a Solar Thermoelectric Energy Harvester With Ultralow-Power Management Circuit,” *IEEE Transactions on Instrumentation and Measurement*, vol. 64, pp. 2918–2925, Nov. 2015.

[29] Yen Kheng Tan and S. K. Panda, “Optimized Wind Energy Harvesting System Using Resistance Emulator and Active Rectifier for Wireless Sensor Nodes,” *IEEE Transactions on Power Electronics*, vol. 26, pp. 38–50, Jan. 2011.

[30] D. Porcarelli, D. Brunelli, M. Magno, and L. Benini, “A Multi-Harvester architecture with hybrid storage devices and smart capabilities for low power systems,” in *IEEE International Symposium on Power Electronics Power Electronics, Electrical Drives, Automation and Motion*, pp. 946–951, Jun. 2012.

[31] C. Park and P. Chou, “AmbiMax: Autonomous Energy Harvesting Platform for Multi-Supply Wireless Sensor Nodes,” in *IEEE Communications Society on Sensor and Ad Hoc Communications and Networks*, vol. 1, pp. 168–177, Sept. 2006.

[32] R. Morais, S. G. Matos, M. A. Fernandes, A. L. Valente, S. F. Soares, P. Ferreira, and M. Reis, “Sun, wind and water flow as energy supply for small stationary data acquisition platforms,” *Elsevier Computers and Electronics in Agriculture*, vol. 64, pp. 120–132, Dec. 2008.

[33] M. Weimer, T. Paing, and R. Zane, “Remote area wind energy harvesting for low-power autonomous sensors,” in *IEEE Power Electronics Specialists Conference*, pp. 1–5, Jun. 2006.

[34] D. Porcarelli, D. Spenza, D. Brunelli, A. Cammarano, C. Petrioli, and L. Benini, “Adaptive Rectifier Driven by Power Intake Predictors for Wind Energy Harvesting Sensor Networks,” *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 3, pp. 471–482, Jun. 2015.

[35] Enerbee, “Generating energy from low speed and low force movements.” `http://www.enerbee.fr/innovation/`. Accessed: 2018-08-06.

[36] A. Sample, D. Yeager, P. Powledge, A. Mamishev, and J. Smith, “Design of an RFID-Based Battery-Free Programmable Sensing Platform,” *IEEE Transactions on Instrumentation and Measurement*, vol. 57, pp. 2608–2615, Nov. 2008.

[37] D. De Donno, L. Catarinucci, and L. Tarricone, “An UHF RFID Energy-Harvesting System Enhanced by a DC-DC Charge Pump in Silicon-on-Insulator Technology,” *IEEE Microwave and Wireless Components Letters*, vol. 23, pp. 315–317, Jun. 2013.

[38] S. Mandal, L. Turicchia, and R. Sarpeshkar, “A Low-Power, Battery-Free Tag for Body Sensor Networks,” *IEEE Pervasive Computing*, vol. 9, pp. 71–77, Jan. 2010.

[39] H. Reinisch, S. Gruber, H. Unterassinger, M. Wiessflecker, G. Hofer, W. Pribyl, and G. Holweg, “An Electro-Magnetic Energy Harvesting System With 190 nW Idle Mode Power Consumption for a BAW Based Wireless Sensor Node,” *IEEE Journal of Solid-State Circuits*, vol. 46, pp. 1728–1741, Jul. 2011.

[40] T. Sogorb, J. Llario, J. Pelegri, R. Lajara, and J. Alberola, “Studying the Feasibility of Energy Harvesting from Broadcast RF Station for WSN,” in *IEEE Instrumentation and Measurement Technology Conference*, pp. 1360–1363, May. 2008.

[41] S. Kim, R. Vyas, J. Bito, K. Niotaki, A. Collado, A. Georgiadis, and M. M. Tentzeris, “Ambient RF Energy-Harvesting Technologies for Self-Sustainable Standalone Wireless Sensor Platforms,” *Proceedings of the IEEE*, vol. 102, pp. 1649–1666, Nov. 2014.

[42] V. Gupta, A. Kandhalu, and R. R. Rajkumar, “Energy harvesting from electromagnetic energy radiating from AC power lines,” in *ACM Workshop on Hot Topics in Embedded Networked Sensors (HotEmNets)*, pp. 17:1–17:6, June 2010.

[43] R. Moghe, Yi Yang, F. Lambert, and D. Divan, “A scoping study of electric and magnetic field energy harvesting for wireless sensor networks in power system applications,” in *IEEE Energy Conversion Congress and Exposition*, pp. 3550–3557, Sep. 2009.

[44] I. Ikeda and T. Hattori, “US Patent 6496391 - Power supply unit utilizing a current transformer,” Dec. 2002.

[45] Acmite Market Intelligence, “Market Report Extract - Global Piezoelectric Device Market.” `https://www.acmite.com/brochure/Brochure-Piezoelectric-Device-Market-Report.pdf`. Accessed: 2018-08-06.

[46] W. Li, Siyuan He, and Shudong Yu, “Improving Power Density of a Cantilever Piezoelectric Power Harvester Through a Curved L-Shaped Proof Mass,” *IEEE Transactions on Industrial Electronics*, vol. 57, pp. 868–876, Mar. 2010.

[47] A. Khaligh, Peng Zeng, Xiaochun Wu, and Yang Xu, “A hybrid energy scavenging topology for human-powered mobile electronics,” in *IEEE Conference of Industrial Electronics*, pp. 448–453, Nov. 2008.

[48] T. Le, J. Han, A. Jouanne, K. Mayaram, and T. Fiez, “Piezoelectric Micro-Power Generation Interface Circuits,” *IEEE Journal of Solid-State Circuits*, vol. 41, pp. 1411–1420, Jun. 2006.

[49] H. Yu, Q. Yue, J. Zhou, and W. Wang, “A Hybrid Indoor Ambient Light and Vibration Energy Harvester for Wireless Sensor Nodes,” *MDPI Sensors*, vol. 14, pp. 8740–8755, May. 2014.

[50] EnOcean GmbH, “ECO 200 - Energy converter for motion energy harvesting.” `https://www.enocean.com/en/enocean_modules/eco-200/`. Accessed: 2018-08-06.

[51] J. Rocha, L. Goncalves, P. Rocha, M. Silva, and S. Lanceros-Mendez, “Energy Harvesting From Piezoelectric Materials Fully Integrated in Footwear,” *IEEE Transactions on Industrial Electronics*, vol. 57, pp. 813–819, Mar. 2010.

[52] N. Shenck and J. Paradiso, “Energy scavenging with shoe-mounted piezoelectrics,” *IEEE Micro*, vol. 21, no. 3, pp. 30–42, 2001.

[53] S. Platt, S. Farritor, K. Garvin, and H. Haider, “The Use of Piezoelectric Ceramics for Electric Power Generation Within Orthopedic Implants,” *IEEE/ASME Transactions on Mechatronics*, vol. 10, pp. 455–461, Aug. 2005.

[54] C. Dagdeviren, B. D. Yang, Y. Su, P. L. Tran, P. Joe, E. Anderson, J. Xia, V. Doraiswamy, B. Dehdashti, X. Feng, B. Lu, R. Poston, Z. Khalpey, R. Ghaffari, Y. Huang, M. J. Slepian, and J. A. Rogers, “Conformal piezoelectric energy harvesting and storage from motions of the heart, lung, and diaphragm.,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 111, pp. 1927–32, Feb. 2014.

[55] Yong Zhu, S. O. R. Moheimani, and M. R. Yuce, “A 2-DOF MEMS Ultrasonic Energy Harvester,” *IEEE Sensors Journal*, vol. 11, pp. 155–161, Jan. 2011.

[56] C. He, A. Arora, M. Kiziroglou, D. Yates, D. O’Hare, and E. Yeatman, “MEMS Energy Harvesting Powered Wireless Biometric Sensor,” in *IEEE International Workshop on Wearable and Implantable Body Sensor Networks*, pp. 207–212, Jun. 2009.

[57] M. Kiziroglou, C. He, and E. Yeatman, “Flexible substrate electrostatic energy harvester,” *IET Electronics Letters*, vol. 46, no. 2, p. 166, 2010.

[58] J. C. Park, D. H. Bang, and J. Y. Park, “Micro-Fabricated Electromagnetic Power Generator to Scavenge Low Ambient Vibration,” *IEEE Transactions on Magnetics*, vol. 46, pp. 1937–1942, Jun. 2010.

[59] Ö. Zorlu, E. T. Topal, and H. Kulah, “A Vibration-Based Electromagnetic Energy Harvester Using Mechanical Frequency Up-Conversion Method,” *IEEE Sensors Journal*, vol. 11, pp. 481–488, Feb. 2011.

[60] R. Mijarez, P. Gaydecki, and M. Burdekin, “Flood member detection for real-time structural health monitoring of sub-sea structures of offshore steel oilrigs,” *IOP Smart Materials and Structures*, vol. 16, pp. 1857–1869, Oct. 2007.

[61] C. Donovan, A. Dewan, H. Peng, D. Heo, and H. Beyenal, “Power management system for a 2.5 W remote sensor powered by a sediment microbial fuel cell,” *Elsevier Journal of Power Sources*, vol. 196, pp. 1171–1177, Feb. 2011.

[62] I. Ieropoulos, J. Greenman, C. Melhuish, I. Horsfield, I. Ieropoulos, J. Greenman, C. Melhuish, and I. Horsfield, “EcoBot-III: A robot with guts,” in *MIT Artificial Life X: Proceedings of the Twelfth International Conference on the Synthesis and Simulation of Living Systems*, Massachusetts Institute of Technology Press, Aug. 2010.

[63] I. A. Ieropoulos, P. Ledezma, A. Stinchcombe, G. Papaharalabos, C. Melhuish, and J. Greenman, “Waste to real energy: the first MFC powered mobile phone,” *Royal Society of Chemistry Journal on Physical Chemistry Chemical Physics (PCCP)*, vol. 15, p. 15312, Aug. 2013.

[64] Y. R. Thomas, M. Picot, A. Carer, O. Berder, O. Sentieys, and F. Barrière, “A single sediment-microbial fuel cell powering a wireless telecommunication system,” *Elsevier Journal of Power Sources*, vol. 241, pp. 703–708, Nov. 2013.

[65] Microchip Technology Inc., “MCP73871 - USB/AC Battery Charger with Power Path Management.” `http://www.analog.com/en/products/power-management/battery-management/coulomb-counter/ltc4150.html`. Accessed: 2018-08-06.

[66] F. Todeschini, C. Planat, P. Milazzo, S. Tricomi, P. Urard, and P. Benabes, “A Nano quiescent Current Power Management for Autonomous Wireless Sensor Network,” in *IEEE International Conference on Electronics, Circuits, and Systems (ICECS)*, pp. 815–818, Nov. 2013.

[67] J. Kim, J. Kim, and C. Kim, “A Regulated Charge Pump With a Low-Power Integrated Optimum Power Point Tracking Algorithm for Indoor Solar Energy Harvesting,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 58, pp. 802–806, Nov. 2011.

[68] G. Yu, K. W. R. Chew, Z. C. Sun, H. Tang, and L. Siek, “A 400 nW Single-Inductor Dual-Input–Tri-Output DC–DC Buck–Boost Converter With Maximum Power Point Tracking for Indoor Photovoltaic Energy Harvesting,” *IEEE Journal of Solid-State Circuits*, vol. 50, pp. 2758–2772, nov 2015.

[69] Y. K. Ramadass and A. P. Chandrakasan, “An Efficient Piezoelectric Energy Harvesting Interface Circuit Using a Bias-Flip Rectifier and Shared Inductor,” *IEEE Journal of Solid-State Circuits*, vol. 45, pp. 189–204, Jan. 2010.

[70] E. J. Carlson, K. Strunz, and B. P. Otis, “A 20 mV Input Boost Converter With Efficient Digital Control for Thermoelectric Energy Harvesting,” *IEEE Journal of Solid-State Circuits*, vol. 45, pp. 741–750, Apr. 2010.

[71] E. Dallago, A. L. Barnabei, A. Liberale, P. Malcovati, and G. Venchi, “An Interface Circuit for Low-Voltage Low-Current Energy Harvesting Systems,” *IEEE Transactions on Power Electronics*, vol. 30, pp. 1411–1420, Mar. 2015.

[72] C. Carvalho and N. Paulino, “Start-up circuit for low-power indoor light energy harvesting applications,” *IET Electronics Letters*, vol. 49, pp. 669–671, May. 2013.

[73] K. Kadirvel, Y. Ramadass, U. Lyles, J. Carpenter, V. Ivanov, V. McNeil, A. Chandrakasan, and B. Lum-Shue-Chan, “A 330nA energy-harvesting charger with battery management for solar and thermoelectric energy harvesting,” in *IEEE International Solid-State Circuits Conference (ISSCC)*, pp. 106–108, Feb. 2012.

[74] Analog Devices, Inc., “LTC3108 - Ultralow Voltage Step-Up Converter and Power Manager.” `http://www.analog.com/en/products/power-management/energy-harvesting/ltc3108.html`. Accessed: 2018-08-06.

[75] Texas Instruments Inc., “BQ25570 - Ultra Low Power Harvester Power Management IC with Boost Charger, and Nanopower Buck Converter.” `http://www.ti.com/product/bq25570`. Accessed: 2018-08-06.

[76] STMicroelectronics, “SPV1050 - Ultra low power energy harvester and battery charger with embedded MPPT and LDOs.” `https://www.st.com/en/power-management/spv1050.html`. Accessed: 2018-08-06.

[77] P. Zhang, C. M. Sadler, S. A. Lyon, and M. Martonosi, “Hardware design experiences in ZebraNet,” in *ACM International Conference on Embedded Networked Sensor Systems (SenSys)*, p. 227, 2004.

[78] PowerStream Technology, “How to charge Lithium Iron Phosphate Rechargeable Lithium Ion Batteries.” `http://www.powerstream.com/LLLF.htm`. Accessed: 2018-08-06.

[79] Samsung Electronics Co., Ltd., “Press release - Samsung Announces Cause of Galaxy Note7 Incidents in Press Conference.” `https://smsng.news/2lcUNo5`. Accessed: 2018-08-06.

[80] D. Linzen, S. Buller, E. Karden, and R. DeDoncker, “Analysis and Evaluation of Charge-Balancing Circuits on Performance, Reliability, and Lifetime of Supercapacitor Systems,” *IEEE Transactions on Industry Applications*, vol. 41, pp. 1135–1141, Sep. 2005.

[81] Y. Diab, P. Venet, and G. Rojat, “Comparison of the Different Circuits Used for Balancing the Voltage of Supercapacitors: Studying Performance and Lifetime of Supercapacitors,” in *European Symposium on Supercapacitors (ESSCAP)*, p. on CD, Nov. 2006.

[82] M. S. Chan, K. Chau, and C. Chan, “Effective Charging Method for Ultracapacitors,” *J-Stage Journal of Asian Electric Vehicles*, vol. 3, no. 2, pp. 771–776, 2005.

[83] S. Arkadiusz, J. Sikula, V. Sedlakova, J. Majzner, P. Sedlak, and T. Kuparowitz, “Voltage Dependence of Supercapacitor Capacitance,” *Polish Academy of Sciences Journal on Metrology and Measurement Systems*, vol. XVII, no. 4, pp. 537–547, 2010.

[84] STMicroelectronics, “EnFilm - rechargeable solid state lithium thin film battery.” `http://www.farnell.com/datasheets/1825905.pdf`. Accessed: 2018-08-06.

[85] D. Porcarelli, D. Brunelli, and L. Benini, “Characterization of lithium-ion capacitors for low-power energy neutral wireless sensor networks,” in *IEEE International Conference on Networked Sensing (INSS)*, pp. 1–4, Jun. 2012.

[86] Murata Manufacturing Co., Ltd., “Small Energy Device Cylinder type (UMAC).” `https://www.murata.com/products/smallenergydevice/uma/umac`. Accessed: 2018-08-06.

[87] Taiyo Yuden Co., Ltd., “LIC1030RS3R8206 - Cylinder Type Lithium Ion Capacitor(Low Resistance Type).” `http://ds3.yuden.co.jp/TYCOMPAS/ut/detail.do?productNo=LIC1030RS3R8206&dataUnit=M`. Accessed: 2018-08-06.

[88] B. Martinez, M. Monton, I. Vilajosana, and J. D. Prades, “The Power of Models: Modeling Power Consumption for IoT Devices,” *IEEE Sensors Journal*, vol. 15, pp. 5777–5789, Oct. 2015.

[89] F. J. Rincon, M. Paselli, J. Recas, Q. Zhao, M. Sanchez-Elez, D. Atienza, J. Penders, and G. De Micheli, “OS-Based Sensor Node Platform and Energy Estimation Model for Health-Care Wireless Sensor Networks,” in *IEEE Design, Automation and Test in Europe (DATE)*, pp. 1027–1032, Mar. 2008.

[90] C. Brandolese, W. Fomacian, F. Salice, and D. Sciuto, “An instruction-level functionality-based energy estimation model for 32-bits microprocessors,” in *IEEE Design Automation Conference (DAC)*, pp. 346–350, 2000.

[91] M. Bazzaz, M. Salehi, and A. Ejlali, "An Accurate Instruction-Level Energy Estimation Model and Tool for Embedded Systems," *IEEE Transactions on Instrumentation and Measurement*, vol. 62, pp. 1927–1934, Jul. 2013.

[92] O. Hahm and S. Adler, "Profiling energy consumption of Wireless Sensor Nodes with almost zero effort," in *IEEE International Conference on Communications (ICC)*, pp. 6273–6277, Jun. 2012.

[93] X. Jiang, P. Dutta, D. Culler, and I. Stoica, "Micro power meter for energy monitoring of wireless sensor networks at scale," in *ACM International Conference on Information Processing in Sensor Networks (IPSN)*, pp. 186–195, Apr. 2007.

[94] Analog Devices, Inc., "LTC4150 - Coulomb Counter/Battery Gas Gauge." `https://www.microchip.com/wwwproducts/en/en536670`. Accessed: 2018-08-06.

[95] Texas Instruments Inc., "BQ2019 - FLASH-Based Precision Multi-Chemistry Charge/Discharge Counter W/High-Speed 1-Wire I/F (HDQ)." `http://www.ti.com/product/bq2019`. Accessed: 2018-08-06.

[96] N. Chang, K. Kim, and H. G. Lee, "Cycle-accurate energy consumption measurement and analysis," in *ACM International Symposium on Low Power Electronics and Design (ISLPED)*, pp. 185–190, 2000.

[97] S. Naderiparizi, A. N. Parks, F. S. Parizi, and J. R. Smith, "μMonitor: In-situ energy monitoring with microwatt power consumption," in *IEEE International Conference on RFID (RFID)*, pp. 1–8, May. 2016.

[98] Analog Devices, Inc., "LTC3335 - Nanopower Buck-Boost DC/DC with Integrated Coulomb Counter." `http://www.analog.com/en/products/power-management/battery-management/coulomb-counter/ltc3335.html`. Accessed: 2018-08-06.

[99] M. García, J. M. Maruri, L. Marroyo, E. Lorenzo, and M. Pérez, "Partial shadowing, MPPT performance and inverter configurations: observations at tracking PV plants," *Wiley Journal of Progress in Photovoltaics: Research and Applications*, vol. 16, pp. 529–536, Sep. 2008.

[100] M. Bodur and M. Ermis, "Maximum power point tracking for low power photovoltaic solar panels," in *IEEE Mediterranean Electrotechnical Conference (MELECON)*, pp. 758–761, 1994.

[101] F. Liu, Y. Kang, Y. Zhang, and S. Duan, "Comparison of P&O and hill climbing MPPT methods for grid-connected PV converter," in *IEEE Conference on Industrial Electronics and Applications*, pp. 804–807, Jun. 2008.

[102] Z. Liang, R. Guo, and A. Huang, "A new cost-effective analog maximum power point tracker for PV systems," in *IEEE Energy Conversion Congress and Exposition*, pp. 624–631, Sep. 2010.

[103] J. Ahmad and H.-J. Kim, "A Voltage Based Maximum Power Point Tracker for Low Power and Low Cost Photovoltaic Applications," *WASET International Journal of Electronics and Communication Engineering*, pp. 712–715, Nov. 2009.

[104] V. Scarpa, S. Buso, and G. Spiazzi, "Low-Complexity MPPT Technique Exploiting the PV Module MPP Locus Characterization," *IEEE Transactions on Industrial Electronics*, vol. 56, pp. 1531–1538, May. 2009.

[105] Y. Chen, K. Smedley, and F. Vacher, "A new maximum power point tracking controller for photovoltaic power generation," *IEEE Applied Power Electronics Conference and Exposition*, vol. 00, pp. 58–62, Feb. 2003.

[106] D. Carli, D. Brunelli, D. Bertozzi, and L. Benini, "A high-efficiency wind-flow energy harvester using micro turbine," in *IEEE International Symposium on Power Electronics Electrical Drives Automation and Motion (SPEEDAM)*, pp. 778–783, Jun. 2010.

[107] A. Shrivastava, N. E. Roberts, O. U. Khan, D. D. Wentzloff, and B. H. Calhoun, "A 10 mV-Input Boost Converter With Inductor Peak Current Control and Zero Detection for Thermoelectric and Solar Energy Harvesting With 220 mV Cold-Start and 14.5 dBm, 915 MHz RF Kick-St," *IEEE Journal of Solid-State Circuits*, vol. 50, pp. 1820–1832, Aug. 2015.

[108] IMC Networks, "MTBF, MTTR, MTTF & FIT : Expanation of Terms." `http://www.bb-elec.com/Learning-Center/All-White-Papers/Fiber/MTBF,-MTTR,-MTTF,-FIT-Explanation-of-Terms/MTBF-MTTR-MTTF-FIT-10262012-pdf.pdf`. Accessed: 2018-08-23.

[109] Department of Defense, United States of America, "Military Handbook - Reliability Prediction of Electronic Equipment." `https://snebulos.mit.edu/projects/reference/MIL-STD/MIL-HDBK-217F-Notice2.pdf`. Accessed: 2018-08-09.

[110] J. Jones and J. Hayes, "A comparison of electronic-reliability prediction models," *IEEE Transactions on Reliability*, vol. 48, pp. 127–134, Jun. 1999.

[111] International Electrotechnical Commission, "IEC 60063:2015 - Preferred number series for resistors and capacitors." `https://webstore.iec.ch/publication/22011`. Accessed: 2018-08-28.

[112] B. Xu, A. Oudalov, A. Ulbig, G. Andersson, and D. S. Kirschen, "Modeling of Lithium-Ion Battery Degradation for Cell Life Assessment," *IEEE Transactions on Smart Grid*, vol. 9, pp. 1131–1140, Mar. 2016.

[113] CDE Cornell Dubilier, "Specifications - Type EDL, Electric Double Layer Supercapacitors." `http://www.farnell.com/datasheets/2576769.pdf`. Accessed: 2018-08-06.

[114] K. Martinez, R. Ong, and J. Hart, “Glacsweb: a sensor network for hostile environments,” in *IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (SECON)*, pp. 81–87, 2004.

[115] G. Werner-allen, K. Lorincz, J. Johnson, J. Lees, and M. Welsh, “Fidelity and Yield in a Volcano Monitoring Sensor Network,” in *ACM Symposium on Operating Systems Design and Implementation*, pp. 381–396, 2006.

[116] A. S. Weddell, N. R. Harris, and N. M. White, “An Efficient Indoor Photovoltaic Power Harvesting System for Energy-Aware Wireless Sensor Nodes,” in *Eurosensors*, Sep. 2008.

[117] T. Zhu, Z. Zhong, Y. Gu, T. He, and Z.-L. Zhang, “Leakage-aware energy synchronization for wireless sensor networks,” in *ACM International Conference on Mobile Systems, Applications and Services*, pp. 319–332, Jun. 2009.

[118] M. Minami, T. Morito, H. Morikawa, and T. Aoyama, “Solar biscuit: A battery-less wireless sensor network system for environmental monitoring applications,” in *IEEE International Workshop on Networked Sensing Systems*, Jun. 2005.

[119] J. Polastre and D. Culler, “Perpetual environmentally powered sensor networks,” in *IEEE International Symposium on Information Processing in Sensor Networks (ISPN)*, pp. 463–468, Apr. 2005.

[120] C.-Y. Chen and P. H. Chou, “DuraCap: A supercapacitor-based, power-bootstrapping, maximum power point tracking energy-harvesting system,” in *ACM/IEEE International Symposium on Low Power Electronics and Design (ISLPED)*, pp. 313–318, Aug. 2010.

[121] T.-N. Le, A. Pegatoquet, O. Sentieys, O. Berder, and C. Belleudy, “Duty-cycle power manager for thermal-powered Wireless Sensor Networks,” in *IEEE Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pp. 1645–1649, IEEE, Sep. 2013.

[122] O. Berder and O. Sentieys, “PowWow : Power Optimized Hardware/Software Framework for Wireless Motes,” in *VDE International Conference on Architecture of Computing Systems (ARCS)*, (Hannover), pp. 1–5, Feb. 2010.

[123] T.-N. Le, A. Pegatoquet, O. Berder, O. Sentieys, and A. Carer, “Energy-Neutral Design Framework for Supercapacitor-Based Autonomous Wireless Sensor Networks,” *ACM Journal on Emerging Technologies in Computing Systems*, vol. 12, pp. 1–21, Sep. 2015.

[124] L. Vračar, A. Prijić, D. Nešić, S. Dević, and Z. Prijić, “Photovoltaic Energy Harvesting Wireless Sensor Node for Telemetry Applications Optimized for Low Illumination Levels,” *MDPI Electronics*, vol. 5, p. 26, Jun. 2016.

[125] W. M. Tan and S. A. Jarvis, “On the design of an energy-harvesting noise-sensing WSN mote,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2014, pp. 1–18, Oct. 2014.

[126] B. S. Srujana, P. Mathews, and V. Harigovindan, “Multi-source Energy Harvesting System for Underwater Wireless Sensor Networks,” *Elsevier Procedia Computer Science*, vol. 46, pp. 1041–1048, 2015.

[127] C. Park and P. H. Chou, “Power utility maximization for multiple-supply systems by a load-matching switch,” in *IEEE International Symposium on Low Power Electronics and Design (ISPLED)*, pp. 168–173, 2004.

[128] Y. K. Tan and S. K. Panda, “Energy Harvesting From Hybrid Indoor Ambient Light and Thermal Energy Sources for Enhanced Performance of Wireless Sensor Nodes,” *IEEE Transactions on Industrial Electronics*, vol. 58, pp. 4424–4435, Sep. 2011.

[129] N. Ferry, S. Ducloyer, N. Julien, and D. Jutel, “Power and Energy Aware Design of an Autonomous Wireless Sensor Node,” *ACSIJ Advances in Computer Science: an International Journal*, vol. 2, Sep. 2013.

[130] D. Carli, D. Brunelli, L. Benini, and M. Ruggeri, “An effective multi-source energy harvester for low power applications,” in *IEEE Design, Automation & Test in Europe (DATE)*, pp. 1–6, Mar. 2011.

[131] A. S. Weddell, N. J. Grabham, N. R. Harris, and N. M. White, “Modular Plug-and-Play Power Resources for Energy-Aware Wireless Sensor Nodes,” in *IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, pp. 1–9, Jun. 2009.

[132] J. Colomer-Farrarons, P. Miribel-Catala, A. Saiz-Vela, and J. Samitier, “A Multiharvested Self-Powered System in a Low-Voltage Low-Power Technology,” *IEEE Transactions on Industrial Electronics*, vol. 58, pp. 4250–4263, Sep. 2011.

[133] L. Vijay, K. K. Greeshma, and NS. Murty, “Architecture for ASIC based batteryless multi-source energy harvesting system,” in *IEEE International Conference on VLSI Systems, Architecture, Technology and Applications (VLSI-SATA)*, pp. 1–6, Jan. 2015.

[134] S. Bandyopadhyay and A. P. Chandrakasan, “Platform Architecture for Solar, Thermal, and Vibration Energy Combining With MPPT and Single Inductor,” *IEEE Journal of Solid-State Circuits*, vol. 47, pp. 2199–2215, Sep. 2012.

[135] Analog Devices, Inc., “LTspice - High performance SPICE simulation software.” `http://www.analog.com/en/design-center/design-tools-and-calculators/ltspice-simulator.html`. Accessed: 2018-08-06.

[136] EE Times, “Fundamentals of power system ORing.” `https://www.eetimes.com/document.asp?doc_id=1273175&page_number=1`. Accessed: 2018-08-06.

[137] R. White, “Using Redundant DC Power in High Availability Systems,” in *IEEE Applied Power Electronics Conference and Exposition (APEC)*, pp. 848–853, Mar. 2006.

[138] Vishay Intertechnology, Inc., “SiP32431 - Ultra Low Leakage and Quiescent Current, 1 A Load Switch with Reverse Blocking.” `https://www.vishay.com/power-ics/list/product-66597/`. Accessed: 2018-08-06.

[139] Lattice Semiconductor Corporation, “iCE40 Ultra / UltraLite - Product page.” `http://www.latticesemi.com/en/Products/FPGAandCPLD/iCE40Ultra`. Accessed: 2018-08-06.

[140] Microsemi Corporation, “IGLOO - Product page.” `https://www.microsemi.com/product-directory/fpgas/1689-igloo`. Accessed: 2018-08-06.

[141] Cypress Semiconductor Corporation, “PSoC 4 - 32-bit Arm Cortex-M0.” `http://www.cypress.com/products/32-bit-arm-cortex-m0-psoc-4`. Accessed: 2018-08-06.

[142] Texas Instruments Inc., “MSP430FR5969 - MSP430FR5969 16 MHz Ultra-Low-Power Microcontroller featuring 64 KB FRAM, 2 KB SRAM, 40 IO.” `http://www.ti.com/product/msp430fr5969`. Accessed: 2018-08-06.

[143] Texas Instruments Inc., “TPS60210 - Regulated 3.3-V 100-mA Low-Ripple Charge Pump with Ultra Low Operating Current.” `http://www.ti.com/product/TPS60210`. Accessed: 2018-08-06.

[144] Wi6labs, “Wi6labs LoRa platform.” `http://www.wi6labs.com/lora/`, 2016. [Online; accessed 10-May-2016].

[145] H. Veendrick, “Short-circuit dissipation of static CMOS circuitry and its impact on the design of buffer circuits,” *IEEE Journal of Solid-State Circuits*, vol. 19, pp. 468–473, Aug. 1984.

[146] Linus Torvalds, “The Linux Kernel Archives.” `https://www.kernel.org/`. Accessed: 2018-09-04.

[147] Riot, “Riot : The friendly Operating System for the Internet of Things..” `https://riot-os.org/`. Accessed: 2018-09-04.

[148] Contiki, “Contiki: The Open Source OS for the Internet of Things.” `http://www.contiki-os.org/`. Accessed: 2018-09-04.

[149] Real Time Engineers Ltd., “The FreeRTOS™ Kernel.” `https://www.freertos.org/`. Accessed: 2018-09-04.

[150] Gregory Nutt, “NuttX Real-Time Operating System.” `http://nuttx.org/`. Accessed: 2018-09-05.

[151] uClinux, “uClinux - Embedded Linux/Microcontroller Project.” `http://www.uclinux.org/`. Accessed: 2018-09-05.

[152] Wisebatt, “A unique simulation platform for IoT.” `https://www.wisebatt.com/`. Accessed: 2018-10-29.

[153] D. Snowdon, S. Ruocco, and G. Heiser, “Power management and dynamic voltage scaling: Myths and facts,” in *ACM Workshop on Power Aware Real-time Computing*, 2005.

[154] M. Fourniol, V. Gies, V. Barchasz, E. Kussener, H. Barthelemy, R. Vauche, and H. Glotin, “Low-Power Wake-Up System based on Frequency Analysis for Environmental Internet of Things,” in *IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA)*, pp. 1–6, Jul. 2018.

[155] F. Ait Aoudia, M. Gautier, A. Courtay, and O. Berder, “Etude du compromis latence-consommation d ’ énergie des radios longue portée ,” in *Colloque du GDR SoC-SiP*, Jun. 2017.

[156] W. Benali, M. Le Bot, C. Langlais, and S. Kerouedan, “Power consumption of Wi-Fi transceivers,” in *IEEE International Symposium on Wireless Communication Systems*, vol. 2016-Octob, pp. 213–217, Sep. 2016.

[157] Semtech, “Semtech LoRa technology overview.” `https://www.semtech.com/lora`. Accessed: 2018-09-05.

[158] Sigfox, “Sigfox, the world’s leading IoT services provider..” `https://www.sigfox.com/en`. Accessed: 2018-09-05.

[159] A. El-Hoiydi and J.-D. Decotignie, “WiseMAC: an ultra low power MAC protocol for the downlink of infrastructure wireless sensor networks,” in *IEEE International Symposium on Computers And Communications (ISCC)*, vol. 1, pp. 244–251, July 2004.

[160] T. N. Le, M. Magno, A. Pegatoquet, O. Berder, O. Sentieys, and E. Popovici, “Ultra low power asynchronous MAC protocol using wake-up radio for energy neutral WSN,” in *ACM International Workshop on Energy Neutral Sensing Systems (ENSSys)*, pp. 1–6, Nov. 2013.

[161] Y. Sun, O. Gurewitz, and D. B. Johnson, “RI-MAC,” in *ACM conference on Embedded network sensor systems (SenSys)*, (New York, New York, USA), p. 1, Nov. 2008.

[162] M. S. Durante and S. Mahlknecht, “An Ultra Low Power Wakeup Receiver for Wireless Sensor Nodes,” in *IEEE International Conference on Sensor Technologies and Applications*, pp. 167–170, Jun. 2009.

[163] S. J. Marinkovic and E. M. Popovici, “Nano-Power Wireless Wake-Up Receiver With Serial Peripheral Interface,” *IEEE Journal on Selected Areas in Communications*, vol. 29, pp. 1641–1647, Sep. 2011.

[164] M. Magno, F. Ait Aoudia, M. Gautier, O. Berder, and L. Benini, “WULoRa: An energy efficient IoT end-node for energy harvesting and heterogeneous communication,” in *Design, Automation & Test in Europe (DATE)*, pp. 1528–1533, Mar. 2017.

[165] IEEE Computer Society. Microprocessor Standards Committee., Institute of Electrical and Electronics Engineers., and IEEE-SA Standards Board., *IEEE standard for floating-point arithmetic*. IEEE, 2008.

[166] Robert Munafo, "Survey of Floating-Point Formats." `http://www.mrob.com/pub/math/floatformats.html`. Accessed: 2018-09-04.

[167] Khronos Group, "Unsigned 10-bit floating-point numbers." `https://www.khronos.org/registry/DataFormat/specs/1.2/dataformat.1.2.html#10bitfp`. Accessed: 2018-09-04.

[168] A. Kansal, J. Hsu, S. Zahedi, and M. B. Srivastava, "Power management in energy harvesting sensor networks," *ACM Transactions on Embedded Computing Systems*, vol. 6, Sep. 2007.

[169] T.-N. Le, O. Sentieys, O. Berder, A. Pegatoquet, and C. Belleudy, "Power Manager with PID Controller in Energy Harvesting Wireless Sensor Networks," in *IEEE International Conference on Green Computing and Communications*, pp. 668–670, Nov. 2012.

[170] J. Recas Piorno, C. Bergonzini, D. Atienza, and T. Simunic Rosing, "Prediction and management in energy harvested wireless sensor nodes," in *IEEE International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology*, pp. 6–10, May. 2009.

[171] S. Peng and C. Low, "Prediction free energy neutral power management for energy harvesting wireless sensor nodes," *Elsevier Ad Hoc Networks*, vol. 13, pp. 351–367, Feb. 2014.

[172] F. Ait Aoudia, M. Gautier, and O. Berder, "Learning to survive: Achieving energy neutrality in wireless sensor networks using reinforcement learning," in *IEEE International Conference on Communications (ICC)*, pp. 1–6, May. 2017.

[173] G. M. de Araújo, A. R. Pinto, J. Kaiser, and L. B. Becker, "Genetic Machine Learning Approach for Link Quality Prediction in Mobile Wireless Sensor Networks," in *Cooperative Robots and Sensor Networks. Studies in Computational Intelligence*, pp. 1–18, Springer, Berlin, Heidelberg, 2014.

[174] G. M. de Araujo, J. Kaiser, and L. B. Becker, "An optimized Markov model to predict link quality in mobile wireless sensor networks," in *IEEE Symposium on Computers and Communications (ISCC)*, pp. 000307–000312, Jul. 2012.

[175] T. Liu and A. E. Cerpa, "Data-driven link quality prediction using link features," *ACM Transactions on Sensor Networks*, vol. 10, pp. 1–35, Jan. 2014.

[176] D. R. Cox, "Prediction by Exponentially Weighted Moving Averages and Related Methods," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 23, pp. 414–422, 1961.

[177] C. Moser, L. Thiele, D. Brunelli, and L. Benini, "Adaptive Power Management in Energy Harvesting Systems," in *IEEE Design, Automation & Test in Europe (DATE)*, pp. 773–778, Apr. 2007.

[178] C. Bergonzini, D. Brunelli, and L. Benini, "Algorithms for harvested energy prediction in batteryless wireless sensor networks," in *IEEE International Workshop on Advances in sensors and Interfaces*, pp. 144–149, Jun. 2009.

[179] J. Lu and K. Whitehouse, "SunCast: Fine-grained prediction of natural sunlight levels for improved daylight harvesting," in *ACM/IEEE 11th International Conference on Information Processing in Sensor Networks (IPSN)*, pp. 245–256, Apr. 2012.

[180] C. Renner, "Solar harvest prediction supported by cloud cover forecasts," in *ACM International Workshop on Energy Neutral Sensing Systems (ENSSys)*, (New York, New York, USA), pp. 1–6, 2013.

[181] S. Kosunalp, "A New Energy Prediction Algorithm for Energy-Harvesting Wireless Sensor Networks With Q-Learning," *IEEE Access*, vol. 4, pp. 5755–5763, 2016.

[182] I. T. Christou, *Quantitative methods in supply chain management : models and algorithms.* Springer, 2012.

[183] A. Jushi, A. Pegatoquet, and T. N. Le, "Wind Energy Harvesting for Autonomous Wireless Sensor Networks," in *IEEE Euromicro Conference on Digital System Design (DSD)*, pp. 301–308, Aug. 2016.

[184] A. Cammarano, C. Petrioli, and D. Spenza, "Pro-Energy: A novel energy prediction model for solar and wind energy-harvesting wireless sensor networks," in *IEEE International Conference on Mobile Ad-Hoc and Sensor Systems (MASS)*, pp. 75–83, Oct. 2012.

[185] A. Cammarano, C. Petrioli, and D. Spenza, "Improving energy predictions in EH-WSNs with Pro-Energy-VLT," in *ACM Conference on Embedded Networked Sensor Systems (SenSys)*, (New York, New York, USA), pp. 1–2, 2013.

[186] D. Fan, L. L. Ruiz, J. Gong, and J. Lach, "Profiling, modeling, and predicting energy harvesting for self-powered body sensor platforms," in *IEEE International Conference on Wearable and Implantable Body Sensor Networks (BSN)*, pp. 402–407, Jun. 2016.

[187] C. K. Ho, P. D. Khoa, and P. C. Ming, "Markovian models for harvested energy in wireless communications," in *IEEE International Conference on Communication Systems*, pp. 311–315, Nov. 2010.

[188] A. Castagnetti, A. Pegatoquet, C. Belleudy, and M. Auguin, "A framework for modeling and simulating energy harvesting WSN nodes with efficient power management policies," *EURASIP Journal on Embedded Systems*, vol. 2012, no. 1, pp. 1–20, 2012.

[189] C. M. Vigorito, D. Ganesan, and A. G. Barto, "Adaptive Control of Duty Cycling in Energy- Harvesting Wireless Sensor Networks," in *IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, pp. 21–30, 2007.

[190] A. G. Watts, M. Prauzek, P. Musilek, E. Pelikan, and A. Sanchez-Azofeifa, "Fuzzy power management for environmental monitoring systems in tropical regions," in *IEEE International Joint Conference on Neural Networks (IJCNN)*, pp. 1719–1726, Jul. 2014.

[191] F. Ait Aoudia, M. Gautier, and O. Berder, "Fuzzy Power Management for Energy Harvesting Wireless Sensor Nodes," in *IEEE International Conference on Communications (ICC)*, May. 2016.

[192] F. Ait Aoudia, M. Gautier, M. Magno, O. Berder, and L. Benini, "SNW-MAC: An Asynchronous Protocol Leveraging Wake-Up Receivers for Data Gathering in Star Networks," in *International Conference on Sensor Systems and Software (S-CUBE)*, pp. 3–14, Springer, 2017.

[193] R. C. Hsu, C.-T. Liu, and H.-L. Wang, "A Reinforcement Learning-Based ToD Provisioning Dynamic Power Management for Sustainable Operation of Energy Harvesting Wireless Sensor Node," *IEEE Transactions on Emerging Topics in Computing*, vol. 2, pp. 181–191, Jun. 2014.

[194] Official Journal of the European Union, " Directive 2014/53/EU of the European Parliament and of the Council of 16 April 2014 on the harmonisation of the laws of the Member States relating to the making available on the market of radio equipment and repealing Directive 1999/5/EC Text with EEA relevance ." `http://data.europa.eu/eli/dir/2014/53/oj`. Accessed: 2018-09-24.

[195] Official Journal of the European Union, " Commission Implementing Decision (EU) 2017/1483 of 8 August 2017 amending Decision 2006/771/EC on harmonisation of the radio spectrum for use by short-range devices and repealing Decision 2006/804/EC (notified under document C(2017) 5464) ." `http://data.europa.eu/eli/dec_impl/2017/1483/oj`. Accessed: 2018-09-24.

[196] Ben Brewer, "Cross Platform Fixed Point Maths Library." `https://github.com/PetteriAimonen/libfixmath`. Accessed: 2017-09-26.

[197] R. Margolies, P. Kinget, I. Kymissis, G. Zussman, M. Gorlatova, J. Sarik, G. Stanje, J. Zhu, P. Miller, M. Szczodrak, B. Vigraham, and L. Carloni, "Energy-Harvesting Active Networked Tags (EnHANTs): Prototyping and Experimentation," *ACM Transactions on Sensor Networks*, vol. 11, pp. 1–27, Nov. 2015.

[198] J. Lehoczky, L. Sha, and Y. Ding, "The rate monotonic scheduling algorithm: exact characterization and average case behavior," in *IEEE Symposium on Real-Time Systems*, pp. 166–171.

[199] N. Audsley, A. Burns, M. F. Richardson, and A. J. Wellings, “Hard Real-Time Scheduling: The Deadline-Monotonic Approach,” in *IEEE Workshop on Real-Time Operating Systems and Software*, pp. 133—-137, 1991.

[200] V. Sivaraman and F. M. Chiussi, “Statistical analysis of delay bound violations at an earliest deadline (EDF) scheduler,” *Elsevier Journal on Performance Evaluation*, pp. 36—-37, 1999.

[201] P. Jansen, S. Mullender, J. Scholten, and P. Havinga, *Lightweight EDF Scheduling with Deadline Inheritance.* No. 2003-23 in CTIT-technical reports, Centre for Telematics and Information Technology (CTIT), May. 2003.

[202] S. Liu, Q. Qiu, and Q. Wu, “Energy Aware Dynamic Voltage and Frequency Selection for Real-Time Systems with Energy Harvesting,” in *IEEE Design, Automation and Test in Europe (DATE)*, pp. 236–241, Mar. 2008.

[203] Shaobo Liu, Jun Lu, Qing Wu, and Qinru Qiu, “Harvesting-Aware Power Management for Real-Time Systems With Renewable Energy,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 20, pp. 1473–1486, Aug. 2012.

[204] Y. Tan and X. Yin, “A dynamic scheduling algorithm for energy harvesting embedded systems,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2016, p. 114, Nov. 2016.

[205] M. Chetto, “Task Scheduling In Energy Harvesting Real-time Embedded Systems,” *OMICS Journal of Information Technology & Software Engineering*, vol. 02, no. 03, 2012.

[206] Y. Chandarli, Y. Abdeddaim, and D. Masson, “The Fixed Priority Scheduling Problem for Energy Harvesting Real-Time Systems,” in *IEEE International Conference on Embedded and Real-Time Computing Systems and Applications*, pp. 415–418, Aug. 2012.

[207] Y. Abdeddaim, Y. Chandarli, and D. Masson, “The Optimality of PFPasap Algorithm for Fixed-Priority Energy-Harvesting Real-Time Systems,” in *IEEE Euromicro Conference on Real-Time Systems*, pp. 47–56, Jul. 2013.

[208] K. Faramarzi, M. Hasanloo, and M. Kargahi, “The PFPASAP algorithm for energy harvesting real-time systems with a non-ideal supercapacitor,” in *IEEE International Conference on Computer and Knowledge Engineering (ICCKE)*, pp. 279–284, Oct. 2015.

[209] Y. Abdeddaïm, Y. Chandarli, and D. Masson, “Toward an Optimal Fixed-Priority Algorithm for Energy-Harvesting Real-Time Systems,” in *IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*, pp. 45—-48, Apr. 2013.

[210] C. Moser, D. Brunelli, L. Thiele, and L. Benini, “Lazy Scheduling for Energy Harvesting Sensor Nodes,” in *IFIP Working Conference on Distributed and Parallel Embedded Systems (DIPES)*, pp. 125–134, 2006.

[211] D. Audet, N. MacMillan, D. Marinakis, and Kui Wu, "Scheduling recurring tasks in energy harvesting sensors," in *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 277–282, Apr. 2011.

[212] H. El Ghor, M. Chetto, and R. H. Chehade, "A real-time scheduling framework for embedded systems with environmental energy harvesting," *Elsevier Journal of Computers & Electrical Engineering*, vol. 37, pp. 498–510, Jul. 2011.

[213] M. Chetto, D. Masson, and S. Midonnet, "Fixed Priority Scheduling Strategies for Ambient Energy-Harvesting Embedded Systems," in *IEEE/ACM International Conference on Green Computing and Communications*, pp. 50–55, Aug. 2011.

[214] V. S. Rao, R. V. Prasad, and I. G. M. M. Niemegeers, "Optimal task scheduling policy in energy harvesting wireless sensor networks," in *IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1030–1035, Mar. 2015.

[215] J. Lu and Q. Qiu, "Scheduling and mapping of periodic tasks on multi-core embedded systems with energy harvesting," in *IEEE International Green Computing Conference and Workshops*, pp. 1–6, Jul. 2011.

[216] A. Maeda, S. Kasahara, and Y. Takahashi, "Dynamic Task Scheduling for Energy Harvesting Nodes in Wireless Sensor Networks," in *ACM International Conference on Queueing Theory and Network Applications (QTNA)*, (New York, New York, USA), pp. 1–8, 2016.

[217] A. Eswaran, A. Rowe, and R. Rajkumar, "Nano-RK: An Energy-Aware Resource-Centric RTOS for Sensor Networks," in *IEEE International Real-Time Systems Symposium (RTSS)*, pp. 256–265.

[218] G. Bhat, J. Park, and U. Y. Ogras, "Near-optimal energy allocation for self-powered wearable systems," in *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pp. 368–375, Nov. 2017.

[219] H. W. Kuhn and A. W. Tucker, "Nonlinear programming," in *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, pp. 481–492, University of California Press, 1951.

[220] F. Ait Aoudia, M. Gautier, M. Magno, O. Berder, and L. Benini, "Leveraging Energy Harvesting and Wake-Up Receivers for Long-Term Wireless Sensor Networks," *MDPI Sensors*, vol. 18, p. 1578, May. 2018.

[221] Le Mag, "Big Data : 8 projets retenus pour préparer la ville de demain." `http://www.lemag-numerique.com/2015/10/big-data-8-projets-retenus-pour-preparer-la-ville-de-demain-7989`. Accessed: 2018-10-04.

[222] W.-H. Wang, M. Palaniswami, and S. H. Low, "Application-Oriented Flow Control: Fundamentals, Algorithms and Fairness," *IEEE/ACM Transactions on Networking*, vol. 14, pp. 1282–1291, Nov. 2006.

[223] Trameto, "Delivering any-many-multi source, micro-energy harvesting." `https://trameto.com/`. Accessed: 2018-10-23.

[224] T. Bouguera, J.-F. Diouris, J.-J. Chaillout, R. Jaouadi, and G. Andrieux, "Energy Consumption Model for Sensor Nodes Based on LoRa and LoRaWAN," *MDPI Sensors*, vol. 18, p. 2104, Jun. 2018.

UNIVERSITE
BRETAGNE
LOIRE / MATHSTIC

UNIVERSITÉ DE
RENNES 1

Titre : Conception et implémentation de stratégies de gestion d'énergie pour modules radio longue portée avec récupération d'énergie

Mots clés : Récupération d'énergie, Réseaux de capteurs, Longue portée, gestion d'énergie

Résumé : L'avènement de l'Internet des Objets a permis de déployer de nombreux réseaux de capteurs sans-fil. Ces réseaux sont utilisés dans des domaines aussi variés que l'agriculture, l'industrie ou la ville intelligente, où ils permettent d'optimiser finement les processus. Ces appareils sont le plus souvent alimentés par des piles ou batteries, ce qui limite leur autonomie. De plus, il n'est pas toujours possible ou financièrement viable de changer ou recharger les batteries.

Une solution possible est d'alimenter ces capteurs en récupérant l'énergie présente dans l'environnement alentour. Ces sources d'énergie sont cependant peu fiables, et le capteur doit être capable d'éviter de vider complètement sa réserve d'énergie. Afin de moduler sa consommation d'énergie, le capteur peut adapter sa qualité de service à ses capacités énergétiques. L'appareil peut ainsi fonctionner en continu sans interruption de service.

Cette thèse présente les méthodes utilisées pour la conception d'un capteur entièrement autonome alimenté par récupération d'énergie ambiante, communiquant sur un réseau longue portée LoRa. Afin d'assurer l'alimentation électrique, une carte permettant de récupérer de l'énergie depuis plusieurs sources d'énergie simultanément a été conçue. Un module logiciel de gestion d'énergie a ensuite été développé afin de calculer un budget énergétique que le capteur peut dépenser, et choisir la meilleure manière de dépenser ce budget pour exécuter une ou plusieurs tâches. Ce travail a ainsi permis le développement d'un prototype de produit industriel entièrement autonome en énergie.

Title : Design and implementation of power management strategies for long range radio modules with energy harvesting

Keywords : Energy harvesting, Wireless Sensor Network, Long Range, Power management

Abstract : The advent of the Internet of Things has enabled the roll-out of a multitude of Wireless Sensor Networks. These networks can be used in various fields, such as agriculture, industry or the smart city, where they facilitate fine optimization of processes. These devices are often powered by primary or rechargeable batteries, which limits their battery life. Moreover, it is sometimes not possible or financially viable to change and/or recharge these batteries.

A possible solution is to harvest energy from the environment to power these sensors. But these energy sources are unreliable, and the sensor must be able to prevent the complete depletion of its energy storage. In order to adapt its energy consumption, the node can match its quality of service to its energetical capabilities. Thus, the device can continuously operate without any service interruption.

This thesis presents the methods used for the conception of a completely autonomous sensor, powered by energy harvesting and communicating through a long range LoRa network. In order to ensure its power supply, a board has been designed to harvest energy from multiple energy sources simultaneously. A power management software module has then been developed to calculate an energy budget the sensor can use, and to choose the best way to spend this budget over one or multiple tasks. This work has enabled the development of an energy autonomous industrial sensor prototype.