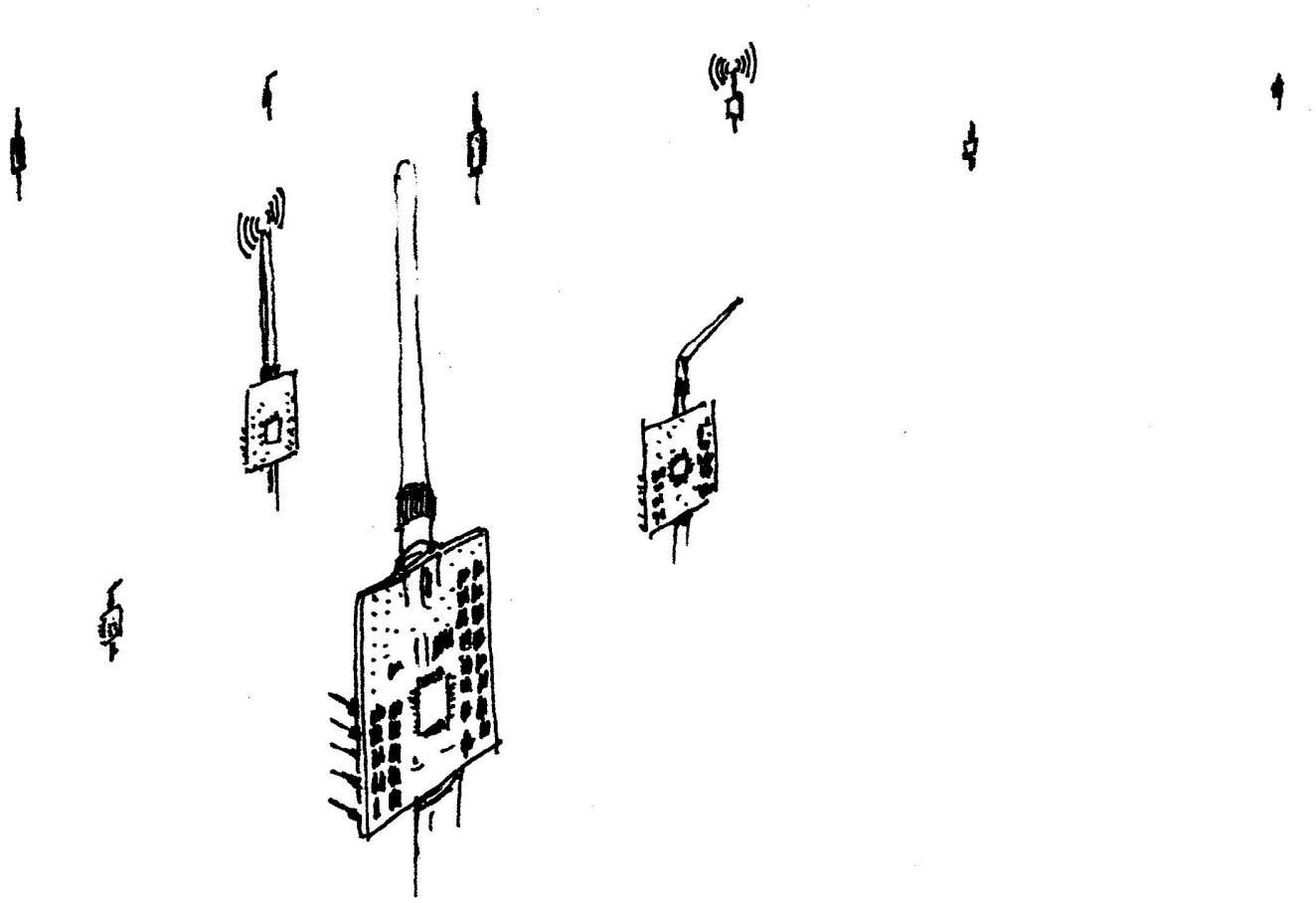


José Pedro de Matos Nogueira Amaro

Wireless Sensor Networks Running on Harvested Power

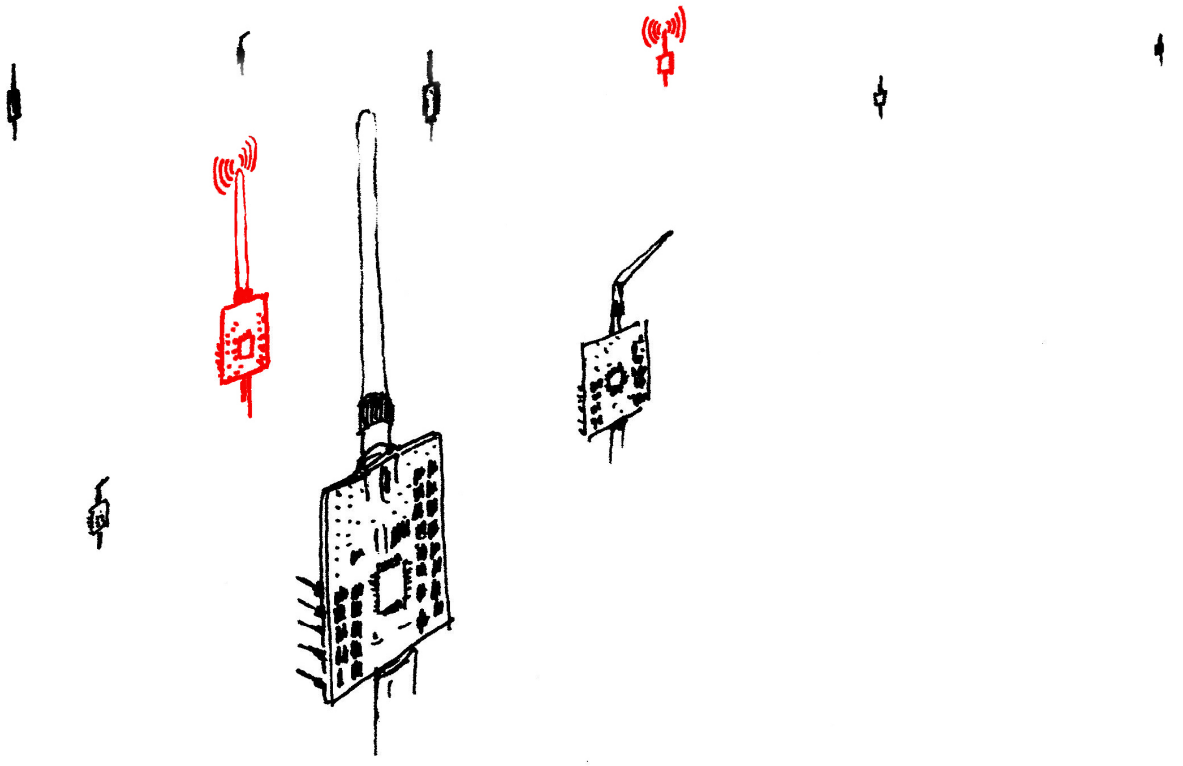
Tese de doutoramento em Engenharia Eletrotécnica e de Computadores, ramo de especialização em Computadores e Eletrónica, orientada pelo Professor Doutor Rui Pedro Duarte Cortesão, com a coorientação do Professor Doutor Jorge Afonso Cardoso Landeck e apresentada ao Departamento de Engenharia Eletrotécnica e de Computadores da Faculdade de Ciências e Tecnologia da Universidade de Coimbra

junho / 2015

· U C ·

UNIVERSIDADE DE COIMBRA

· U C ·
FCTUC FACULDADE DE CIÊNCIAS
E TECNOLOGIA
UNIVERSIDADE DE COIMBRA



José Pedro de Matos Nogueira Amaro

Wireless Sensor Networks Running on Harvested Energy

Ph.D Thesis

Advisor: Prof. Rui Pedro Duarte Cortesão

Co-advisor: Prof. Jorge Afonso Cardoso Landeck

June 2015

Cover image by ***José Plácido***

"Em memória de minha mãe, Fortunata."

Abstract

Ubiquitous computing is a concept in which data processing and information management are integrated within everyday objects and activities. Wireless Sensor Networks (WSN) naturally implement this concept, having the potential to become a widespread technology, with multiple applications, such as active clothing and large area environmental monitoring. Data acquisition and processing using a WSN is becoming an efficient tool to help systems and users with on-line decisions. However, despite all efforts from academia and industry, deploying, commissioning and operating these systems are time and manpower consuming tasks. The overall aim of the present work is to investigate and develop solutions for a network of wireless sensors with a large number of nodes that is able to achieve low deployment, maintenance and operational costs. Within this context, the power supply is one of the main constraints when operating WSN. The most obvious energy source for a WSN node is a battery. Batteries seriously limit WSN usage and are associated to the increasing cost of large networks. Low cost networks may be achieved when battery use is substantially reduced or eliminated. This consideration may also be relevant in environments with electrical power grid availability. Supplying WSN devices with mains power is not always an advantage, since node installation may require complex and time consuming actions. In this work, a solution that addresses WSN low cost implementation and operation has been designed, implemented and tested. The proposed solution is aimed at electrical energy monitoring of large buildings which, in modern economies, are responsible for an important part of the global electrical energy consumption. Moreover, the implementation of energy saving strategies may benefit from disaggregated data monitoring. To tackle this problem, a number of technological solutions exist, that are however, expensive in terms of equipment, installation and maintenance. Monitoring a building with such a network offers a clear advantage over wired systems. The development of a self-powered, battery-free current sensor node for large wireless sensor networks

may contribute to the implementation of monitoring solutions for large buildings. The proposed system is able to harvest energy from the environment and is easily installed by non-technical personnel. A solution to monitor disaggregated consumption has also been designed and implemented. Based on a contact-less power source with a Split-Core Current Transformer (SCCT), the proposed solution is able to run a complex WSN protocol. The proposed device is able to power a battery-free wireless node, estimating also the current drawn by the electrical load with a single SCCT. Low power operation has been achieved by adapting a complex protocol implementation to a battery-free device. The proposed device is able to identify data as well as to operate within specific protocol rules. With this device, nodes execute bidirectional communications and are compliant with the IEEE 802.15.4/Zigbee standards. Nodes listen for occupied radio carrier before transmitting and are able to transfer the standard regulated data payload. Protocol operations such as registering with the network or joining a group of nodes within the network have been implemented.

Keywords Wireless Sensor Networks, ZigBee Protocol, Energy Harvesting, Large Networks.

Resumo

O conceito de computação ubíqua descreve a capacidade dos objetos efetuarem processamento, gestão da informação e comunicação de dados, tornando-os invisíveis para as atividades dos indivíduos. As Redes de Sensores sem Fios (RSsF) interpretam e implementam naturalmente este conceito. O potencial de utilização de RSsF em múltiplos ambientes é enorme. Estes sistemas podem ser utilizados em cenários tão diferentes como roupas inteligentes ou monitorização ambiental. Utilizar uma RSsF para recolher informação, pode tornar-se uma forma muito eficiente para os utilizadores e os sistemas tomarem decisões. Uma RSsF poderá constituir uma clara vantagem sobre alguns sistemas de ligações com fios. No entanto a colocação no terreno, o início de operação e a manutenção, são processos muito dispendiosos em termos de mão-de-obra. O objetivo deste trabalho é o de contribuir para o desenvolvimento de redes de sensores sem fios com um grande número de nós, com baixos custos de colocação, operacão e manutenção. Neste contexto, o fornecimento de energia é um dos principais constrangimentos ao funcionamento das RSsF. As baterias são a fonte de energia mais óbvia para um nó de uma RSsF. No entanto, as baterias constituem uma séria limitação à operação destes sistemas e são associadas ao aumento do custo das grandes redes. A eliminação das baterias ou a redução da sua utilização poderá contribuir para a descida dos custos de instalação e operação destas redes. A substituição das baterias poderá ser relevante mesmo em cenários com disponibilidade de utilização de energia da rede elétrica. A instalação de um dispositivo ligado à rede elétrica pode exigir um conjunto de ações complexas e prolongadas. A solução proposta é dirigida à monitorização da rede elétrica de grandes edifícios que, nas economias modernas, são responsáveis por uma parte importante do consumo de energia. A implementação de estratégias de poupança de energia pode beneficiar da monitorização de dados desagregados. Para fazer face a este problema, existem uma série de soluções tecnológicas, que são no entanto dispendiosas em termos de equipamento, instalação e manutenção. A monito-

rização de um grande edifício com uma RSsF pode apresentar uma clara vantagem sobre os sistemas com fios. Neste contexto, a implementação de um dispositivo sem baterias e autoalimentado pode contribuir para a implementação de soluções de monitorizacão de consumos de energia em grandes edifícios. O sistema proposto é capaz de captar a energia da rede elétrica sem necessidade de contacto físico e é facilmente instalado por pessoal não técnico. O protótipo desenvolvido permite ainda monitorizar o consumo desagregado de cargas numa rede elétrica. O dispositivo possui uma fonte de energia sem contacto utilizando um sensor transformador de corrente, executando um protocolo complexo de comunicação. O nó implementado é compatível com as normas IEEE 802.15.4/Zigbee e a sua operação foi adaptada a este protocolo complexo. A implementação do dispositivo proposto é capaz de identificar os dados que lhe são dirigidos, bem como relacionar os dados à sua origem operando dentro de uma topologia de rede. O dispositivo executa igualmente comunicações bidirecionais em conformidade com as normas referidas.

Palavras-chave Redes de Sensores sem Fios, Protocolo ZigBee, Recolha de energia do meio ambiente, Redes com elevado número de nós.

Agradecimentos

O processo que culminou na elaboração desta dissertação foi um dos desafios mais exigentes da minha vida profissional. O volume de trabalho exigido pelos trabalhos de tese, acumulado com as responsabilidades já estabelecidas na minha vida pessoal e profissional, revelaram-se formidáveis obstáculos. Este trabalho foi concluído com o apoio de um conjunto de pessoas sem os quais não acredito que tal tivesse sido possível. Quero agradecer em primeiro lugar ao meu orientador, o Professor Rui Cortesão. O seu apoio e incentivo revelaram-se cruciais em momentos fundamentais do trabalho. Ao longo destes anos, o Professor Rui Cortesão foi um orientador exigente e meticuloso que moldou de forma fundamental o trabalho apresentado. Quero também expressar o meu agradecimento ao Professor Jorge Landeck pela contribuição na definição das características do sistema proposto. Agradeço igualmente ao Professor Fernando Ferreira pelas conversas e pontos de vista que fomos partilhando ao longo do trabalho. Uma palavra para o Paulo Santos que, como diretor de desenvolvimento da ISA, fez a primeira sugestão para um trabalho inovador. Aos meus colegas no Instituto Superior de Engenharia de Coimbra, em particular ao Inácio Fonseca e à Fernanda Coutinho que sempre se adaptaram às minhas rotinas pouco convencionais. Agradeço à minha família: À Florinda, que suportou as minhas insónias e a minha incapacidade para programar horários e calendários familiares. Agradeço também à Margarida, à Mafalda e à Joana que se habituaram às minhas demoras, atrasos e ao pouco tempo que sobrou para elas ao longo destes anos. Elas são a minha principal motivação e a fonte da minha vontade em terminar este trabalho. Agradeço ainda ao meu pai que foi mantendo em andamento aquela parte da minha vida que foi interrompida por força deste trabalho de doutoramento.

Este trabalho foi parcialmente financiado pela Fundação para a Ciência e Tecnologia (FCT) através da bolsa de doutoramento Nº SFRH/BD/45602/2008.

José Pedro Amaro

junho 2015

List of Acronyms

AC Alternating Current

AES Advanced Encryption Standard

AMRI Advanced Meter Reading Infrastructure

AODV Ad-Hoc On-Demand Distance Vector

API Application Programming Interface

APL Application Layer

ARM Advanced Risc Machine

BLE Bluetooth Low Energy

CAP Contention Access Period

CCA Clear Channel Assessment

CFP Contention Free Period

CPS Cyber Physical Systems

CSMA-CA Carrier Sense, Multiple Access with Collision Avoidance

COTS Components Off The Shelf

DC Direct Current

EP Endpoint

FFD Full Function Devices

GPS Global Positioning System

GSM Groupe Spécial Mobile

GTS Guaranteed Time Slot

IDE Integrated Development Environment

IEEE Institute of Electrical and Electronics Engineers

ISM Industrial, Scientific and Medical

IoT Internet of Things

IR Infrared

Kbps Kilobit per second

MANET Mobile Ad-hoc Networks

MAC Medium Access Control

MEMS Micro-Electro-Mechanical Systems

MCU Microcontroller Unit

OS Operating Systems

OSI Open Systems Interconnection

OUI Organizationally Unique Identifier

PAN Personal Area Network

QoS Quality of Service

SoC System-on-Chip

SCCT Split-Core Current Transformer

SPI Serial Peripheral Interface

SPICE Simulation Program with Integrated Circuit Emphasis

RF Radio Frequency

RMS Root Mean Square

RFD Reduced Function Devices

RERR Route Error

RREP Route Repply

RREQ Route Request

TDMA Time Division Multiple Access

TI Texas Instruments

UbC Ubiquitous Computing

UTC Coordinated Universal Time

WPAN Wireless Personal Area Networks

WSN Wireless Sensor Networks

ZC Zigbee Coordinator

ZCL Zigbee Cluster Library

ZDO Zigbee Device Object

ZDP Zigbee Device Profile

ZED Zigbee End Device

ZNP Zigbee Network Processor

ZR Zigbee Router

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The growing demand for communications equipment is one of the main driving forces of the semiconductor industry. Within this context, communications are increasingly implemented with devices that are built with Micro-Electro-Mechanical Systems (MEMS) technologies. The MEMS [1, 2] technology may be described as the group of techniques that allows analog, digital and radio frequency circuits to be included in the same silicon die. When processing capabilities are also included within these systems, they may be addressed as System-on-Chip (SoC). These SoC technological developments with multi-functional, low cost, low power, and small size systems enable the implementation of the Ubiquitous Computing (UbC) concept.

First introduced by Mark Weiser [3] in its classical paper "The Computer for the Twenty-First Century" [4], Ubiquitous Computing is described as embedding computers within people's lives. Mark Weiser envisioned that computers will *disappear* from eye site, and become embedded within the objects, in the same way electrical motors recently did. Ubiquitous Computing is related to computer controlled environments in which objects, being mobile or not, have sensory, computational and communication capabilities. Objects can self organize in a network so that, operating as a single system, can form a perception of the environment. These networks with sensory capabilities, that result from the collaborative effort of a large number of nodes, stand for a significant evolution when compared to traditional data sensing systems [5]. In [5] two traditional sensing methodologies are identified. Either sensors are placed away from the events they try to measure, or sensors without processing capabilities are placed within the sensor field, having their data transferred to a central processing unit. In either case, systems do not have a common and

cooperating perception of monitored events.

Taxonomy proposals [6] include the Mobile Ad-hoc Networks (MANET), Wireless Sensor Networks (WSN) and Cyber Physical Systems (CPS) concepts to describe different functionalities for UbC. The concept, may also be associated with the easily understandable idea of the Internet of Things (IoT) [7, 8]. Wireless Sensor Networks [9, 10], in which a number of devices with sensing, computational and radio frequency capabilities can self organize to collect data, are one solution to implement the concept of an Ubiquitous Computing Environment. Figure 1.1 illustrates the concept of the wireless connection between devices that form a system to extend the capabilities of a single isolated node.

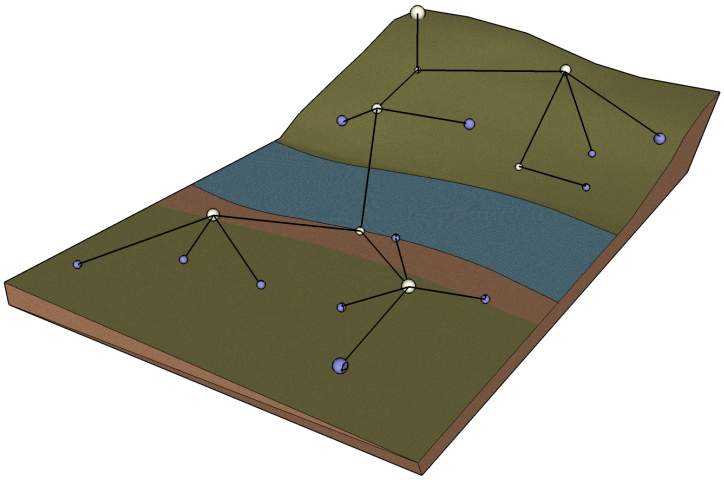


Figure 1.1: Deployed Wireless Sensor Network illustrating the nodes and their connections.

A number of Wireless Sensor Network applications have been presented in literature. Experiments with temperature, acoustic, magnetic, infrared and vibrations sensors among others, have been presented. In [11, 12, 13, 14] wild life and agriculture development are monitored with wireless sensor networks. Also home automation systems can be developed and implemented using wireless networks. In this case, intrusion, fire, flooding and gas alarms have been presented. Also domestic appliances can be provided with communication and processing capabilities [15, 16]. Patient monitoring in hospitals or even in home environment will be a reality in a near future. In [17], a heart rate monitoring system is presented with the obvious advantage of wireless communications dispensing clumsy wired systems. The military institution is also paying special attention to wireless networks. In [18] and [19] two such applications are presented. A sniper detection

system based on wireless sensors with tight signal synchronization is presented in [18]. This acoustic application of sensors determines a sniper location by measuring the sound travel time and comparing the different values obtained by the nodes. Also in [19] an auto regenerative mine field is presented. Achieving wireless networks full potential of development is a challenge to both the scientific and industry communities.

The paradigm states that these networks may have a large number of nodes, and that each node is able to communicate with neighbors within radio range. The nodes are randomly placed with variable space density and should be able to self organize. Data should be routed to the required destination. In some cases, wireless networks are to be placed in environments adverse to the human presence. They should therefore be able to work unattended, recovering from node loss or malfunctioning. All these functionalities should be achieved with minimum power consumption and long operation times. Nodes are placed in isolated areas communicating by radio frequency links. They have limited power supplies that can, at this time, be described as batteries. Isolated nodes cannot be maintained or repaired. Namely on energy source depletion, they become useless and are referred to as being dead. A dead node can affect the network normal behavior, therefore nodes must manage their power source in a criterious way, so that a prolonged life time is achieved.

The initial paradigm is yet to be implemented. In [20], a number of relevant guidelines are presented to implement a 400+ nodes using IEEE 802.15.4/Zigbee and Texas Instruments (TI) microcontrollers. A different approach is presented in [21] where a survey of real-world WSN implementations is presented. Even though the main purpose of that work is to determine the main characteristics an operating system for WSN must present, a number of implemented networks has been identified. The largest identified network proposes 180 nodes [22], while the maximum achieved lifetime of 1825 days [23] with a second lifetime of 625 days [24]. Also a number of challenges with implementation, deployment and configuration are reported in [21].

1.1 Motivation

The energy source is the main constraint when operating WSNs [9, 8]. The most obvious energy source for a WSN node is a battery, being these rechargeable devices or non-rechargeable ones (i.e. primary or secondary). However, batteries have a number of drawbacks that seriously limit the usage of WSN, and are in fact the main limitation to

its widespread use. The WSN implementation/operation cost can be significantly reduced if node battery utilization is substantially reduced or eliminated. Energy scavenging, also known as energy harvesting, is the alternative to battery use. A popular example of a scavenger power system is the Seiko Kinetic, the famous wrist watches fed by arm movement. A number of operational consequences result from a dead node. Namely the need for battery replacement and/or network reconfiguration may entail added cost to WSN operations. A simple calculation can be done: Considering the deployment of a 500 node network, with 10 €/hour labor cost and 5 minutes for the actual deployment and commissioning process of each node, entails an approximate total cost of 417 €. Considering the node cost determined in [25], the same 500 nodes cost less than the actual deployment and commissioning process. The same reasoning can be done regarding maintenance costs that are mainly related to battery replacement in battery operated nodes. It is easy to imagine that large network prices grow with the number of nodes, generating prohibitive costs and unacceptable return of investment periods.

Energy harvesting may be an important asset for a wireless network despite in-loco power availability. In these cases, there is plenty of available energy supply, but installing a node may require wire cutting and/or screw fastening. These considerations may be extended to industrial environments. In the high competitiveness of worldwide large modern industrial plants, maintenance tasks are critical to implement overall reliability and effective resource management systems. For instance, when using a large number of three-phase squirrel-cage induction motors, the implementation of integrated motor monitoring and diagnosis systems is highly desirable [26]. Moreover, maintenance tasks are frequently executed without stopping machine operation and thus performed under high power load. Connecting a WSN node to a power line in industrial environments, requires therefore a skilled technician to execute hazardous operations. Home automation provides a different scenario with similar considerations, if one considers the need for distributed energy consumption monitoring in large commercial buildings for energy management purposes. In both scenarios, monitoring a large number of electrical charges may require the installation of a significantly large number of wireless devices. Again in these cases, despite local power availability, the use of devices with no need for maintenance may decrease operation costs.

Together with the energy source, WSN protocol operation is a fundamental issue. Data must be identified, network traffic must be routed and nodes must coordinate with each other. Nodes must follow a protocol to be part of the network and are required to execute

bidirectional communications. Protocol operations such as registering with the network or joining to a group of nodes within the network are required. Moreover nodes must be able to listen before transmitting thus requiring more energy than if they were only sending a small number of bytes. A communications protocol requires energy availability that is strongly correlated with its complexity.

WSNs, whose nodes can easily be moved require no skilled workers for node placement and replacement, present a significant cost reduction of network operation. This thesis addresses the problem of reducing network operational costs. These costs are mainly addressed by replacing batteries with an energy harvesting solution, without losing operational functionalities. Also installation costs are addressed by using easily installable devices. Network protocol operation is investigated within the context of stringent short term energy constraints. Moreover, the use of energy harvesting solutions entails a significant shift from long term energy limitations to short term operational constraints. These solutions have been used to implement a device that is able to operated under the described constraints. The home automation scenario is addressed by the implementation of a solution that is able to estimate disaggregated current consumption in electrical loads.

1.2 Contributions

A device that includes a power manager, a harvesting circuit and a communication interface has been designed, tested and implemented. The proposed device is a WSN node that runs a communication protocol and is exclusively powered on harvested energy. Within the context of a large building electrical installation, an algorithm has been implemented to estimate mains current using only one Split-Core Current Transformer (SCCT), that at the same time is the nodes' single energy source. The estimation is implemented while maintaining the operational constraints posed by the scavenged energy and the use of an interactive communications protocol. The research in this thesis has generated the following publications:

- Journal publications
  - J. P. Amaro, Rui Cortesão, Fernando J.T.E. Ferreira and Jorge Landek, "Harvested Power Wireless Sensor Network Solution for Disaggregated Current Estimation in Large Buildings", Instrumentation and Measurement, IEEE Transactions on, PP(99): 1 - 1, January 2015. [27].

  - J. P. Amaro, Rui Cortesão, Fernando J.T.E. Ferreira and Jorge Landek, "Device and Operation Mechanism for Non-Beacon IEEE802.15.4/Zigbee Nodes Running on Harvested Energy", Elsevier Ad-Hoc Networks Journal, 26(0): 50 - 68, March 2015. [28].

- International Conference publications

  - J. P. Amaro, Rui Cortesão, Fernando J.T.E. Ferreira and Jorge Landeck, "Energy Harvesting for Zigbee Compliant Wireless Sensor Network Nodes", IECON, 38th Annual Conference of the IEEE Industrial Electronics Society. November 2012, Montreal, Canada [29].
  - J. P. Amaro, Rui Cortesão, Fernando J.T.E. Ferreira and Jorge Landek, "Powering Wireless Sensor Networks Nodes for Complex Protocols on Harvested Energy", Procedia Technology, Volume 5, Pages 518 - 526, CENTERIS, 4th Conference of ENTERprise Information Systems - Aligning technology, organizations and people. October 2012, Vilamoura, Portugal [30].
  - J. P. Amaro, Rui Cortesão, Fernando J.T.E. Ferreira and Jorge Landek, "Infield operation monitoring of induction motors using wireless modules running on harvested power", IECON, 37th Annual Conference of the IEEE Industrial Electronics Society. November 2011, Melbourne, Australia [31].
  - J. P. Amaro, Rui Cortesão, Fernando J.T.E. Ferreira and Jorge Landek, "A study on energy harvesting for wireless sensor networks using a split-core Current transformer", CETC, Conference on Electronics Telecommunications and Computers. December 2011, Lisbon, Portugal [32].
  - J. P. Amaro, Rui Cortesão, Fernando J.T.E. Ferreira, Jorge Landeck, N. Vinagre and R. Brás, "Low Cost Wireless Sensor Network for In-Field Operation Monitoring of Induction Motors", ICIT, IEEE International Conference on Industrial Technology. March 2010, Valparaiso, Chile [25].

1.3 Thesis Organization

The remainder of this document is organized as follows. In Chapter 2 a state of the art is presented. Hardware platforms, routing, medium access protocols and synchronization mechanisms are investigated. Operative tools such as operating systems and simulators

are described. Also battery and energy harvesting technologies are investigated and described. Due to the high relevance of IEEE 802.15.4/Zigbee for the proposed work, a description of the protocol is introduced in Chapter 3. The Zigbee protocol specifies a wireless technology based on the IEEE 802.15.4 standard [33] for Wireless Personal Area Networks (WPAN). To the purpose of this work the choice of this protocol is justified by the fact that if the system is able to operate using IEEE 802.15.4/Zigbee, a different protocol with a simpler implementation will also be able to operate. Chapter 4 presents the device that is able to run the IEEE 802.15.4/Zigbee from harvested energy. A magnetic power generator is used to build a contact-less energy harvesting system for a wireless module. The magnetic field generated by the alternating current flowing through a power line is harvested to power the proposed device. A SCCT is used to generate a current from the induced electromotive force. The toroidal coil is able to scavenge energy from power cables connecting electric charges (e.g. electric motors, lighting, computers). Also a simulator model that is able to predict system operation has been developed and is presented in Chapter 4. In Chapter 5 a solution is implemented with the IEEE 802.15.4/Zigbee software stack from TI. This solution, referenced as Z-Stack, is freely distributed by TI and is used to implement a number of commercially available devices. A number of changes to the original Z-Stack implementation are proposed so that a full IEEE 802.15.4/Zigbee compliant node is operated on harvested power. Changes to the original TI software are described. Moreover, a model that accounts for system operational and environmental parameters is proposed and analyzed. An analysis on the limitations and advantages of the proposed system is presented. An estimation technique for the mains current is discussed in Chapter 6. The single SCCT device scavenges energy from load power cables to power the IEEE 802.15.4/Zigbee node and is used to estimate the current that runs in the electrical wires. Conclusion and perspectives for future developments are presented in Chapter 7.

Chapter 2

State of The Art

A WSN is a system formed by a group of devices with radio frequency communication, computational and sensing/detection capabilities. The WSN technology, entails that three knowledge domains are addressed and tightly related. Hardware, software, and network architecture insights are mandatory requirements to implement one such system [34, 35]. Moreover, each one of these knowledge domains is usually considered to be an independent body of knowledge. All these design knowledge domains must be taken into account within the context of stringent energy resources, small form factor devices and severe memory constraints. A large number of contributions on WSN may be found in literature and industry [10]. The search of useful information within the proposed contributions may therefore be a difficult task. In this chapter, a review of this technology is presented. The presented review follows a methodology that starts by searching through the several surveys that are proposed in literature. These surveys are the starting point for the identification of relevant literature contributions.

In [36], WSNs are classified by their monitoring functions. Environmental and object monitoring, or object/environment interactions monitoring, are identified as the tree main uses of WSNs. An application based classification is presented in [5]. In that work, military, environmental, health care, building automation and other commercial applications are identified as applications for this technology. A different classification is presented in [9]. In this work, five service layers are identified in accordance with the Open Systems Interconnection (OSI) network model. In the OSI model, each layer represents a group of functions that provides data either upwards or downwards as needed in the layer stack. A function becomes part of a layer if it shares conceptual similarities with other layer func-

tions. The review presented in this thesis follows the OSI layer classification to organize the identified relevant proposals. In [9] these layers are described as:

- Physical layer
- Medium Access Control (MAC) layer
- Network layer
- Transport layer
- Application layer

In this description the MAC layer addresses the relevant functions of the Data Link OSI layer.

The hardware architecture addresses the physical considerations of the nodes. The Microcontroller Unit (MCU) choice, the required sensors, the actuators and the communication module must be interconnected to operate as desired. The network architecture refers to the node deployment methodology and the wireless communication protocol to be implemented. Operating system and middleware architecture address the software component of WSNs. Network communications are possible since two or more devices agree on a set of dispositions that are addressed in literature as a protocol. The defined protocol determines hardware specifications, medium access requirements, synchronization of communication events, data routing, security, and quality of service constraints.

2.1 Hardware Platforms for Wireless Sensor Networks

The MCU, transceiver, power source and sensor-elements are the common components of the hardware platforms for WSNs. The MCU controls and coordinates nodes operation, directs the transceiver, reads the sensors and commands the actuators. The transceiver is the wireless communication module that transmits and receives data packets using Radio Frequency (RF) or Infrared (IR). The power source is the energy supplier for the sensor node, which can be battery operated, Alternating Current (AC) mains powered, or use some form of energy harvesting (e.g. solar power). The sensors are the components that extract information from the environment and the actuators allow interaction with the environment. Most hardware platforms are implemented with Components Off The Shelf (COTS).

Initially developed at Berkeley University, René [37] is the first commercially available WSN node that has been built by the well known Crossbow company. Mica [37], Mica2[9, 38], mica2Dot[39], and MicaZ[9] were the next Crossbow's products that are currently a commercial product from MEMSIC, Inc [40]. These early platforms use an ATMEGA128L based processor from ATMEL [41] and are able to run TinyOs [42] operating system. Radio frequency modules are Industrial, Scientific and Medical (ISM) band (433MHz, 868, 902 MHz e 2,4GHz) compliant. From these motes, only MicaZ is IEEE 802.15.4 [43] compliant. MEMSIC Imote2.NET Edition[40], Imote2[40] and Sun Microsystems Sun SPOT [44] have been implemented with Advanced Risc Machine (ARM) processors and a CC2420 radio transceiver from Chipcon. These platforms represent a clear change to the initially presented Berkeley platforms. These modules break the link with the TinyOs operating system and don't use the ATMEGA microprocessor core. BTnode rev2 and BTnode rev3[45, 46] have been developed by the Zürich Federal Institute of Technology in Switzerland. They are based on an Atmel ATMEGA128L, their communication modules are a Bluetooth and a Chipcon CC1000. A TinyOs operating system is used on both nodes. FireFly [47] nodes use an IEEE 802.15.4 (Chipcon2420) compliant transceiver and the 8-bit MCU from Atmel ATMEGA128. A very particular characteristic is the use of an AM/FM radio receiver to periodically acquire a time synchronization pulse. A synchronization pulse may be emitted using an AM/FM carrier transmitter throughout all the installation. This synchronization pulse can naturally be transmitted by any radio station. FireFly runs a specially developed real-time sensor operating system named Nano-RK [48]. Fleck [49, 50], is supported by an Atmega128L microprocessor, uses a Nordic nRF903 radio frequency module and is able to run Berkeley TinyOS operating system.

Particles [51, 52] have been developed at the Technical University of Karlsruhe. Particles modules are a set of PCB[1] plug-in modules. These include sensor boards without microprocessor, communication and programming boards and microprocessor/radio frequency boards. All boards that have a MCU use PIC [53] MCUs. A full set of library code was developed. TinyNode 584 [54] uses a MSP430F1611 MCU from Texas Instruments and a Xemics XE1205 radio frequency at 433 MHz, 868MHz or 915 MHz ISM band operation.

All investigated platforms use one of the following MCUs and radio frequency modules:

Processors ARM7 or ARM9, Atmel AVR, Intel Xscale (ARM), Intel 8051, Microchip

[1] Printed Circuit Board

PIC or Texas Instruments MSP430.

Radio Modules Chipcon [55] [2] CC1000, Chipcon CC1020, Chipcon CC1021, Chipcon CC2420, Xemics, XE1205 or Zeevo ZV4002, Nordic nRF90E.

A different approach uses a common integrated circuit substrate with a MCU and the RF module. These SoC implement all functionalities in one single chip entailing large form size and cost reduction. The CC2530/1 [56] from Texas Instruments implements an 8-bit 8051 MCU with a CC2420 while the CC2538 [57] includes a 32-bit ARM7 MCU with the same radio module. Atmel SAM R21 [58] is a 32-bit ARM6 MCU that also includes a 2.4 GHz IEEE 802.15.4 wireless module. The ATMEGA series is also built with an included 2.4 GHz Transceiver with the reference ATmega128RFA [59]. Freescale supplies three families of devices with an included IEEE 802.15.4 transceiver: The 8-bit MC1320x[60] and the 8-bit MC1321x[60] families are based on the HC9S08 architecture and the MC1322x [61] family with a 32-bit ARM7 MCU core. Silicon Labs implements the EM35x [62] and EM358x [62] chips with 32-bit ARM7 MCU that include 2.4 GHz IEEE 802.15.4 radio transceivers. ST Microelectronics presents the STM32W108 [63], which is a System-on-Chip with a 2.4 GHz, IEEE 802.15.4-compliant transceiver and a 32-bit ARM7 MCU.

A third architecture option is named network processor. Two MCUs are implemented with the radio module integrated in one of them or used as a different chip. In this case all communication protocol is addressed by one MCU and the radio while the second MCU addresses sensor/actuator control. The communication between the two MCUs is implemented with a Serial Peripheral Interface (SPI) and a dedicated Application Programming Interface (API). One example of a network processor architecture can be found in [64] where a MSP430 MCU is used to control a Zigbee node with a CC2530.

A remarkable resemblance may be identified between all investigated platforms. The center topics of wireless sensor networks explain device similarities. All nodes must present low power characteristics, low price and ISM band radio frequency communication capabilities. These characteristics are mandatory to implement long lasting, self organizing networks. Also, national and international regulations must be complied. These legal considerations are taken into account by radio frequency modules since they must operate in the ISM bands. ISM bands leave a number of frequencies open to free communications standards. Either scientific or commercial, all nodes present a processing unit with la-

[2] Chipcon was bought by Texas Instruments in 2006

tency capabilities and other low energy use techniques. All radio frequency units may be shutdown, while keeping the processor active or in low activity mode. Some modules have more than one communication port so that, besides ISM bands, infrared communication may also be used. Some important differences may be identified in the nodes memory size. This can be an important characteristic if one considers the high memory usage of some wireless protocols. All platforms have wired communication capabilities such as SPI, I2C and RS232. Also, analog to digital conversion circuitry is available in all nodes. Table 2.1 summarizes the main characteristics of the dual chip described modules. Table 2.2 describes the characteristics of the investigated SoC modules. To the author knowledge, in [65] the most comprehensive list of modules may be found.

Table 2.1: Relevant characteristics for WSN dual chip hardware platforms.

Platform	Available memory flash/RAM	Communication modules	Software	802.15.4 compliant	Architecture	Low power	High power
René	8/0.5 KB	TR1000	-	no	AT90/RF module	1 μA	12 mA
Mica	128/4 KB	TR1000	TinyOS	no	ATMEGA/RF module	1 μA	12 mA
Mica2, mica2dot	128/4 KB	CC1000	TinyOs	no	ATMEGA/RF module	1 μA	26.7 mA
MicaZ	128/8 KB	CC2420	TinyOs	yes	ATMEGA/RF module	1 μA	18.8 mA
Imote2.NET Edition, Imote2	128/4 KB	CC2420	TinyOs	yes	ARM/RF module	390 μA	66 mA
Sun SPOT	4000/512 KB	CC2420	Java	yes	ARM/RF module	36 μA	35 mA
BTnode rev2, BTnode rev3	128/180 KB	CC1000	TinyOs	no	ATMEGA /RF /BT[3]	1 μA	16.5 mA
FireFly	128/4 KB	CC2420	NanoRK	yes	ATMEGA/RF	0.2 μA	24.8 mA
Fleck 1,2,3	128/4 KB	nRF903	TinyOs	no	ATMEGA/RF	33 μA	12.5 mA
Particles	128/4 KB	TR1001	Smart-its	no	PIC18/RF	0.7 μA	12 mA
TinyNode 584	48/10 KB	XE1205	TinyOs	6LowPAN	MSP430/RF	4 μA	62 mA

2.2 Medium Access Control Protocols

The Medium Access Control (MAC) layer is responsible, among other functions, for data consistency check, error detection and amount of transferred data. Node operation for

Table 2.2: Relevant characteristics for WSN single chip hardware platforms.

Platform	Available memory flash/RAM	Communication modules	Software	Compliance	Architecture	Low power	High power
CC2530/1	256/8 KB	CC2520	Z-Stack	802.15.4	8051	1 μA	24 mA
CC2538	512/32 KB	CC2520	Z-Stack	802.15.4	ARM	1.3 μA	24 mA
CC2540/1	256/8 KB	CC2520	BLE Stack	Bluetooth	8051	1 μA	24 mA
ATSAM R21	256/32 KB	AT86RF233	BitCloud	802.15.4	ARM	4.06 μA	13.8 mA
ATMEGA256RF	256/32 KB	AT86RF233	BitCloud	802.15.4	ATMEGA	1.5 μA	18.6 mA
MC132xx	128/8 KB	Proprietary	BeeStack	802.15.4	HCS08	2.65 μA	34 mA
EM35xx	512/64 KB	Proprietary	EmberZNet	802.15.4	ARM	1.25 μA	31 mA
STM32W108	256/16 KB	Proprietary	ST RF4CE	802.15.4	ARM	0.8 μA	31 mA

medium access has to deal with a number of challenges that result from failure prone radio links, hardware limitations (e.g. oscillator inaccuracy), event-driven traffic characteristics and node density. In the context of the MAC layer, the main causes for power consumption, other than the desired data transmission, are the following [66]: Collisions due to one or more simultaneous transmission attempts entails that nodes may be required to resend a packet. The packet retransmission is an energy consuming task with impact on the data delivery time. Overhearing is characterized by nodes receiving packets that are destined to another node. Overhearing entails a large impact on stored energy mainly when a high node density exists; Protocol implementation requires that a number of non data octets are transferred. These packets are defined in literature as overheads and result in the increase of the useful per byte power consumption, since control messages don't usually contain data.

The hardware limitation that results from oscillator inaccuracy makes it impossible to synchronize nodes with the required precision. Due to this oscillator inaccuracy, a WSN node doesn't have an accurate measure of when to wake up to receive a packet. This drawback is handled by placing nodes in idle listening mode which consumes a significant amount of their stored energy. In this mode, data reception is implemented by the listening to the radio channel.

The investigation of MAC protocols is therefore an open research field since the perfect optimization is yet to be achieved. In the early years of WSN development, the research efforts where mainly placed on energy efficiency. Recently, however, the focus has been placed also in the Quality of Service (QoS) rather than exclusively on energy management.

In either case, in WSN where operation requires nodes to sleep, the main goal of MAC protocols is to turn off the transceiver whenever possible since it is the hardware component that consumes most of the energy. The nodes radio modules operate therefore in duty-cycled modes. The medium access in these duty-cycled operation mechanisms may be achieved with various algorithms.

In [67] a number of early MAC protocols are reviewed and evaluated. Table 2.3 obtained in [67] makes a relevant comparison between a number of these early mechanisms. In [66] a follow-up on these protocols is presented and a taxonomy proposed. MAC protocols are divided in four groups according to their operation mechanism. Fig 2.1, obtained from [66], illustrates this comprehensive list of MAC protocols. In non synchronous MAC protocols, data transmission is preceded by a large radio frequency preamble, so that all receivers are able to detect it. In a simple approach to this mechanism, all nodes have

Table 2.3: Early MAC protocols obtained from [67].

	Synchronizes	Medium Access	Adaptability
S-MAC TMAC DSMAC	No	CSMA	Good
WiseMAC	No	np-CSMA	Good
TRAMA	Yes	TDMA/CSMA	Good
SIFT	No	CSMA/CA	Good
DMAC	yes	TDMA/Slatted Aloha	Weak

the same channel sampling interval, thus ensuring that all potential receivers can detect the preamble and remain awake to receive the subsequent data frame. The drawback is that non-target nodes can only determine that a packet is not destined to them at the end of the preamble transmission [66]. Non-target nodes are therefore subject to unnecessary energy consumption. A number of protocol proposals address this overhearing problem by using a series of beacon like preambles instead of a single long one. These beacons may contain data that allows non-target nodes to determine data destination and thus returning to sleep mode. A receiver-initiated mechanism is implemented in several asynchronous MAC protocols to increase the amount of transmitted data, The receiver executes a data request being therefore the communication initiator. In this case, the radio channel is occupied by the sender only at the receiver request. The sender must, however, remain with the radio module turned on in receive mode. Several approaches implement dynamic

preamble length thus allowing nodes to reduce wakeup times.

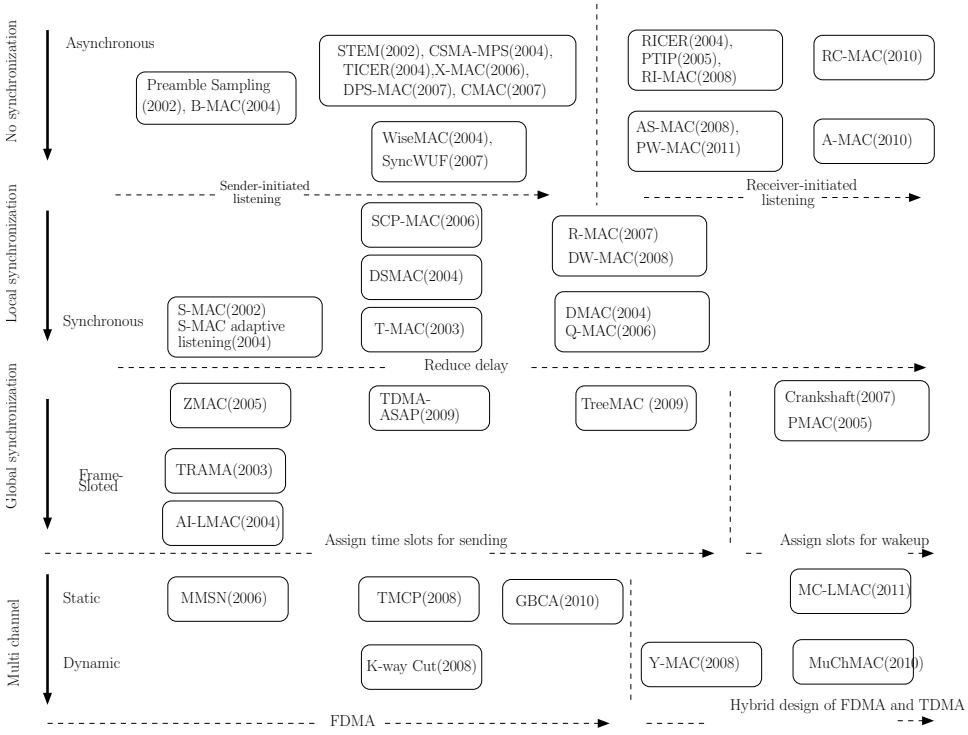


Figure 2.1: A taxonomy of WSN MAC protocols.[66].

Synchronous MAC protocols, operate by listening to the radio channel for a certain amount of time at a pre-determined moment. A node that coordinates the communications for a group is known as a synchronizer and broadcasts its schedule at a determined interval. A follower hears the synchronizer and determines its receiving schedule. All nodes within each cluster or group of nodes must be synchronized by a time coordinator. This coordinator may be placed at a distance of one or several radio hops away. Idle listening, overhearing, over-emitting and collisions are avoided by using this scheduling mechanism, according to which each node is programmed to wake up and listen for data at specified intervals. This approach requires nodes to be synchronized, thus increasing data transfer requirements so that they all share a common timetable. The communications are scheduled by the synchronizer node and commands such as Request-to-send or Clear-to-send may be implemented. Synchronous protocols allow nodes to be synchronized with more than a small group of nodes, thus implementing a global synchronized timetable of higher complexity.

A different mechanism is implemented with a frame-slotted derived from the Time Division Multiple Access (TDMA) mechanism. In frame-slotted MAC protocols, a common time frame may be used where time slots are reserved for each node. This frame division may be implemented with a number of variations and adapted to a network topology. In multichannel MAC protocols, parallel transmissions are allowed because different frequency channels are implemented. These protocols require, however, more sophisticated radio modules that are able to effectively use radio spectrum. The use of these protocols entails a number of complex details, such as channel allocation and the need for cross-channel communications.

A number of international standards implement MAC protocols. The IEEE 802.15.4/Zigbee is discussed in the next chapter. WirelessHart [68] for process control, and ISA-100.11a [69] for industrial applications present their MAC protocols. ISA-100.11a is flexible because it provides more configurable parameters than WirelessHART [70, 66]. In WirelessHART, a 10 ms fixed time slot size is used while in ISA100.11a, a configurable mechanism based on a superframe is implemented. A frequency hopping mechanism is implemented in both standards that use TDMA for channel access. Both standards implement RF interference avoid mechanisms for industrial environments. IEEE 802.15.4 defines 16 channels (channels 11-26) in the 2.4 GHz band. WirelessHART uses channels 11-25 because channel 26 cannot be legally used in several countries [66]. Channel 26 is optional in ISA-100.11a. A slotted hopping scheme between the radio channels has been adopted by WirelessHART. In this mechanism the communications in each time slot is executed in a different radio channel.

Three radio channel hopping schemes are defined in ISA100.11a. Radio channel may be changed as in WirelessHart by hopping in each slot but also a slow hopping mechanism may be used. The slow time hopping mechanism entails that nodes maintain the radio channel for several contiguous time slots before communication channel changing. Network join procedure is implemented in ISA100.11a by taking advantage the slow hopping mechanism. Also devices with weak time synchronization characteristics or whose data transfers are event-based may easily transfer data with the slow hopping mechanism. A slow hopping period is shared by a group of nodes that transfer data using a Carrier Sense, Multiple Access with Collision Avoidance (CSMA-CA) scheme. Each device trigged event entails an immediate packet transmission thus reducing time latencies when sending an alarm or other urgent message. The drawback of the slow hopping mechanism is the increased energy consumption because nodes are required to listen for the

radio channel while waiting for incoming data. A hybrid hopping mechanism is therefore proposed. This mechanism, combines slotted and slow hopping, where slotted hopping addresses predictable (e.g. periodical) data transfers, while slow hopping may be used to timely receive randomly generated messages (e.g alarms). In WirelessHART, the hopping mechanism is implemented by the network coordinator, while in ISA100.11a, implements five protocol defined hopping patterns. In addition, both WirelessHART and ISA-100.11a may prevent the use of noisy channels by blocking the hop to those with detected radio collisions. The strong frequency hopping schemes and the flexibility in the synchronization mechanisms allows characterization of WirelessHART and ISA-100.11a as protocols for industrial applications [70, 66].

2.3 Node Synchronization Protocols

One very important consideration for wireless networks is node synchronization. Due to fundamental energy constraints, battery powered nodes must reduce power consumption by shutting down the main current sinks. The most obvious candidate for shutdown is the radio frequency module, even though other microprocessor functions can also be placed in dormant state. In fact, all system functions like humidity or smoke sensors can be placed in low power mode. The drawback of shutting down modules, namely the radio module, is the inability to communicate with the outside world, making the node effectively absent from the network. The return from this latency mode requires a tight control over node synchronization. An unsynchronized node can find itself with no one to exchange data with. It is therefore of critical importance to be able to wake up nodes at the same time - that is to synchronize them.

Node time tracking is achieved by counting oscillations of an electronic device referred to as crystal or oscillator. The time in a microprocessor is implemented as a counter that is incremented with one or more oscillators and is referred to as a software clock. Throughout this work, unless otherwise noted, the software clock is simply referred to as clock. However, building oscillators with absolute time synchronization is not possible due to manufacturing differences, various physical effects, temperature and crystal aging. A consequence may be illustrated by the fact that nodes with unsynchronized clocks may cause the same event to be observed at different time stamps.

There are two main synchronization strategies for wireless networks. Measuring relative time may be used when loose timing constraints are enough. This mechanism relies

on message and event ordering. The idea is to be able to determine event precedence, namely if one event occurred before another. Comparing local clocks, for example at a sink node, is enough to determine event order. In some systems this is the only implemented synchronization method, namely if no duty-cycle operation is required. If events reported from different nodes must be characterized by their original occurrence, nodes must share a common global schedule. This is also true if nodes operate in duty-cycle mode as discussed in the previous section. For this operation mode, the drift and offset of the independent clocks of each node must be tracked in correspondence with neighboring nodes.

An ideal clock oscillator shows a constant frequency with a duty-cycle of 50%. The clock frequency varies unpredictably causing an effect named as drifting. A different phenomenon that causes time measurement errors is known as clock skew which is characterized by changes in the duty-cycle. The real clock value is approximated in literature by using estimators; two approaches have been proposed [71, 72]: The offset-only and the joint skew-offset model. If $C(t)$ is the function that describes a clock signal at instant t, then

$$C_i(t) = t + \theta_i \tag{2.1}$$

is the approximated clock function of a node i, where θ_i is the offset of the i^{th} nodes clock to an ideal clock [72]. The clock of two nodes may be related by

$$C_j(t) = C_i(t) + \theta_{n_i \to n_j} \tag{2.2}$$

where $\theta_{n_i \to n_j}$ is the relative offset between nodes n_i and n_j clocks. This simple model, that doesn't account for drifting, requires frequent synchronizing messages to be exchanged. A more complex model is described by

$$C_i(t) = \alpha_i t + \beta_i \tag{2.3}$$

where α_i is the absolute skew and β_i is the offset of the i^{th} nodes clock, respectively. Similarly, relative clocks of two nodes C_i, C_j may also be related by

$$C_j(t) = \alpha_{n_i \to n_j} C_i(t) + \beta_{n_i \to n_j} \tag{2.4}$$

where $\alpha_{n_i \to n_j}$ and $\beta_{n_i \to n_j}$ stand for the relative offset and skew between the two nodes. The relative offset and skew between two nodes must therefore be computed and shared

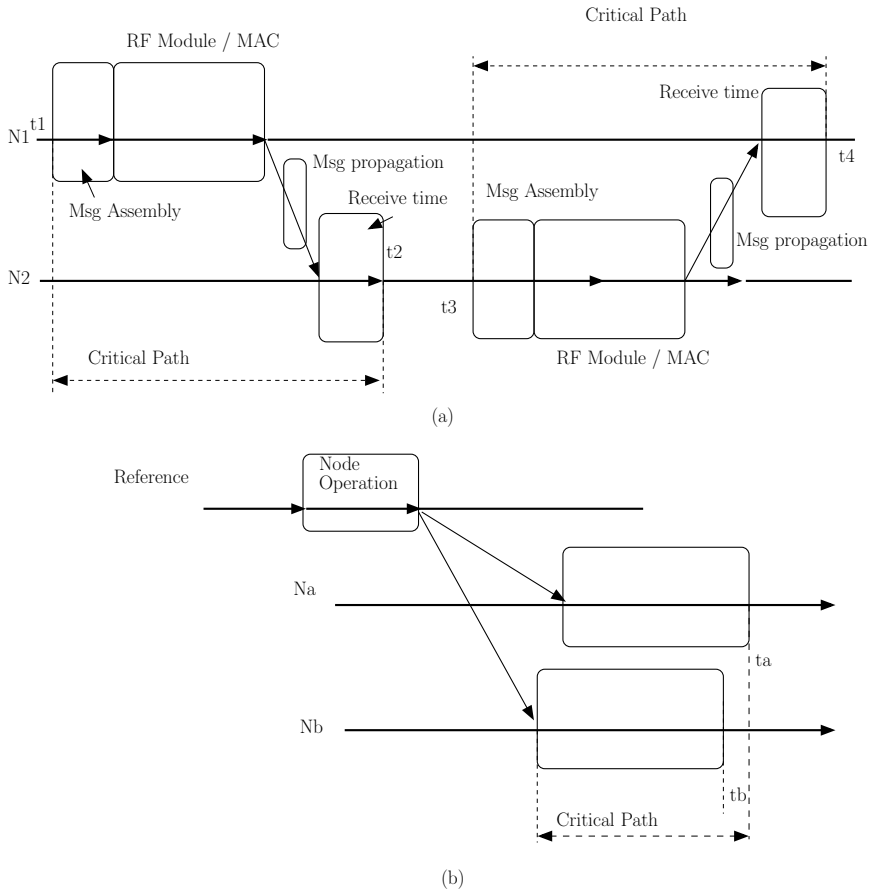


Figure 2.2: Receiver-to-receiver (b) vs sender-to-receiver (a) share synchronization parameters mechanisms.[72].

by all synchronous nodes. Fig. 2.2 [73] illustrates the two mechanisms that are proposed in literature to compute (2.4). In a receiver-to-receiver mechanism, as shown in Fig 2.2 (b), a reference message is broadcasted and only t_a and t_b are shared between receiver nodes. The sender-to-receiver mechanism (Fig.2.2 (a)) shares (t_1, t_2, t_3, t_4) to compute the relative clock offset and skew of the nodes. To compute the synchronization parameters, a number of implementation challenges have been identified. To achieve nodes synchronization, a communication channel must exist so that data from (2.4) may be shared. This communication between nodes presents delays that must be reliably estimated. The challenge of accurate network clock synchronization arises from the non-deterministic nature of the delay estimation process. Errors in estimation of channel latencies are introduced by random events which cause delays in synchronization messages delivery. Channel latency

is characterized by four distinct components as illustrated in Fig 2.2 [73]. The send time is spent by the node assembling the message to be delivered to the radio module. This random inserted delay includes kernel processing, transfer to the network interface, and other operating system delays. The access time varies according to the RF module access mechanism and the MAC protocol and it is also non-deterministic. The propagation time is measured by the time the message takes from sender to receiver. This time may be very short in one hop communication, or may entail a random long period of time due to the delays introduced by the hop in each router, where queuing and switching delays may occur. The receiving time corresponds to the period the RF module takes to get the message delivered to the operating system.

Timestamps may be accounted at both the first and the fourth time steps where the operating system is able to detect events. In both cases the time stamps show less variability if they are obtained at the stack lower levels (e.g. the MAC layer). The communication latency introduces therefore errors to the clock estimation process. Some literature proposed mechanisms stamp their time at the MAC layer, but others choose the application layer where significant delays may be introduced. Other approach includes obtaining multiple time stamps at different layers. Moreover, after receiving the required parameters, nodes must compute (2.2) and (2.4) using estimation techniques or other numerical approaches.

In [72], a number of literature proposed synchronization protocols are tested and the behavior of implemented models discussed. The discussion is limited to those protocols that have been implemented in hardware. The authors claim that simulated only protocols fail to address a number of fundamental WSN node constraints. Node software limitations are apparent when the proposed synchronization algorithms are implemented after having been tested in simulated environments. These simulations rely on high-end computational-rich machines and ignore the fact that the proposed algorithms are to run on computational-limited platforms. In this case, complex calculus are possible at the cost of large memory footprints thus inserting latency in the time stamping implementation.

In Table 2.4 a list of proposed protocols that model offset only is presented. Table 2.5 presents a similar list for offset and skew compensated clock models. On both tables, relevant protocol characteristics are addressed. As discussed, model calculus varies according to the stack layer where the time stamping is implemented. It is therefore relevant that time accounting (time stamping) is done at the application level or at a lower layer. The most efficient time stamping within the investigated protocols is executed at the

Table 2.4: Offset-only synchronization protocols [72].

Protocol	Fast drifting	Delay variations	Fault-tolerance	Security
TPSN	No	MTS	No	No
Sec-TPSN	No	MTS	No	Yes
Harmonia	2 Clocks	No	No	No
Glossy	2 Clocks	NOP Insertion	No	No
Syntonistor	AC Power	MTS	No	No

Table 2.5: Offset-skew synchronization protocols [72].

Protocol	Fast drifting	Delay variations	Fault-tolerance	Security
RBS	ES	MTS+R2R	No	No
FTSP	ES	MTS+ST	No	No
GTSP	ES	MTS+ST	No	No
ATS	ES	No	No	No
RATS	ES	MTS	No	No
PulseSync	ES	ES+ST	No	No
RRTE	ES	No	No	No
CS-MMS	ES	MTS	No	No
R^4Syn	ES	MTS+R2R	Yes	No

MAC layer. This characteristic is denoted by MTS (Mac-layer time stamping) on both tables. Moreover, only one of these protocols addresses fault tolerance to synchronization messages propagation. The remaining proposals consider that synchronization messages are never lost. Moreover, security is addressed only in one of the studied protocols. A Receiver-to-receiver mechanism is denoted as R2R. Most hardware platforms include two clock sources: an internal high speed oscillator that is able to run in the MHz range; and a slow external clock source that is usually a 32.768 Hz crystal. The high speed oscillators usually present high drifting while the external oscillators are more stable and show lower resolution. For the synchronization techniques that exclusively model the clock offset, the fast drifting issue may be addressed by using the low resolution oscillator with no calculus involved. As an alternative, an external synchronization signal (e.g AC 50/60 Hz power line frequency or FM radio signal) may be used. Estimation techniques are implemented in

all mechanisms that address both offset and clock skew. The delay variations are tackled by controlling where within the stack the timestamp is obtained. A number of protocols also use the technique of several timestamps during the process (denoted ST). Fig. 2.3 illustrates a taxonomy for the investigated synchronization protocols.

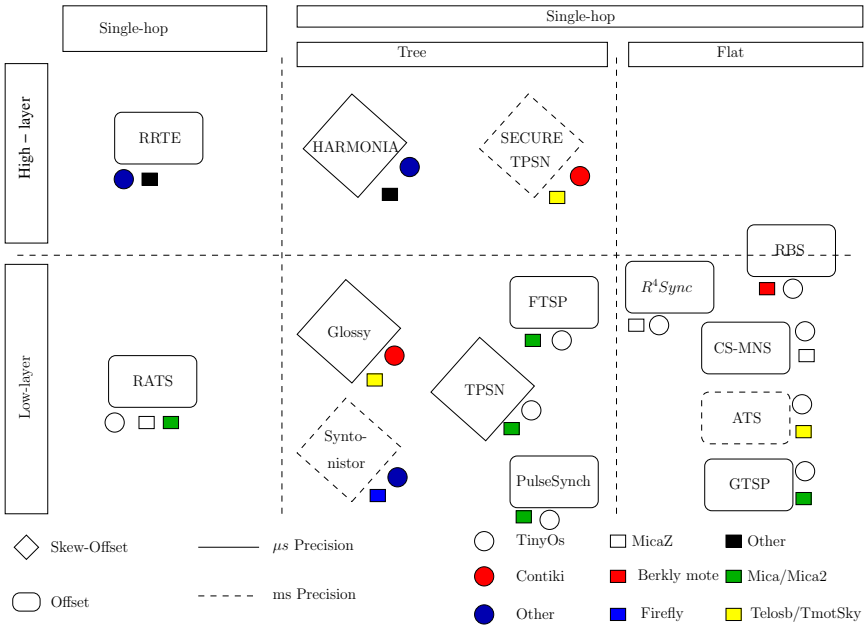


Figure 2.3: Synchronization mechanisms taxonomy.[72].

2.4 Data Routing

Data routing is defined as the group of tasks responsible for delivering data to its destination. These tasks are essential if more than one node stands between the data source and its destination. In the early classification, two groups of routing protocols have been considered: On-demand and table-driven protocols. A different classification proposal for early routing algorithms, establishes a division between centralized/distributed and proactive/reactive categories [74]. These early proposals also include a number of hybrid approaches between these groups. In [75], this categorization is referred to as the classical dimensions of routing algorithms. The classical dimension of routing protocols entails a number of considerations for classification purposes. Data-centric protocols are addressed as a solution for large networks where an individual addressing approach is not practical. When the destination node (sink node) queries for required data, attribute-based

routing schemes are proposed. In a WSN this means that different data can have diverse destinations and paths. Examples of this type of routing algorithms are Flooding and Gossiping [76] mechanisms. In Flooding, data from each node is sent to all its neighbors; each receiving node will then repeat the task to its own neighbors. Data will eventually arrive to the sink, but at the risk of implosion caused by duplicated messages. Gossiping prevents this implosion problem by sending data to only one neighbor. The drawback of these mechanisms is geographic overlapping of data readings. The SPIN [76] protocol uses a data negotiation/querying mechanism. In this case, high level data descriptors are used, and the data interest of the nodes is advertised to their neighbors. High level descriptors define, for example, an area or sensor type. Similarly to SPIN, Direct Diffusion [77, 78], that uses a gradient to establish a unique data-path. Direct Diffusion differs from SPIN in the way the query mechanism operates. In the latter, the queries are sink originated and determine if specific data is available, whereas in the former mechanism, sensors advertise data availability. Several variations of Direct Diffusion are also proposed in [79, 80, 81, 82]. In [79] authors propose that a number of less optimal routing paths are used to reduce energy cost of optimal paths. A path cost is calculated as the sum of costs of each hop in a routing path. In [80], a change to data flooding is proposed when a little amount is requested. A calculus of the number of hops needed to transfer data to the sink is proposed in [81, 83]. This way, the minimum number of hops to the sink can be determined by each node. The number of hops is known as height of the node. Moreover, within the group of data-centric protocols, Cougar [84] and Acquire [85] proposals consider the network as a large distributed database. Both these proposals require a very tight control over node synchronization.

Besides data-centric protocols, [86] distinguishes hierarchical protocols as those aiming to efficiently manage sensor energy consumption by restraining communications within a group named cluster. Each sensor belonging to a cluster only interacts with the cluster head that is, in turn, responsible for data fusion and forwarding to the rest of the network and eventually to a data sink. One example of a hierarchical protocol is LEACH [87]. LEACH nodes are grouped in clusters and each group elects a leader. The cluster head is chosen by power availability, being the node with the largest reserve. A cycle is implemented so that the cluster head changes, preventing one single node to be depleted and also allowing the energy reserves from the group to remain balanced. Several variations of LEACH have been proposed in PEGASIS [88], TEEN [89] and APTEEN [90]. These mechanisms differ in the way the cluster head is selected and in the way the cluster is

organized. The energy-aware routing for cluster based network protocol [91] is a three level topology node architecture where each level corresponds to a different type of node. Base level nodes are energy constrained that are capable of data sensing and relaying. Gateways are less energy constrained nodes that are able to organize clusters and forward data to the higher and third level command nodes.

In [92], a multi level protocol is presented. In this case, nodes must be at radio reach of a stationary router node. Router nodes form the backbone of the network. Node location is a useful information for node data routing because it allows a straightforward path to the sink, while keeping the remaining nodes in latency mode. The works presented in [93] and [94] use Global Positioning System (GPS) modules to obtain node position. In both works, nodes only send data to nearby neighbors allowing reduced energy consumption. The main drawback is the amount of energy consumption of the GPS module that is usually needed to obtain the nodes geographic location. The alternative of GPS is to manually configure the node location which may not be feasible in large networks. A balanced solution can be achieved if only some nodes have GPS modules, in which case location of nearby nodes can be obtained. In location-based routing mechanisms, node forwarding selection is based on greedy selection so that each node selects the node nearest to the sink to send data. An example of such mechanism is the Greedy Perimeter Stateless Routing (GPSR) [95] where greedy forwarding is used until a node doesn't have a node that is closer to the data sink. In this case, a hole in the network is identified and GPSR forwards data using a right-hand rule around the hole perimeter. Trajectory-Based Forwarding TBF [96] routes data along a predefined path. This path is expressed as a formula, and each node solves the formula to identify the forwarding node. A computational overhead is introduced, but it is compensated by the smaller amount of routing information data the protocol requires. In TEDD [97] the trajectory is defined according to global node remaining energy. In Mobicast [98] and Face Aware Routing FAR [99] target zones are defined instead of target nodes.

The specific challenges of the WSN development led to the proposal of techniques that implemented protocols such as Geographical and Hierarchical routing. Also, Power-aware routing protocols are addressed as a separate group with specific performance metrics. Figs. 2.4 and 2.5 illustrate this recently proposed taxonomy that describes WSN routing protocols functionalities. In [86, 74, 75] a number of routing protocols for WSN are classified according to this taxonomy. These authors considered a taxonomy that includes Multi-path, Geo-casting, Multicasting, Hybrid, Mesh and Flow-aware routing protocols.

The studied early protocols have also been added to this taxonomy.

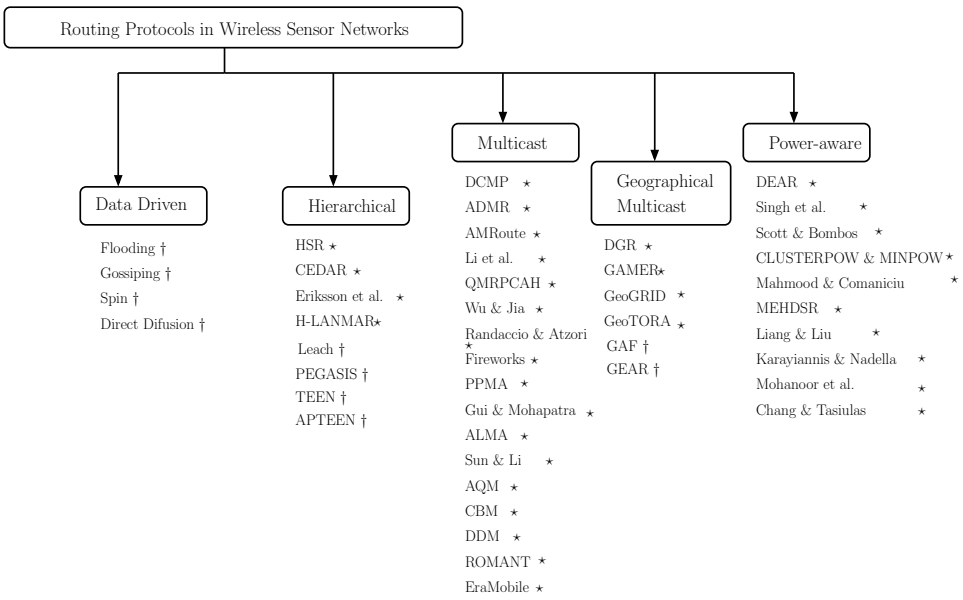


Figure 2.4: Routing algorithms list and taxonomy (part 1). Protocols identified with ⋆ are obtained from [75] and with † are obtained from [86].

The proposed taxonomy is based on a number of metrics that describe routing mechanisms. In table-driven routing approaches, the nodes implement a proactive mechanism where a table with next hop nodes is built with collected information. The collected information is exchanged among the other network routers. Update messages are periodically exchanged between routers thus maintaining updated routing tables at the routers. Whenever a node must send or forward a message, the routing table is consulted so that the next hop address is known. The easy access to the routing table allows message forwarding to be a fast process. This process requires however periodical route updates to all reachable nodes. The updates are processed by broadcasting messages between routers with the new routing data. Route maintenance requires therefore several overhead communications thus entailing the use of a large amount of bandwidth. Moreover, this overhead traffic is a continuous process even in the absence of data traffic. If on-demand routing is implemented, the discovery mechanism is executed as needed by the nodes. In this case there is no pre-built routing table. Whenever a packet requires forwarding, the next hop must be discovered.

In reactive routing, the route discovery process starts when a new packet arrives. This

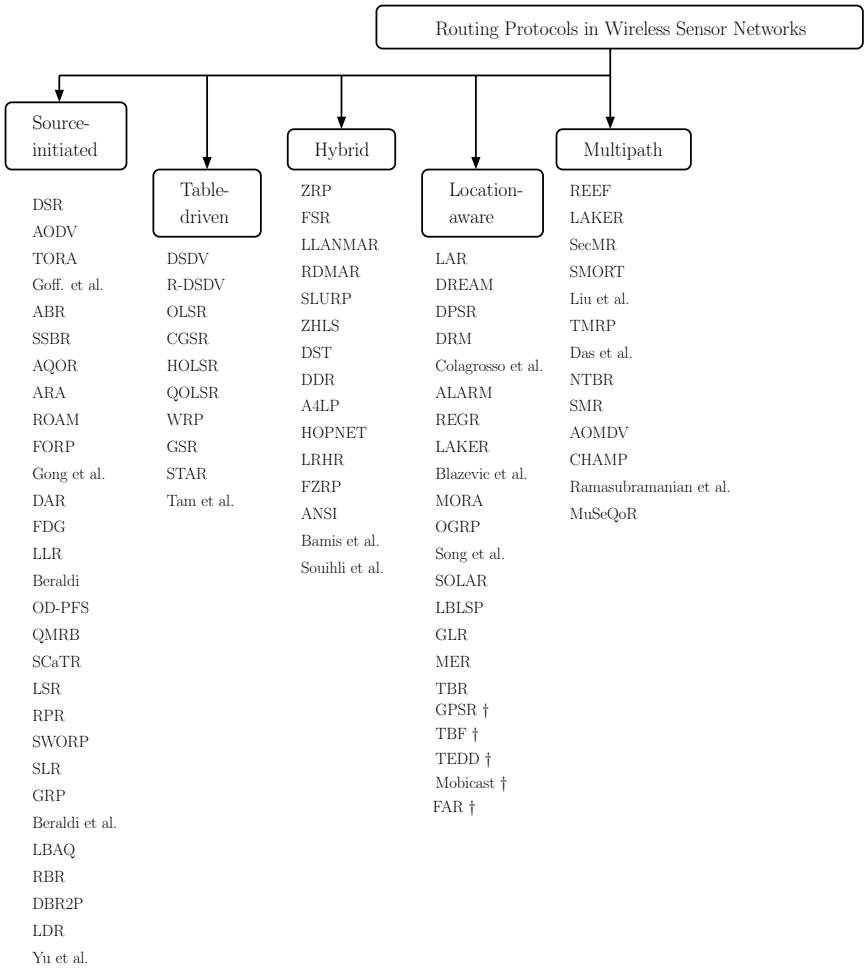


Figure 2.5: Routing algorithms list and taxonomy (part 2). Protocols identified with † are obtained from [86] while the remaining are obtained in [75]

algorithm requires less overhead messages, and may therefore present better scalability characteristics. The main challenge of reactive routing lies in the amount of time required for message delivery. Also in these recent algorithms, several routing proposals introduce GPS to provide location and global time information to nodes. With this information, geographical routing may be implemented. In this case, the destination geographic location may be used as the message address and the location information may be used in the routing process, thus replacing node address. The network topology is therefore fully known to the nodes, including the source and destination node's location. Included in this group of routing algorithms, geo-casting addresses message forwarding to a group

of nodes according to its location. A destination node's location may assume different configurations (e.g. points, polygons or circles).

As in the early proposed protocols, nodes may be organized into clusters thus implementing hierarchical routing. Within each cluster one or more nodes may assume the head role and thus be responsible for message forwarding to their neighboring nodes. The cluster head assumes a number of leading actions within the cluster, receiving data to its neighbors and answering on their behalf. Nodes that are not cluster heads cannot communicate with nodes belonging to other clusters. Different mechanisms may be implemented to organize nodes in clusters, and various nodes my assume the head function in their respective time frame.

Multiple paths from source to destination nodes may be used in Multi-path routing algorithms. This approach increases network tolerance to failure and allows a balanced resource utilization. Path discovery may however entail increased complexity and packet overhead. Algorithms should guarantee path disjointedness, traffic distribution and multiple path maintenance as their main characteristic elements.

In the context of WSN, the power issue is the main research topic. Algorithms have therefore been proposed where power awareness is the main addressed characteristic. Routing algorithms where power aware data forwarding is the main focus are addressed in literature as a separate classification group. An hybrid approach is also included in routing algorithm classification. Hybrid routing algorithms start with a table-driven approach and then the demand originated by additional nodes is addressed by using reactive flooding.

2.5 Operating Systems

WSN Operating Systems (OS) are a relevant research option adopted by several groups. This option is justified by the need to free developers from the low level programming tasks, providing an unchanged cross platform environment. Functionalities such as the processor, memory and peripheral device management, task scheduling and multi-task coping are mandatory on traditional operating systems running on high end computers (e.g. desktop or mainframes computers).

Embedded systems are usually considered to be quite the opposite case for OS related matters since traditionally they are low level programmed and do not require an OS. Also, when addressing OS on a WSN, one must deal with dynamic loading and unloading modules, effective power management policies, concurrency mechanisms and generic API. All

these characteristics must be cross-platform aware. In [100, 101] two surveys that classify WSN OS are presented. The developed framework on which they are classified is based on a set of characteristics an OS should present, namely flexible architecture, that allows service reconfiguration while at the same time maintaining small kernel size. An efficient execution model preventing tasks from blocking program execution and management of race conditions for data access are relevant OS characteristics. An API must be created so that programmers can easily access network, sensor data, memory access, power and task management functions. Run time reprogramming is an important characteristic in WSN. All nodes must be over-the-air re-programmable while running without affecting the system/network operation. Sleep times, data access, data communication and synchronizing must be performed automatically by the OS. Real-time operation is required for some applications, meaning that time constraints should be met during system operation. Table 2.6 obtained from [100] presents a survey on existing operating systems. Table 2.6 compares proposed OS, by classifying them according to a set of relevant characteristics.

An OS can be monolithic or modular depending on the way modules are coded at compile time or loaded as needed. The former case results in a smaller OS image as opposed to the latter. Event based, thread based or hybrid execution models represent the OS performance. Moreover, Table 2.6 characterizes the studied OS according to the ability to reprogram a running node and power management capabilities. Other important factors to consider on WSN are simulation support and hardware portability. Simulation is a fundamental tool for large scale networks as it is a huge logistic problem to deploy a large number of nodes. Hardware portability is a relevant factor due to the large number of platforms described in Section 2.1. TinyOs [37], [102] is the most cited OS for WSN. Developed at Berkeley University for their own motes, TinyOs developers introduced a programming language (nesC) as an extension to the OS. The nesC is a C like programming language that is used to handle TinyOs event based behavior. Mantis [103], Nano-RK [48] and Contiki [104] are other OS proposals that are C programmed. Also, the Operational System Abstraction Layer [105] distributed by Texas Instruments is extensively used by CC2530 programmers.

Table 2.6: Operating system characteristics summary [100].

Feature	TinyOS [37] [102]	MantisOS [103]	Nano-RK [48]	Contiki [104]	SOS [106]	MagnetOS[107]	TI OSAL[105]
Architecture	mono- lithic	modular	mono- lithic	modular	modular	monolithic	monolithic
Execution Model	event-based	thread-based	hybrid	hybrid	event-based	event-based	event-based
Reprogramming	no	yes	yes	yes	no	no	yes
Power Management	yes	yes	yes	yes	yes	yes	yes
Simulation Support	yes	yes	no	no	no	no	no
Portability	yes	yes	no	yes	no	yes	no

2.6 Network Simulation Tools

Network simulation may be a relevant step to develop and implement a WSN. Simulation is specially important when addressing large network behavior and detecting faulty node events. A simulated model may reduce time and deployment/operation costs of a wireless network [108]. Also, testing new applications and protocols is a simplified process when using a simulator. The work introduced in [108] presents a survey on simulation tools for wireless sensor networks. Any wireless network simulator requires detailed node behavior modeling, namely MCU code real time running characteristics, hardware interface timing access and realistic radio frequency propagation models. In Figure 2.6, a number of simulation layers are introduced [108]. Each layer represents a simulation task required to develop a simulator for a wireless sensor network. Figure 2.6 illustrates a communication

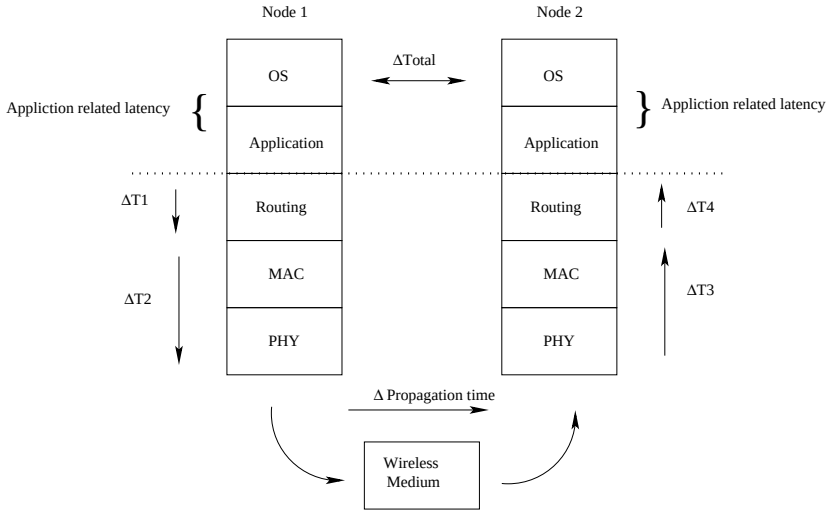


Figure 2.6: Wireless network simulators layer model.[108].

event between two nodes and the time delay associated with each layer. An application related delay must be considered as the time a MCU requires to process and eventually reply to an event. This application delay is quite variable and is affected by parameters such as MCU type, program compiler, oscillator frequency, among many others. The routing layer is essential to evaluate different protocols associated with different network topologies. Moreover, MAC and physical layers are essential to determine delay or even communication success.

In [108] five simulation tools have been identified. NS-2 [109] is a popular discrete event simulator. Originally developed for wired and later for mobile networks it supports

IEEE 802.11 and 802.15.4 type wireless MAC. The 802.15.4 is commonly used for wireless networks since it is able to model basic node energy behavior that is mandatory in several proposed examples of WSN. With its core written in C++ language, NS2 uses two programming languages since two basic functions are required in a simulator. On the one hand, a compiled programming language is used to operate with bytes, packet headers, and managing large data sets as required by some algorithms. This part of protocol simulations requires C++ language as run-time speed is important. On the other hand slightly network configurations or node parameters efficiently use scripting languages to change and configure the simulating environment. This fast changing model and re-run is implemented using a Tcl interpreter. Detailed protocol implementation is provided by C++ thus entailing a slow implementation time but a fast running computation. The Tcl environment is able to configure a simulation with few easy steps. Ns (via tclcl) provides glue to make objects and variables appear on both languages [109].

Similar to NS2, NS3 [110, 111] is an open source discrete-event network simulator that is available for research and development. NS3 has been designed to replace NS2 but it is not its updated version. NS3 is a new simulator and it is not backward-compatible with NS2.

OMNeT++ [112] is a public source component-based discrete event network simulator. It is described by the authors as a modular, component-based C++ simulation framework with an included library. Also a graphical runtime environment for an Eclipse IDE is announced. OMNet++ claims to include the support for simulation of communication networks, queuing networks, performance evaluation, among others, that may be developed as independent projects. Real time simulation extension are provided and several widespread programming languages may be used generate simulation environments (e.g. Java and C#). Database and SystemC integration are also announced provided functions. OMNeT++ is capable of running most TinyOS simulations by a NesCT application that converts TinyOS source to the simulator compatible C++ code [112].

The OPNET modeler is a commercial network simulator that includes a library of detailed protocol and application models. This library has support for Multi-Tier Applications, Voice, TCP, IP, Frame Relay, FDDI, Ethernet,IEEE 802.11 Wireless LANs, UMTS and IP Multicast among a large list of protocols. The Standard Model Library includes vendor specific and generic device models such as routers, switches, workstations, and packet generators [113]. Several versions include wireless network simulations as the support for Zigbee compatible IEEE 802.15.4 MAC.

Prowler [114] is a probabilistic, event-driven wireless network simulator designed for cross-platform use. Running in Matlab environment was originally written for Berkeley MICA motes (c.f. Section 2.1), but it is extensible to other hardware platforms. Original code simulation (C code) is not an easy option with Prowler because a translator to Matlab language (m-file) is not provided. The advantages introduced by the use of the Matlab environment include the fast simulation development and application prototyping. Also the Matlab GUI interface and its graphical programming characteristics may provide fast development characteristics.

2.7 Energy Sources for Wireless Sensor Networks

The energy source is the main constraint when operating any autonomous battery powered electronic system, and is of particular importance for WSN operation. The most obvious energy source for a node is a battery. The most important and basic definition of battery is its category, i.e primary or secondary. Secondary batteries are rechargeable devices unlike primary batteries that are non rechargeable.

Batteries may be identified by a cell, a module or a pack. The characteristics of a cell are determined by the battery chemistry supplying from one to a few volts. Battery modules consist of several cells connected in series or parallel. A battery pack may be built by several modules connected together, also in series or parallel. Large voltages are obtained with battery packs that are built of several modules and cells. The C-rate describes the discharge current profile and is measured in relation to its maximum charge capacity. A 1C rated battery is able to supply current that discharges it entirely in 1 hour. For a battery with a capacity of 1000 mA-hours, a 1C rate equates to a discharge current capacity of 1000 mA. A 5C rate for this battery is able to supply 5 Amps, and a C/2 rate defined the ability to supply 500 mA. The battery state is addressed by several parameters. The State of Charge (SOC), expresses the present battery capacity as a percentage of its full charge value. The Depth of Discharge (DOD) is the battery percentage that has been discharged. DOD is addressed in literature as deep or shallow discharge. Terminal and Open-circuit voltages define the battery capacity to supply a load or maintain a voltage with no load. The internal resistance determines the physical ability to supply large currents for a long period of time A low internal resistance is a desirable characteristic [115]. Either primary or secondary, batteries provide a stable and reasonably reliable power source. Batteries have nevertheless a number of drawbacks that

seriously limit the use of WSN, and are in fact the main limitation to its widespread use. Battery replacement can only be achieved by power energy scavenger systems that, to this date, entail major limitations. This section presents the state of the art for both battery and power scavenger technologies. Several technologies, with different characteristics, have been used for battery assembling.

All batteries operate through a chemical reaction process. This chemical reaction is obtained by four elements: an anode that contributes with electrons, a cathode which accepts them, an electrolyte and a separator. Different compositions of these elements define different types of batteries, each with its own characteristics, relevant according to different system architectures. Battery composition and utilization profile are easily identified by name. When comparing battery technologies, several considerations that apply to all types of batteries, must be taken into account. Energy density is a very important characteristic for WSN as battery stored charge per unit weight may present a crucial difference in size and mote life. Also, power density determines batteries ability to deliver current per unit weight thus determining how much current may be delivered to a load. An ideal battery would be able to deliver as much current as requested by the load (power density) for as long as required (energy density). The amount of times a battery can be charged/discharged before final disposal is also an important factor. The discharge, self discharge and recharge profiles should be taken into account since they largely influence battery life expectancy and may also determine the DC power supply architecture, affecting node cost. Other considerations like environmental impact, safety usage and voltage supply influence battery choice. Finally, as in all engineering artifacts, cost is a major selection constraint.

In the following subsections the characteristics of the main battery technologies used in WSN are briefly discussed, and their advantages and disadvantages are enumerated. Battery technology characterization data has been gathered from multiple sources, with different considerations that are sometimes contradictory. Comparisons are therefore difficult to obtain. Due to different manufacturing processes, batteries perform differently even considering models from the same manufacturer. Moreover, several battery characteristics are aimed at Electric Vehicles implementation, thus entailing very different concerns than those addressed by WSN. The best approach to determine the optimum battery technology should take into account its particular use. Frequently, battery manufacturing is optimized for the actual use thus entailing a very relevant effect on its operation.

2.7.1 Lithium Based Batteries

Lithium based batteries are currently the most promising chemical technology, and those whose market registers the largest growth. When compared with all other technologies (chemistries), Lithium based cells show a number of advantages that support widespread use. Each cell nominal voltage can be set to 3.6 volts, allowing battery pack designs with only one cell. Lithium-ion packs have typically twice the energy density of the previously most used technology (nickel-cadmium). Moreover, further development promises higher energy densities. When compared with other type of cells, Lithium-based cells shows a much smaller self-discharge, no memory effect and cause little environmental impact when disposed. Lithium based cells present a self-discharge rate of 0,1 to 1%/month [115], resulting in a 10 year shelf life expectancy.

Despite its overall advantages, lithium-ion drawbacks can be easily identified. Its voltage cannot be lower that 2.0V and a protection circuit is required to prevent malfunctions and/or destruction. Protection circuits that limit both the upper and lower peak voltages are built in each pack, preventing excessive charging and discharging. Also, cell temperature monitoring must be performed to prevent overheating. The flat discharge voltage curve can be seen as both an advantage and a disadvantage. Lithium batteries require a high pulsating charging current thus increasing recharge circuit complexity. A special case of Lithium-based cells are the Lithium Polymer batteries. This technology presents similar performances and characteristics to Li-Ion cells, lower manufacturing costs, light construction and small form factor design.

2.7.2 Zinc Based Batteries

Also known as "Leclanché cells", Zinc-Carbon (Z-C) cells are low cost batteries that produce a 1.5V typical voltage output and are not rechargeable. Z-C cells are very popular in household small and low power portable devices (e.g. flashlights and portable radios). Their double function zinc casing/anode is the main drawback due to frequent leakage of the mildly acid electrolyte. This leakage frequently damages powered devices. Z-C batteries are therefore used in non-critical applications. Together with Z-C batteries, Zinc-Manganese Dioxide Alkaline cells - also known as "Alkaline Batteries" - are very common in household appliances. Alkaline batteries have a useful expected life-time of five to six times more than zinc-carbon batteries. They produce a 1.5V output voltage and, like Z-C batteries, present a mildly sloppy discharge curve vs output voltage. Both Z-

C and alkaline batteries are usually non chargeable (primary) batteries, except for a small number of rechargeable alkaline batteries offered by a limited number of manufacturers. The number of recharges is nevertheless small (approx. 25).

2.7.3 Nickel Based Batteries

Nickel-Cadmium (Ni-Cd) and Nickel-Metal-Hydride (NI-MH) are Nickel based rechargeable batteries. Ni-Cd technology is characterized by their low cost and high discharge current, making it suitable for electric power tools and other high current applications. Other relevant characteristics of Ni-Cd batteries are the flat voltage 1.2V level discharge current and the 1000+ charging cycles. Nickel-Cadmium batteries are well known for their memory effect caused by incomplete discharge. When not completely discharged, Ni-Cd batteries are not able to fully recharge to their original capacity. Due to Cadmium components, these batteries cause significant environmental impact and added cost with recycling processes. Ni-MH batteries do not have the memory effect of Ni-Cd, and have higher energy density and lower discharge current. Nickel based batteries offer high leakage current, being unsuitable for long time storage periods and for applications with very low discharge currents. This particular characteristic makes Nickel based batteries unsuitable for powering WSN motes.

2.7.4 Lead Acid Batteries

Each Lead-Acid cell supplies about 2.14 V when fully charged. Their are used in the 12 V six cell car batteries for combustion motor startup. Developed at the end of the 19th century, lead-acid cells have been the first commercially available technology for batteries. Led-Acid battery technology is nowadays a low cost solution and is therefore a popular choice for a number of applications due to its low market price. Self discharge rates vary from 4% to 6% per month [116]. In systems that are permanently connected to charging sources, whether it is solar, wind, or an AC powered charger, this is seldom a problem. Price and availability are the sole reason they are mentioning in this work. Lead-acid batteries are still a good choice to power any system that requires high current (i.e in the order of tens of mA), and is disconnected from AC mains.

2.7.5 Super-capacitors and Micro-Fuel Cells

Capacitors are passive devices that are able to store electrostatic energy in the form of an electric field between two charged terminals. They present significantly lower energy density than batteries. Capacitors main advantage over batteries is their smaller charge/discharge times, thus showing higher power densities than batteries. Capacitors are therefore ideal to provide short high current bursts with low duty cycles such as those required by WSN nodes communication tasks. Within the context of WSN node operation, the capacitors fast recharge characteristic is well suited for the radio frequency communications required current bursts. This provides system designers with the possibility of combining capacitors and batteries to address nodes duty cycle operation power requirements. Batteries are able to provide nodes with current for low power applications while capacitors are able to power RF communications for short periods of time. As stated in [117], super-capacitors differ from standard capacitors because they present a larger electrode surface area as well as a thinner electrode-electrolyte interface. Unlike standard capacitor where capacitor area is formed by the surface of flat plates, supercapacitors use porous materials thus increasing effective plate area. This plate area increase is achieved by using carbon plates for each electrode. capacitances in in the order of $2000F$ are therefore commercially available with standard battery sizes packages. This technology is able to provide capacitance values as high as $8000F$ [117].

Fuel cells generate electricity by combining hydrogen and oxygen. This process is done without combustion and its byproduct is water. Hydrogen atoms are separated from their electron by the catalyst. Electrons are thereafter driven through the external circuit while protons are diffused through the fuel cell structure and recombined with their original atoms plus oxygen [117]. This technology, initially used by NASA space program with large form factor, has recently been presented in portable formats. A miniature fuel cell technology is the direct methanol fuel cell (DMFC) as shown by the work presented in [118]. A different technology, Polymer Electrolyte Membrane (PEM), is proposed in [119] to power WSN nodes.

2.7.6 Energy Harvesting

Energy harvesting is the process through which energy is collected from the environment and stored. Energy can be captured from a number of sources. Solar power, salinity gradients, thermal energy, kinetic energy, wind energy, nuclear radiation and radio frequency

may be sources for harvesting energy to power embedded systems [120, 121, 122, 123, 124, 125, 126, 117].

Photovoltaic solutions are perhaps the most obvious power source for WSN nodes. Photovoltaic panel powering can be differentiated into two profiles depending on whether they are placed outdoors or indoors. Outdoor solar energy harvesting produces a large variation of scavenged power as day light varies from a clouded to a sunny day. Indoor variation is due to different light sources and different illumination profiles [120, 115, 127, 128, 129, 130, 131, 132].

Mechanical force can generate energy if an inertial mass is used to create movement. Movement can be converted into electric energy using three mechanisms: electrostatic, piezoelectric and electromagnetic [120, 133, 134, 135]. A vibrating piezoelectric device is used to convert into electricity mechanical energy from force, vibration or pressure. The piezoelectric harvester consists of a capacitor formed by one or several piezoelectric layers sandwiched between metallic electrodes. The principle of operation of the energy scavenger is based on the mechanical strain exerted on a piezoelectric mass. In piezoelectric transducers, vibrations or movement cause the deformation of a piezoelectric capacitor thus generating a voltage [115]. This property of piezoelectric materials has led researchers to develop various piezoelectric harvesters to power different applications. In [136] a survey on Piezoelectric harvesting technologies is proposed. Electrostatic energy harvesting is done by changing the capacitance of a variable capacitor. This mechanism uses plate vibration of a previously charged capacitor to produce electrical energy. Figure 2.7 illustrates this mechanism. Promoting changes on a magnetic field through mechanical vibration can be

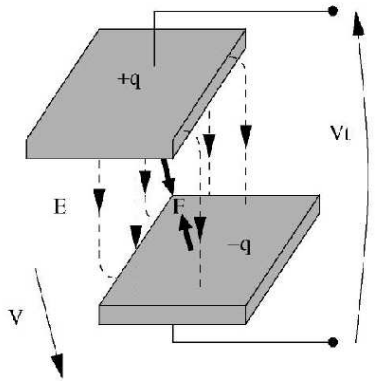


Figure 2.7: Electrostatic energy harvesting description [137].

described as electromagnetic energy harvesting. An electromagnetic induction is therefore achieved by using a permanent magnet, a coil and a mechanical device. Figure 2.8 illustrates this mechanism in a wrist watch power source.

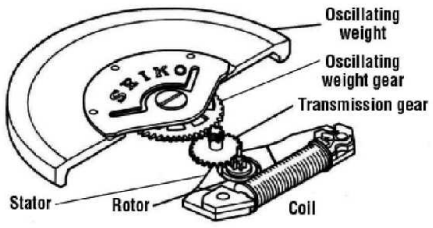


Figure 2.8: Electromagnetic energy harvesting mechanism [137].

The Seebeck effect is used for thermal energy harvesters. This effect states that a voltage is created at the junction of two different metals or semiconductors if there is a difference of temperature between both plates. The elements forming this thermal effect are known as thermopile thus forming the energy harvester. Each thermopile is built with a number of thermocouples connected thermally in parallel and electrically in series and placed between a hot and a cold areas [138, 139, 140, 141]. Radio frequency is identified as a possible energy harvesting source for embedded systems. RFID[4] is a common technology where a small circuit is powered by radio frequency bursts. The exact same principle can be adopted by WSN nodes as a radio frequency burst can be used to power a group of nodes [115, 142, 143, 144, 145].

Nuclear diodes junction batteries are presented in [146, 147, 148]. These works suggest a 50+ year lifelong energy source with a 1.5 V voltage value and currents of nano to micro ampere. The $63Ni$ isotope is a device that directly converts nuclear to electric energy with a very low alpha radiation emission. This system could power a WSN node even if it is a low current power source, due to the high power density it presents. Nuclear power source is also the focus of [149] as radiation is used to induce vibration on an oscillating cantilever. If a mechanical link to a electromagnet exists, electric power can be generated. Even if both these works fail to present an energy density estimate, they show a promising path on how to develop a long lasting energy source for WSN nodes. Table 2.7 summarizes the main characteristics of the most relevant harvesting technologies.

[4]RFID: Radio-Frequency Identification

Table 2.7: Energy scavenged power from several sources [120, 117]

Energy Source	Characteristics	Efficiency	Volumetric Power ($\mu W/cm^3$)	Volumetric Energy (J/cm^3)
Battery	Primary	$60\% - 70\%$	$n.a.^\star$	1200 - 3800
	Secondary	$80\% - 90\%$	$n.a.^\star$	600 - 1100
Supercapacitors	$n.a.^\star$	$n.a.^\star$	$n.a.^\star$	10 - 20
Micro-fuel Cell	$n.a.^\star$	$n.a.^\star$	$n.a.^\star$	1000 - 3000
Light	Outdoor	$10 \leftrightarrow 24\%$	$100mW/cm^2$ †	$n.a.^\star$
	Indoor	.	$100\mu W/cm^2$ †	$n.a.^\star$
Thermal	Human	$\approx 0.1\%$	$60\ \mu W/cm^2$ †	$n.a.^\star$
	Industrial	$\approx 3\%$	$1 \leftrightarrow 10mW/cm^2$ †	$n.a.^\star$
Vibration	$\approx Hz \leftrightarrow human$	$25 \leftrightarrow 50\%$	≈ 4	$n.a.^\star$
	$\approx kHz machines$	.	≈ 800	$n.a.^\star$
RF	GSM 900 MHz	$\approx 50\%$	$0.1\mu W/cm^2$ †	$n.a.^\star$
	Wi-Fi	$\approx 50\%$	$0.001\ \mu W/cm^2$ †	$n.a.^\star$

* Not applicable.

† Usually specified in cm^2 rather than cm^3.

2.7.7 Energy Harvesting Converter Circuits

In general, the output of an energy harvester can't be used to directly power embedded system circuits [115, 150, 132]. In a harvester system, the power availability is variable over time, and can range from zero to the maximum power delivered by the scavenger device. Conversion circuits must therefore be used to accommodate voltage and power levels. Power management circuits must be able do adapt to harvester conditions and characteristics. The variability in voltage, power density, and duty cycle requires careful design of conversion circuitry to effectively collect harvested energy. Energy can be collected from either a single energy source or multiple concurrent energy sources. The power management unit should be able to handle very low feeding power and should also be self-starting. If the power generated is on the order of milli-Watts, an efficient power management system is easy to construct, the task becomes non-trivial in the 100 μW range [115].

The energy scavenger mechanism option dictates the type of conversion electronics. The main design constraint is efficiency, meaning how much of the energy provided by the harvester is actually converted and delivered to the circuit application. Thermoelectric generators, solar cells and nuclear diodes produce a variable DC output voltage. They

therefore require a DC-DC converter that is able to adapt its power conversion ratio so that a stable voltage is provided to the charge. Vibration and RF energy harvesters provide a variable AC output voltage. A rectifying AC-DC converter is therefore required before the output voltage regulator.

A number of COTS manufacturers propose solutions for energy harvesting management. Linear Technology offers LTC3108, LTC3109, LTC3558, LTC3105 and LTC3330 [151] circuits that are able to harvest energy from various sources and provide stable voltage output supplies. LTC3109 is a dual circuit of LTC3108 while LTC3558, LTC3105 and LTC3330 are Set-Up DC/DC converters that can be used to power a device from a harvesting source. LTC3108 offers two stable voltage supplies and a surplus storage output that can be wired to a large capacitor thus operating as a charge reservoir. Maxim Integrated MAX17710 [152] offers two outputs (regulated and non-regulated) that are able to power an embedded system. A number of harvesting sources can be used and a regulator output may be controlled by digital input lines. Texas Instruments BQ25504 [153] also offers scavenger capabilities but requires a regulator at the output to power system load.

2.8 Related Work

The work presented in [154] reviews and compares available options for powering WSN nodes from large electric and magnetic fields which exist near high-voltage electrical installations such as substations. In [155] a microprocessor controlled electromagnetic energy harvesting device from a single high voltage transmission line is presented. This system is able to scavenge energy from line currents between 65 A and 130 A. A magnetic power generator and a voltage multiplier are used.

In [156] a communication concept for inverter fed electric motors is proposed. Powerline communication is used to send motor operational data. Power supply is obtained by inductive coupling from the remaining motor phases. In [157] a small electromagnetic energy harvesting device is proposed, where several topologies for the magnetic coupler are tested. This energy harvesting device is used to charge a battery that can in turn power a wireless device. These works analyze the harvesting ability of the proposed devices, but no mention is found that relates protocol complexity with the developed devices.

The protocol implication of running a WSN node on harvested energy is investigated in [158, 159]. The presented results determine the optimal amount of energy to be drawn from the storage devices given the known state of charge (SOC). Moreover, the impact of

defective SOC estimation is computed and the harvested energy addressed as a random variable. This work assumes the microcontroller to be the heart of the energy management process but addresses its operation using a generic approach. Also in [160, 161], solutions are proposed that may be related to the work presented in this thesis.

In [160] a harvesting device running on capacitor stored energy is described. This device offers a software controlled mechanism that maximizes the nodes sensing and communication. The system scavenges energy from the environment radio frequency and accounts for capacitor leakage currents to control the sensing and communication mechanisms. In [161] a device is presented that is able to scavenge energy from high and medium voltage power lines electric fields. Simulations and an implemented DC regulator are presented. A free-standing inductive harvester for uses where a magnetic field induced by power cords is available, is investigated in [162]. Also, in [163], a device is proposed that is able to obtain energy from the magnetic field induced by power cords. This device is able to scavenge energy from small currents. A device that is able to harvest energy from the electric field existing near high voltage DC lines is presented in [164].

A number of techniques for disaggregated end-use energy sensing for smart grids are surveyed in [165]. WSNs are an important tool to implement measurement systems [166, 167, 168]. In [29, 169, 170, 171] energy harvesting devices are able to power IEEE 802.15.4/Zigbee compliant nodes. These contributions focus their analysis on the Zigbee data transfers and the sustainability of the system under regular communications. The IEEE 802.15.4/Zigbee protocol entails a number of non regular communication action that are not addressed in these proposals. Moreover, the sustainability of a nodes operation with actions that are not related to the protocol in not considered. Adding a harvested energy powered node, besides the complex protocol operation, the ability to execute sensing and processing capabilities may be a relevant contribution. Addressing Zigbee non regular communications that absorb large currents from the power source, may also be a valid contribution.

In [126]the concept of Energy Neutral Operation is introduced for nodes powered by harvested energy. Node operation such that the used energy is always less than the harvested, is discussed in this work. Operation modes are divided in two groups: Harvest-Use and Harvest-Store-Use devices. Within the context of node operation, $E_s(t)$ is the energy produced by the harvesting source and $E_c(t)$ is the energy consumed by the node. In the Harvest-Use mode, node sustainable operation entails that $E_s(t) > E_c(t)$. A Harvest-

Store-Use operation is more flexible and is defined as

$$E \geq E_0 + \int_0^T [E_s(t) - E_c(t) - E_{leak}(t)]\, dt \geq 0 \tag{2.5}$$

where E is the nodes finite storage capacity, E_0 is the nodes initial stored energy, E_{leak} is the energy lost by the storage element leakage currents for a period of time T.

Chapter 3

Wireless Sensor Networks - The IEEE 802.15.4/ZigBee Protocol

The focus of this chapter is placed on IEEE 802.15.4/Zigbee [33, 172]. IEEE 802.15.4/Zigbee is a well specified, studied and described standard. Moreover, a number of protocol stack implementations are freely distributed ([173, 174, 175]), making it a very useful tool for test and verification of WSN with an added characteristic of reduced time for market implementation. To the purpose of this work, the choice of this complex protocol is justified by the fact that if the system is able to operate using IEEE 802.15.4/Zigbee, a different protocol with a simpler implementation will also be able to operate.

In [31] a simple protocol has been implemented. In this case, nodes transmit their data without requiring an acknowledgement. Data is not therefore required to be retransmitted and collisions are not detected nor addressed. Also, nodes do not join the network and their transmitted data is recognized by the remaining network nodes by the packet header. Moreover nodes are not required to route data as a star topology has been implemented. IEEE 802.15.4/Zigbee is one example of a protocol that requires devices to send and receive data using coordinated actions. Node must be able to route other nodes data and data may be transmitted over several hops before final destination delivery. Throughout this work, IEEE 802.15.4/Zigbee is therefore classified as a complex protocol when compared with other implementations.

The Zigbee protocol specifies a wireless technology based on the IEEE 802.15.4 standard [33] for WPAN. Designed to enable RF communications between multiple devices, IEEE 802.15.4/Zigbee, is a flexible and adaptable protocol allowing the implementation of

low power and low cost systems. Besides the protocol definition itself, in [176] and [177] two clear and concise descriptions of IEEE 802.15.4/Zigbee can be found. A large number of bibliographic documentation describes protocol working details, as well as implementation efforts and solutions. This chapter is a State-of-the-Art extension for IEEE 802.15.4/Zigbee characteristics and operational mechanisms based on the TI protocol stack implementation [173] known as Z-Stack. A TI CC2530 [56] microcontroller with a radio frequency module for 2.4-GHz is proposed for this work. Throughout this study experimental results are obtained with a CC2530 device running Z-Stack.

A general protocol overview is carried out, followed by a description of the IEEE 802.15.4 operational mechanisms. Zigbee is described in the chapter follow-up. Throughout the text, IEEE 802.15.4/Zigbee will be referred as Zigbee unless otherwise noted. Zigbee fundamental characteristics include:

- **Standards-based wireless technology** - This is one of the main characteristics addressed by Zigbee - it is a standard. The Zigbee and the IEEE 802.15.4 standard, in which it is supported, are defined worldwide. Zigbee is a collection of definitions, thoroughly described, so that all compliant devices communicate with each other.

- **Low data-rates and low power consumption** - When describing Zigbee, all commercial and non-commercial products refer its low power and low data rate characteristics.

- **Support for small to large networks** - Before joining a Zigbee network, a device must be given a 64-bit address. This is a global number with an Organizationally Unique Identifier (14 bits OUI) and 40 bits that are assigned to each radio module manufacturer. OUIs are obtained from the Institute of Electrical and Electronics Engineers (IEEE) thus ensuring that they are globally unique. Moreover, by joining to the network the device receives a 16-bit address known as the network address. Either the OUI or the 16-bit network address are valid node identifiers within the network.

- **Security and Reliability** - These two characteristics are addressed by Zigbee standards. Security has its own defined data structures and employed reliability measures. Adjacent Zigbee networks do not interfere with each other and operation is possible in the neighborhood of networks such as Wi-Fi and Bluetooth.

Reliability measures include listen before sending communication arrangements using CSMA-CA. Zigbee has a built in acknowledgement mechanism that allows a receiving device to acknowledge an incoming message. The network may present mechanisms to ensure that messages reach their destinations. Security is achieved by data encryption using a 128-bit encryption system based on the Advanced Encryption Standard (AES) algorithm. Message timeout is used to discard messages that are too old. A frame counter is added to the messages, allowing a node to determine how old a received message is.

3.1 Zigbee Competitors

The Zigbee protocol operates in the 868MHz band at a data rate of 20Kbps in Europe, 914MHz band at 40Kbps in the USA, and the 2.4GHz at 250Kbps ISM bands worldwide [33]. For characterization purposes, the Zigbee specification is usually compared with other IEEE protocols such as Bluetooth [178], Groupe Spécial Mobile (GSM) also known as Global System for Mobile Communications, WLAN (Wireless LAN - IEEE 802.11) usually known as WiFi, and non IEEE protocols such as Zwave [179], Enocean [180] and KNX [181].

All comparisons to Zigbee and WiFi are bound to be unfair for both parts. WiFi is developed for high data rate communications, such as video and image-rich web browsing, and is designed for resident equipment and corresponding applications. Cabled traditional Ethernet networks can, in most cases, be replaced by WiFi which, like cabled networks, requires configuration. The protocol is described as wireless local area networks (WLAN). With power thirsty and large bandwidth characteristics, it is not recommended for automation applications that do not require Audio/Video transmissions. WiFi is supported on the IEEE 802.11 protocol definition and uses the same radio frequencies as Bluetooth and Zigbee.

The first use of Bluetooth was for cable replacement of serial data point-to-point connections. Power requirements of early Bluetooth are similar of those of WiFi. Original Bluetooth devices are used in applications such as setting up networks, printing communications, or file transfer. Early Bluetooth is designed for non-resident/mobile equipment and its applications. These applications are generally described as a WPAN and are ratified as IEEE Standard 802.15.1. Further developments led to reduced power Bluetooth devices, as well as small network self-organization capabilities with Bluetooth Low En-

ergy (BLE). BLE is a strong Zigbee competitor for network star topologies and has the advantage of being native to the large majority of mobile phones and computers. The recent 4.1 and 4.2 Bluetooth versions, provide an interesting base for developing networks for small sensor devices.

The Z-Wave protocol [179] was explicitly developed by Zensys for home control applications. Z-Wave is a proprietary protocol with two basic kind of devices: controlling devices and slave nodes. Slave nodes reply and execute commands sent by controlling devices that initiate messages within the network. There is always a single, referred to as the primary, controller that operates as the network topology information manager. Slave nodes can have several forms depending on their function. Routing slaves forward commands to other nodes, enabling the controller to exchange data with nodes out of direct radio distance. Source routing is used in the early protocol versions which means that the whole data path is determined at the moment the frame created is issued from the sender [182]. Networks can be formed with up to 232 devices. FSK (frequency shift keying) modulation is used at 908.42MHz in the United States and 868.42MHz in Europe. The RF data rate is advertised as being up to 40 Kbps.

Zensys sells a SoC IC with the transceiver, an 8051 microcontroller core and a triac controller that includes a zero crossing detection circuit [179]. The microcontroller implements the Z-Wave protocol as well as the application software that are sold in closed library functions. Each developer has the freedom to implement C code based on those closed functions. The protocol and device class specifications are shipped only with each acquired kit and are not publicly available.

EnOcean uses energy harvesting from the environment to power a wireless transmitter-only sensor node [180]. With this option, a significant reduction in maintenance operations is achieved as there is no need for battery replacement. Energy is provided by piezoelectric elements or solar cells using only the amount needed to collect all sensor data and transmit a telegram. EnOcean uses a proprietary protocol optimized for energy saving. Messages are up to a maximum of six bytes long and are transmitted at a data rate of 120 Kbps. Additionally, leading zeros are not transmitted and transmission is announced to takes less than 1 ms. The transmission reliability is not controlled by the protocol as acknowledgments are not available because battery-less transmitter modules do not contain a RF receiver. The low probability of collisions is however presented as an argument to use transmit only modules. There is only one supplier of EnOcean radio modules. There are currently 4 radio telegram types (corresponding to the available transmitter modules)

identifying various combinations of Boolean and 8-bit integer values, ensuring a basic level of interoperability. Documentation for these modules is freely available, but only allows to guess the radio protocol [182].

Konex is a home automation standard proposed by the KNX association. The Konex protocol proposes a twisted-pair, a power line and a wireless transmission medium. The wireless transmission medium is called KNX RF and operates at 868.3 MHz using FSK modulation at a data rate of 16.4 Kbps. KNX radio frequency protocol allows transmit-only devices in addition to the classic bidirectional devices. Transmit-only devices are cheaper since only a subset of the protocol stack has to be implemented. Moreover, eliminating the receiver extends the battery lifetime. The drawback of transmit-only devices is preventing the possibility of downloading applications and device configuration, which is one of the most important characteristics of the KNX protocol. KNX RF does not use link acknowledgments since transmit-only devices are not able to receive them.

Digimesh [183], developed by Digi International, is a proprietary peer-to-peer networking protocol for wireless network connectivity. The most relevant characteristics announced by Digimesh are a peer-to-peer architecture which includes support for sleeping routers and dense mesh networks. As a portable protocol, DigiMesh can be deployed in a number of wireless products servicing a broad array of application needs, including multiple frequencies and hardware platforms [182].

JenNet [184] is a proprietary protocol stack developed by Jennic for short-range wireless networking applications based on the IEEE 802.15.4 specification. Jennic is a fabless semiconductor company that develops wireless connectivity products and microcontrollers for applications in energy, environment, asset tracking and consumer markets. The company products include low power wireless microcontrollers, modules, development platforms, protocol and application software, with a focus on IEEE802.15.4, Zigbee and 6LoWPAN [185] standards. JenNet offers an alternative to the Zigbee protocol. An application programming interface (Jenie) is supplied thus allowing the development of applications.

Wireless communication protocols for industrial applications are specified by the Wireless HART [68] protocol and the ISA100.11a [69] as explained by the work presented in [70]. In [186] the Zwave protocol has been used to implement an Advanced Meter Reading Infrastructure (AMRI) based on a wireless network. A number of non-standard, non-commercial protocols are also proposed in literature. In [31] a simple protocol was proposed and its advantages and operational limitations identified.

The 6LoWPAN defines an Internet Protocol (IP) adaptation for mesh networks and the IoT applications. 6LoWPan is the acronym for "IPv6 over Low-Power Wireless Personal Area Networks" and is an interface layer that allows IPv6 packets to be transported over the IEEE 802.15.4 protocol. The use of an IP-based infrastructure over IEEE 802.15.4 allows the integration of WSN with the traditional IP networks. Originally defined for IEEE 802.15.4, 6LoWPAN definition is addresses by recent radio frequency applications such as sub-1 GHz low-power protocols, Bluetooth and low-power Wi-Fi [185].

3.2 Zigbee Overview

The link between Zigbee and IEEE 802.15.4 can be described using the referenced OSI layers identified in Chapter 2. As illustrated in Fig. 3.1, the IEEE 802.15.4 defines the Physical and Medium Access Control layers for wireless networks. Zigbee is built on top of IEEE 802.15.4 and provides a definition for two layers of the OSI model: the Application Layer (APL) and the Network Layer (NWL) [177]. The MAC layer provides services to

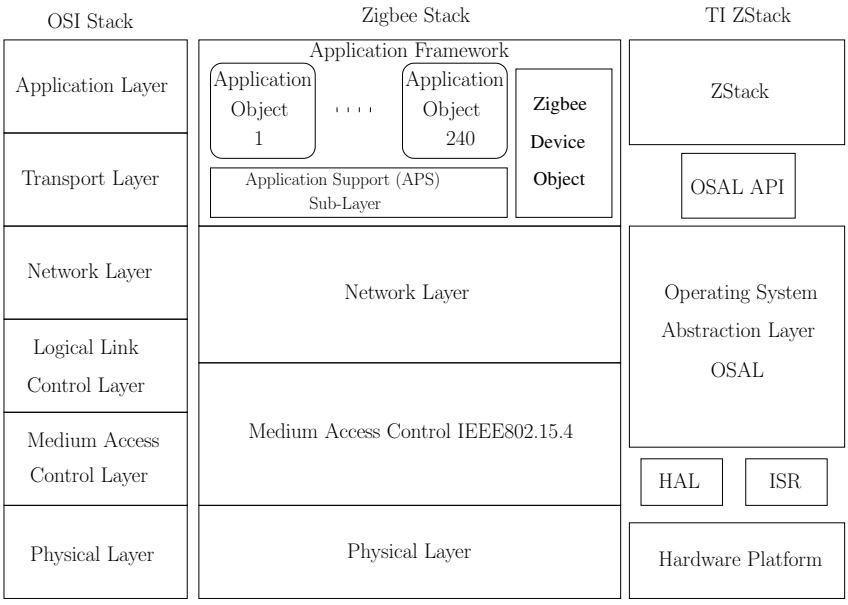


Figure 3.1: Zigbee OSI layers implementation for Zigbee and Z-Stack.

the application layer through the MAC management entity (MLME) and data transfer services using the MAC Common Port service (MCPS) [172].

IEEE 802.15.4 defines two physical devices: Full Function Devices (FFD) and Re-

duced Function Devices (RFD). FFD devices may be the network coordinator, they can operate in any topology and may route data to any device in the network. RFD cannot become the network coordinators, may only transfer data to their parents and have a simple implementation (e.g. hardware). IEEE 802.15.4 definitions allow RFD to be low-power/battery-powered devices.

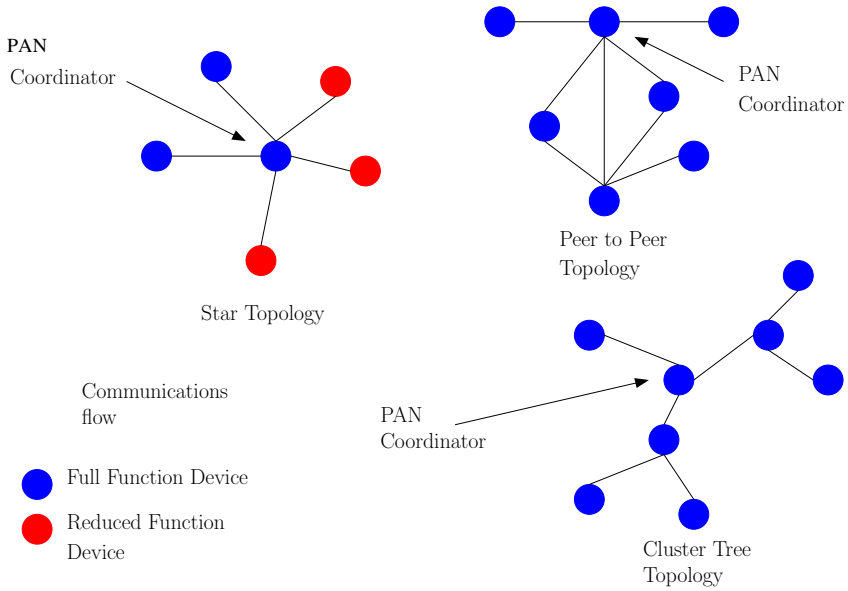


Figure 3.2: Zigbee network possible topologies [177].

Zigbee protocol definition uses the IEEE 802.15.4 FFD to implement its Zigbee Router (ZR) and Zigbee Coordinator (ZC). Zigbee End Device (ZED) are usually associated with IEEE 802.15.4 RFD. There is only one coordinator (ZC) in each network and it is this device that establishes the network. The coordinator stores information about the network. Routers (ZR) are the intermediate nodes, relaying data from other devices. End Devices (ZED) can be low-power/battery-powered devices. They have sufficient functionality to talk to their parents (either the coordinator or a router) and cannot relay data from other devices [187, 176, 177].

In Figure 3.2 three possible Zigbee network topologies are presented. Nodes can be organized in star, peer-to-peer or mesh, and cluster tree topologies. In all topologies, a Zigbee network requires at least one FFD to act as a network coordinator, while endpoint devices may have reduced functionalities. As illustrated in Figure 3.2, the star topology has a unique FFD operating as a ZC. The star topology is a simple organization method that can be easily implemented. All data traffic must be done through the ZC node. The

ZC will forward messages to other nodes in the star as well as to other networks. As a consequence, this node will soon deplete its power source if it is battery powered.

In a common procedure to all topologies, the ZC defines a 64-bit number that is used as a network identifier and referenced as Personal Area Network (PAN) identifier. Due to short radio range, the star topology is used only in small scale networks. In a mesh topology each node may communicate directly with any other node in range. Network identification is still provided by the ZC that is not required in all communications. Nodes can send data using ad-hoc associations and multiple hops to route data. The mesh topology allows the implementation of large scale networks as radio distance is not a limitation. Complexity to network operation is nevertheless introduced. A clustering topology is implemented in Zigbee as an hybrid mesh-tree organization known as cluster-tree network. A single route between any pair of nodes, as well as synchronization mechanisms can be used. The cluster-tree network topology represents a trade off between star and mesh configurations. This topology requires a less complex implementation than the mesh topology and solves the problem of being able to work in large scale networks. The ZC identifies the entire network with one ZR per cluster. Any of the FFD can act as a ZR providing synchronization services to other devices and ZRs. Table 3.1 [177] summarizes relevant differences between Zigbee star, mesh and cluster-tree topologies.

	Star	Mesh	Cluster-Tree
Scalability	No	Yes	Yes
Synchronization	Yes(No)	No	Yes
Inactive periods	All Nodes	ZEDs	All Nodes
Guaranteed bandwidth	Yes(GTS)	No	Yes(GTS)
Redundant paths	N/A	Yes	No
Routing protocol overhead	N/A	Yes	No
Commercially available	Yes	No	Yes

Table 3.1: Characteristics of allowed Zigbee topologies [177].

3.3 Medium Access Control Specification

A TDMA [188, 189, 190, 191] scheme, with a CSMA-CA [192, 193] mechanism, may be used by radio transmission for medium access. The TDMA allows the optional use of a superframe structure, as illustrated in Figure 3.3. The superframe is bounded by network beacons and has a variable size defined by the network coordinator. Periodic data may

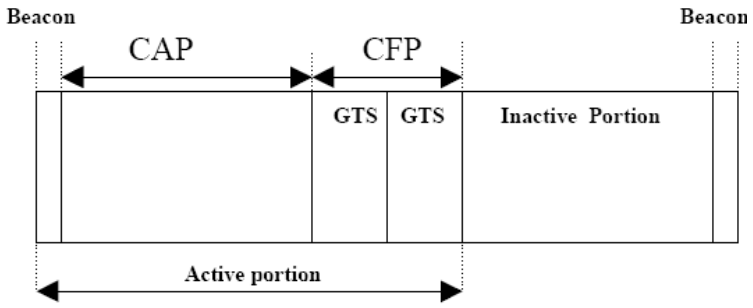


Figure 3.3: Zigbee superframe illustration.

be handled using the beaconing framework whereby nodes listen for the beacon, transfer data, check messages and go back to sleep. A beacon frame is transmitted in the first slot of each superframe. The beacons may be used to synchronize network nodes, to identify the PAN, and to describe the time structure of superframes. The superframe is divided in tree subframes [177]. A Contention Access Period (CAP) immediately follows the beacon frame. Any device wishing to communicate during the CAP must share de radio channel with other devices using a time slotted CSMA-CA mechanism. Applications may choose a Guaranteed Time Slot (GTS) option. GTS can be used by any device for specific actions during each superframe. The GTSs are issued during the Contention Free Period (CFP) which always occurs immediately after the CAP. The PAN coordinator may allocate up to seven of these GTS. A GTS may have more than one slot period. However, a portion of the CAP remains for contention based access of other networked devices or new devices wishing to join the network. All contention based transactions must be completed before the CFP. Also, each device transmitting in a GTS has to ensure that the transaction is complete before the next GTS or the end of the CFP. All transactions have to be completed by the time of the next network beacon transmission [176]. A inactive period allows devices to sleep. If a coordinator does not wish to use a superframe structure it may turn off periodic beacon transmissions. Intermittent data may be handled either in a beacon-less system or in a disconnected mode. In a disconnected operation the device

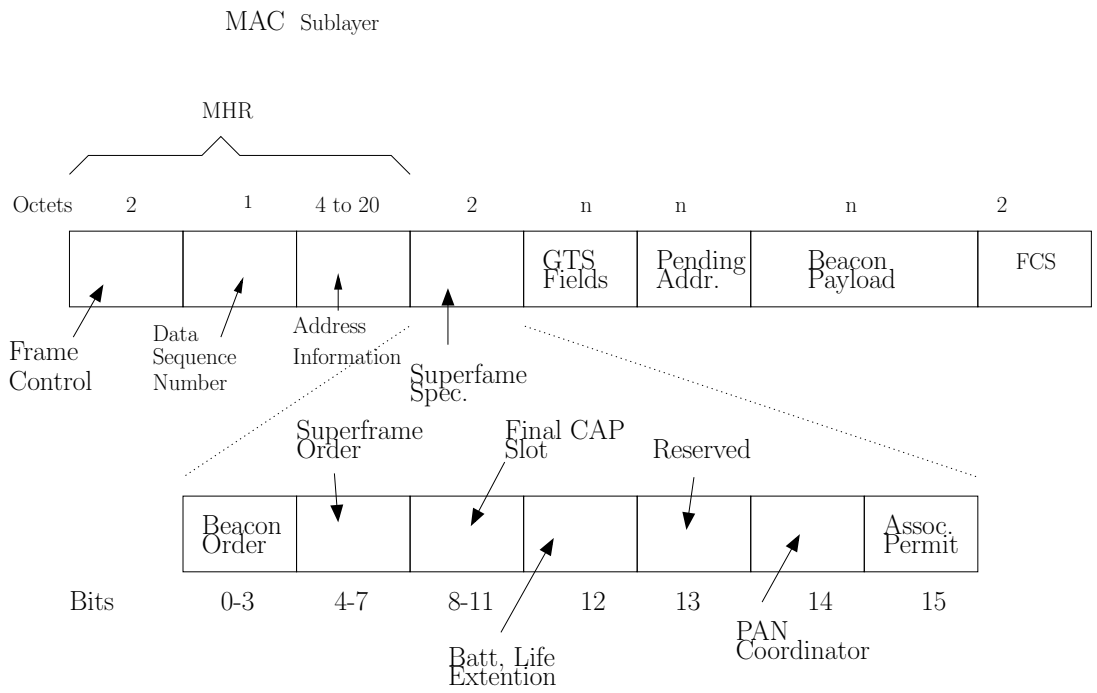


Figure 3.4: Zigbee Beacon Frame Structure where MHR is the MAC common frame header.

only attaches to the network when it needs to communicate. Beacon-less networks may be used to handle intermittent or non-periodic data. In this case, devices simply use a CSMA-CA mechanism to transfer data between devices. These operational modes require nodes to previously connect to the network before executing actual data transactions. Zigbee operation is, in this case, done without the use of a superframe. In this mode, beacons are sent by the PAN coordinator for network identification purposes only, and unslotted CSMA-CA is used for communications. The coordinator must have its radio turned on and ready to receive data. End devices will periodically wake up and poll the coordinator for pending messages. The ZC sends pending messages or signals that no messages are available. Coordinator to coordinator communication poses no problems since both nodes are always active. Other than these so called connected modes, devices may connect to the network only when required, using a disconnected operation mode where protocol configuration is executed previously to all data transactions. Each of these operation modes requires different MAC attributes and energy consumption profiles.

MAC transactions are executed using four frame structures: Beacon frame, transmitted by a ZC for network information and synchronization; Data frame, used for data transfers;

Acknowledgment frame, used at various levels to confirm successful frame reception; MAC

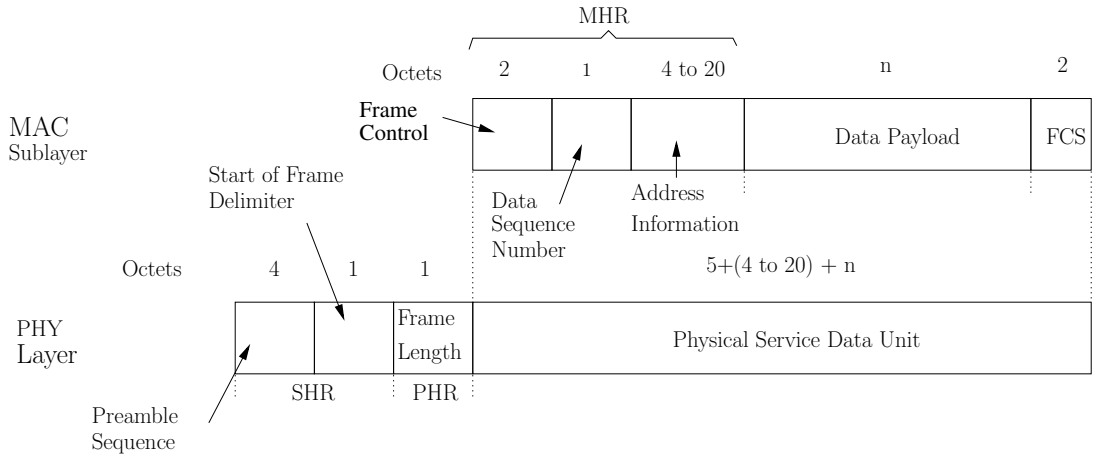


Figure 3.5: IEEE 802.15.4 Data Frame Structure.

command frame, used for handling all protocol entity control transfers (IEEE802.15.4 MAC commands). Fig. 3.4 illustrates the Beacon Frame structure, identifying the fields required for network formation. Fig. 3.5 shows the IEEE802.15.4 Data Frame structure with particular relevance to its number of bytes. In this figure SHR and PHR are the Synchronization Header and the Frame Length for the Physical Layer and MHR is the MAC Layer Header.

The control of the medium access mechanism is of fundamental importance to reduce node power consumption or achieve data delivery quality. Zigbee operation and concomitant power consumption is not, however, exclusively dependent on programming. Node behavior and drawn current are highly dependent on environmental conditions (e.g radio frequency propagation conditions) and network operation. This behavior may be illustrated by Figs. 3.6 and 3.7 where drawn current experimental measurements are shown. Fig. 3.6 shows a Zigbee data transfer with MAC level acknowledgement while Fig. 3.7 illustrates a data transfer with application level acknowledgement. The various device operations are characterized by drawn current and identified by a time slot number. Table 3.2 characterizes all time slots for a CC2530 operating as a ZED. The first time slot - slot 1 - is associated with CC2530 sleep mode. Data processing without radio associated power consumption is illustrated by time slot 2. The CSMA-CA algorithm is executed in time slot 3. In this slot the node repeatedly executes a Clear Channel Assessment (CCA) with a backoff mechanism which presents variable time length. The CSMA-CA time slot

Table 3.2: Data exchange drawn current profiles.

Slot	Time Length	Current
1	Variable. Time between two consecutive transmission	1 μA
2	Application dependent data processing. Processor runs at 32Mhz oscillator that requires 300 μs of start-up time.	0.76 mA
3	CSMA/CA algorithm	27 mA
4	Switching Tx/Rx or Rx/Tx radio module	14 mA
5	Data request transmission. Fixed number of bytes	33 mA
7	Receive data from coordinator.	27 mA
8	Tx of Application layer Ack. Application dependent	33 mA
12	MAC Layer Ack	33 mA

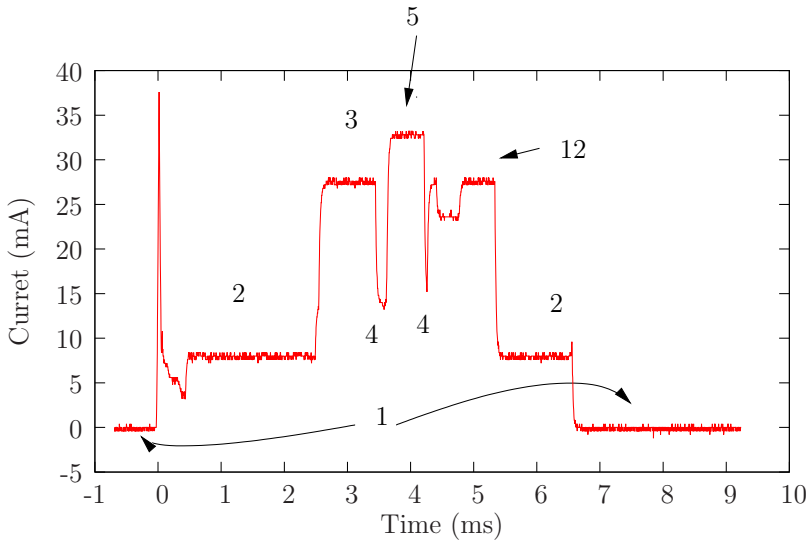


Figure 3.6: Drawn current during data requests with MAC layer Acknowledgement.

behavior may be described by (3.1) [194], [195]:

$$t_{CSMA} = \sum_{i=0}^{aMaxBO} ((rand([0, 2^{min(aMinBE+i, aMaxBE)}] \times t_{aUnitBO}) - 1) + t_{cca}) \tag{3.1}$$

where $rand$ is a random function, $aMinBE$ and $aMaxBE$ are the initial and maximum backoff exponents for successive carrier sensing action, $t_{aUnitBO}$ is the unit backoff period for 2.4 GHz frequency band ($aUnitBackoffPeriod$ in IEEE 802.15.4 definition), $aMaxBO$ is the maximum number of times the CSMA mechanism is repeated after the first occupied channel occurrence and t_{cca} is the radio module listen time for a channel assessment. Prior to data transmission, nodes execute a radio wave carrier sense to determine if other nodes are transmitting. The first CCA is executed after a random backoff

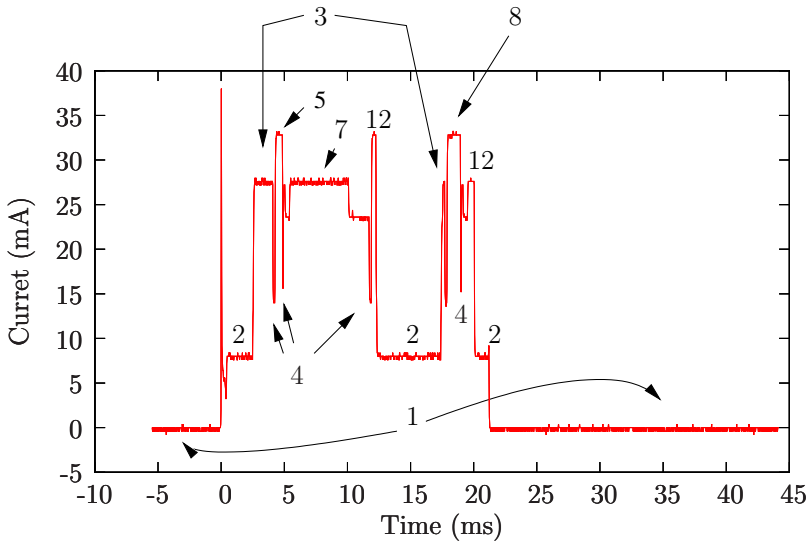


Figure 3.7: Drawn current during data requests with Application layer Acknowledgement.

period. If the radio channel is busy, a new backoff period is measured until a new CCA is executed. This action is repeated $aMaxBO$ times and a communication failure reported to higher layers if the radio channel is never considered available. Time slots 4 have fixed length and are related to radio module switching from receiving to transmitting mode and vice-versa. Data request is transmitted in time slot 5 whose length depends on the fixed number of bytes this specific frame entails. Time slot 12 is also has a fixed time length and is related to sending or receiving MAC level acknowledgment without the CSMA-CA mechanism. The time to transmit or receive a number of bytes, illustrated by time slot 7, is given by [194]

$$t_{TxRx}(n) = \frac{8\dot{(}SHR + PHR + MacHr + n)}{dr} \tag{3.2}$$

where SHR, PHR are the Synchronization Header and the Frame Length for the Physical Layer, MHR is the MAC Layer Header, dr is the data rate of 250 kbps for the 2.4 GHz frequency band and n is the number of bytes to send or receive. SHR, PHR, MHR meaning is illustrated in Fig. 3.5. Time slot 8 shows application layer acknowledgment transmission.

3.4 Network Formation and Access Mechanism

Wireless networks nodes are micro-controller based devices. If low power based operation is required, micro-controller devices must be able to enter in a mode of low current consumption. ZEDs use this mechanism to lower their current consumption. When entering in sleep mode, the processor shuts down its high frequency oscillator together with peripherals and the processor core. Returning from low power modes is executed by a hardware interrupt mechanism. This interrupt driven wakeup may be based on internal timers or external hardware. In the first case, a low frequency oscillator is used whereas in the latter, external actions are required. Also, when using timers with running oscillators the low power current is larger than if external interrupts are used. External circuitry, however, frequently entails large current consumption or is based on a mechanical intervention (e.g. pushbuttons). Even for such cases where external actions wake up the processor, some systems require time synchronizations that lead to frequent communications with a master. Nodes must therefore periodically wakeup on internal timer interrupts to acquire and process data and may eventually communicate with their router or coordinator. Different periodicity and wake up mechanisms lead to different consumption profiles.

In a disconnected operation the device only joins the network when it needs to communicate. Connected operation mode requires that a join procedure [196], [197] is executed once by the nodes. The Join procedure is therefore mandatory so that a node may be part of the Zigbee network. By joining the network, a device associates with other device that is already part of the network. A device that has other devices associated with it plays the role of a coordinator to those devices. Prior to joining the network, each node must execute a network scan, thus choosing a suitable parent by using project design parameters. After potential parent identification, nodes issue an association request command frame. Parents must then determine if requesting nodes may join and send a response frame accordingly. If joining is successful, the response frame contains the short address that the device will use to be identified within the network. Fig 3.8 illustrates protocol behavior for a Join operation. As shown, joining is an end device initiated procedure that sends an Association Request to a ZC. This procedure entails an Association Request action from the ZED as well as an Acknowledgement (Ack) reception. After a *aResponseWaitTime* [198] delay period, a Data request is executed to determine if the join request has been accepted. The answer is issued using an Association response. ZEDs may sleep after sending the join request. In this case, ZEDs must wake up to send a data request

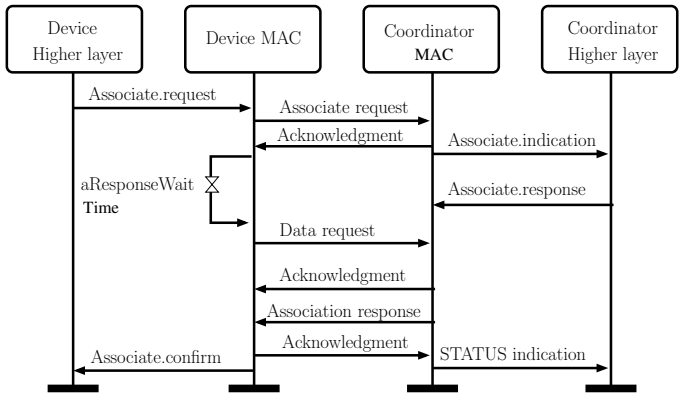


Figure 3.8: IEEE 802.15.4 Join procedure sequence diagram.

for the join procedure and wait for ZC response and Acknowledgments. The *aResponseWaitTime* is an IEEE 802.15.4 defined MAC specification attribute that determines the maximum time a device shall wait for a response to a command frame. The fixed-time $aResponseWaitTime$ is computed by $32 \times aBaseSuperframeDuration$ [198], [172], thus entailing a 491.52 ms wait time.

Fig 3.9 illustrates the Active Scan procedure sequence. Active Scans [199], [200] consist

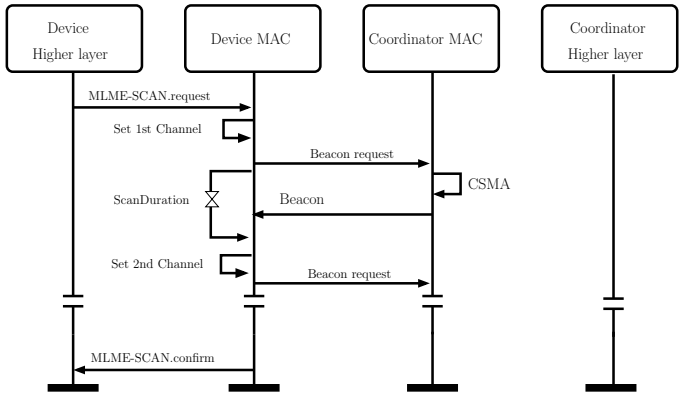


Figure 3.9: IEEE 802.15.4 medium Active Scan procedure sequence diagram.

on a sequence of beacon requests, using MAC frame commands, to determine the existence of network coordinators or routers using different radio channels. This operation is executed with the radio frequency module turned on for each channel scan a period of time *ScanDuration*. Both Fig. 3.8 and Fig. 3.9 show that network joining may entail a prolonged turn on time for the radio frequency module which leads to an increase of energy consumption. Join procedure is fundamental for protocol operation as nodes use

it to acquire a valid network address. Only after acquiring a network address is protocol operation possible.

In Fig. 3.10 Active Scan and network join procedures current consumption are illustrated. In non-beacon Zigbee networks a node wishing to join the network first searches

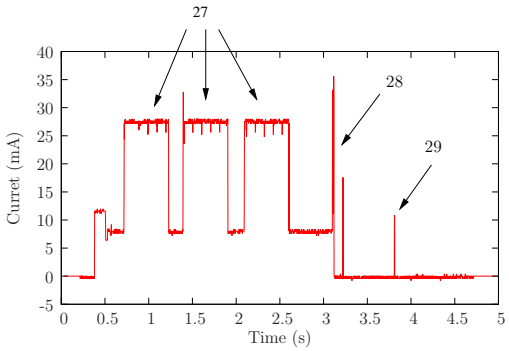


Figure 3.10: Active Scan and Association Zigbee device absorbed current.

for the best parent. The best parent is selected by issuing a beacon request using a Command Frame and listening for answers, thus maintaining radio module in Rx mode for a period of time. This action may be repeated for several radio channels and the radio wake time is variable. The best parent choice is a system option, but usually the chosen is the one with the strongest radio signal. A node must therefore remain in active Rx mode for a longer period of time as shown by time interval 27 in Fig. 3.10. This time interval is dependent on IEEE 802.15.4 defined constants as shown in (3.3).

$$t_{Scan} = \sum_{FistChn}^{LastChn} aBaseSuperFrameDuration(2^{ScanDuration} - 1) \tag{3.3}$$

where $aBaseSuperFrameDuration$ is one of this constants and represents a base time of 15.36 ms in the 2.4 GHz frequency band [198], [172], t_{Scan} is the protocol defined $ScanDuration$ parameter that allows software control over scan time duration, $FistChn$ and $LastChn$ are respectively the first and last channel of the scanning mechanism. The time slot 28 shows the required time duration to execute a join operation. Table 3.3 summarizes network joining time slot description. Active Scans, network Joining, Zigbee device object binding and data transfers are used in the protocol for normal operation. Radio frequency communications are, however, prone to failure and are highly dependent on environmental conditions [201, 202] and on their own operation as collisions often occur. Communication failure may be addressed as network abnormal operation. Com-

Table 3.3: Active scan and join mechanism drawn current profiles.

Slot	Time Length	Current
27	Channel Scan	33 mA
28	Join procedure (see Section 3.3)	-
29	Data request (see Section 3.3)	-

munications failure are dealt by Zigbee that provides protocol dispositions for abnormal operation. The protocol provides a number of commands that allow network rejoining and announce of missing the network. Both normal and abnormal operation are addressed differently by ZED and Zigbee Routers. A router may send and/or forward data through any router within its radio hearing distance that is part of the same network. A missing router may be replaced by one of its peers within earing range. ZEDs may not operate this freely even though they may be at hearing radio distance from many routers. A ZED may only communicate with its parent. Losing a parent is therefore, from the ZED point of view, catastrophic. If a ZED is unable to exchange data with its parent, it must find another parent so that the connection with the network may be maintained. A protocol network rejoin process is therefore required.

All Zigbee devices must be built to comply with these procedures. Zigbee defines application objects to implement the desired functionalities. Two nodes with linked functionalities share common application object structures and communications are possible by encapsulating these data structures into MAC data frames. The process that connects two nodes application objects is described as binding. Binding is implemented through the procedure illustrated by Fig. 3.11, that illustrates a data request sequence diagram that fulfills WSN purpose of data transfer between nodes.

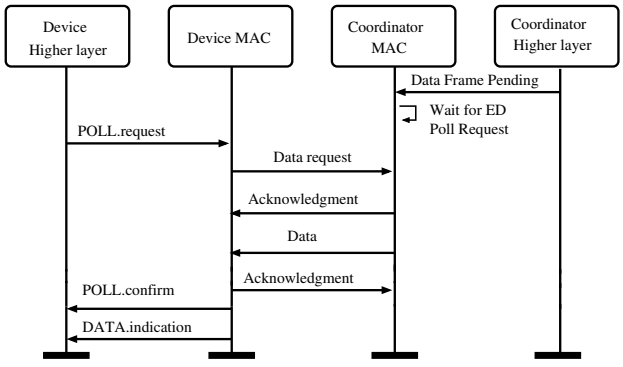


Figure 3.11: IEEE 802.15.4 data request procedure sequence diagram.

3.5 Node Identification and Routing

After joining the network, a node has the ability to communicate with its parent. The purpose of building a network is the ability to relay data from one node to any other node. The process by which data is relayed throughout the network is identified as routing and is strongly related with the node addressing scheme. In this section, Zigbee 2007 and Zigbee-Pro addressing schemes are described. Addressing and routing are implemented by the network layer of the OSI model and are therefore defined by the Zigbee protocol. In accordance with the network possible topologies described in Section 3.2, each node is connected to a parent device from which it is known as its child. Only routers may have children and are able to relay data for them. A router may have children that are also routers and have children themselves. A ZED lacks routing capabilities and may not therefore have children of its own.

3.5.1 Tree Addressing in Zigbee 2006 and 2007

The Tree Addressing scheme proposed by the Zigbee 2006 and 2007 version allocates node addresses sequentially. In this scheme, the ZC provides to the children that may be potential parents a sub-block of network addresses. These parents assign addresses to their children. If Zigbee *nwkUseTreeAddrAlloc* attribute is set to true, the Application Layer (APL) uses the Tree Addressing scheme. The ZC is the root of this scheme and is placed at depth zero. The depth parameter can be described as the minimum number of hops required for a frame to reach the coordinator at depth zero. Tree addressing defines the number of children a router may have as well as how many of those may be routers. Router address is computed by [198]

$$Cskip(d) = \begin{cases} 1 + C_m(L_m - d - 1) & \text{,if } R_m = 1 \\ \frac{1+C_m-R_m-C_m \times R_m^{L_m-d-1}}{1-R_m} & \text{,if } R_m > 1 \end{cases} \tag{3.4}$$

where C_m is the number of children a parent is able to accept, R_m is the number of accepted children with routing capabilities, L_m is the network maximum depth, d is the depth of the device within the network. At a determined depth the first child router receives an address which is incremented by one from its parent address. The following devices placed at the same depth (and therefore children to the same parent) receive an address that is a multiple of the parents $Cskip(d)$ value. A value of $Cskip(d)$ equal to zero means that the device cannot accept any children. The address of a ZED is obtained

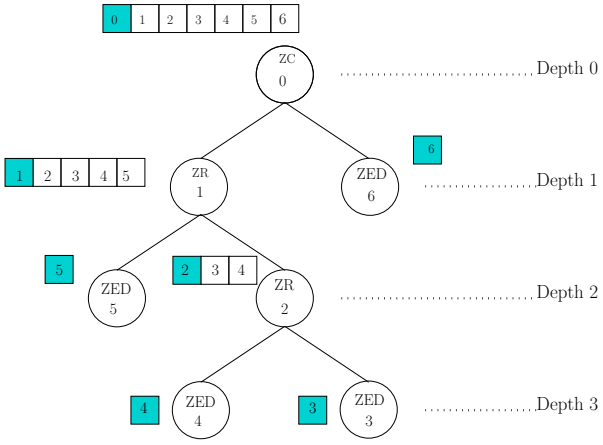


Figure 3.12: Tree addressing scheme with $Depth = 3, MaxRouters = 1, MaxChild = 2$.

by

$$n^{th}ZED_{ad} = P_{ad} + Cskip(d) \times R_m + n \tag{3.5}$$

where P_{ad} is the parent address and $n^{th}ZED_{ad}$ is the address of the n^{th} node. The computed $Cskip(d)$ parameter may define the routing mechanism. When a device is to relay a message on behalf of another device, frames may be forward upward to the parent or downward to one of the children. The message is to be relayed to a descendant node if

$$N_{ad} < Dest_{ad} < N_{ad} + Cskip(d-1) \tag{3.6}$$

where N_{ad} is the nodes address and $Dest_{ad}$ is the destination address. The destination address within descendants nodes can be computed by

$$\text{Next hop address} = N_{ad} + 1 + int\left(\frac{Dest_{ad} - (N_{ad}+1)}{Cskip(d)}\right) \times Cskip(d) \tag{3.7}$$

where *int* is the function that returns the integer part of a number within brackets.

3.5.2 Stochastic Addressing in Zigbee Pro

In Zigbee Pro, a random scheme is implemented for address assignment [198]. Short addresses are randomly assigned to nodes and a device announcement is broadcasted to the network. If an address collision is detected, a conflict warning is also broadcasted to the network so that all conflicting nodes are forced to repeat the short address generation. ZEDs do not participate in the address conflict resolution. Parents (routers) whose children end devices have a conflicting address are responsible for the address reassignment. End

devices must issue a device announcement after receiving the new address and thus re-initiate the conflict resolution process.

The tree routing mechanism is not feasible in the stochastic addressing scheme. The Ad-Hoc On-Demand Distance Vector (AODV) algorithm is therefore proposed by the Zigbee definition [198]. In this scheme, a discovery address table and a routing table are defined for each node. If a node is to send data to another node, it must first learn the destination address. Also, the remaining nodes of the network must know how to forward data between the source and destination nodes. A Route Request (RREQ) frame is issued

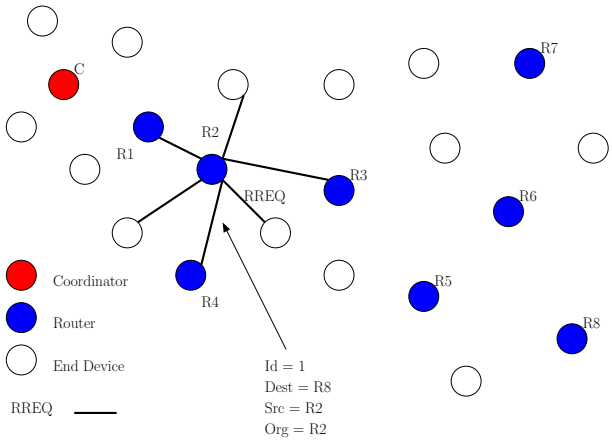


Figure 3.13: AODV routing establishment mechanism. Route request initiator where router R2 starts a routing establishment to node R8.

by initiator node. Fig. 3.13 illustrates this procedure where node R2 broadcasts a RREQ to all its neighbors. A RREQ frame payload contains the nodes destination and originator addresses (Org and Dest) as well as the request identification (Id). Each receiving node extracts from the received MAC frame the message originator and the link cost. The link cost is, in the case of Z-Stack, the radio strength of the received signal in which a low cost link is associated to a high energy receiver signal. The discovery address table is built with these data. Upon reception of a RREQ, each router repeats it to its neighbors, updating the cost field of the generated frame. The process is repeated until the destination address is reached. Fig. 3.14 illustrates this process where node R3 rebroadcasts the RREQ packet. In this case node R2 ignores the received request. The destination node may receive several RREQ frames but keeps the one with the smaller associated cost. A Route Repply (RREP) frame is transferred in unicast mode by the destination node R8 back to the node from which it received the lowest cost RREQ. All nodes in the reply path create a route table

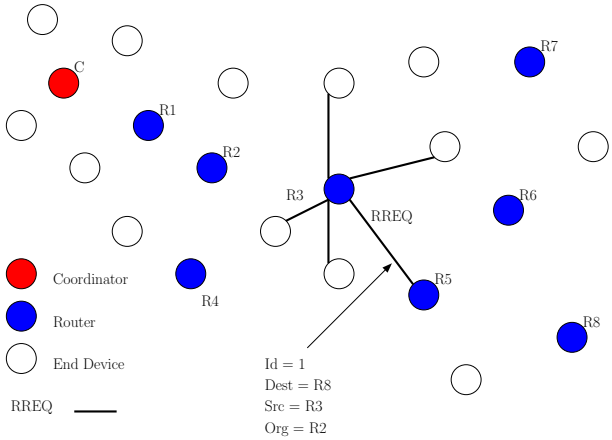


Figure 3.14: AODV routing establishment mechanism. Route requests are broadcasted throughout the network creating discovery table entries.

entry with the address stored in their discovery table. The discovery table is afterwards discarded while the routing table is kept. The AODV algorithm is implemented with a

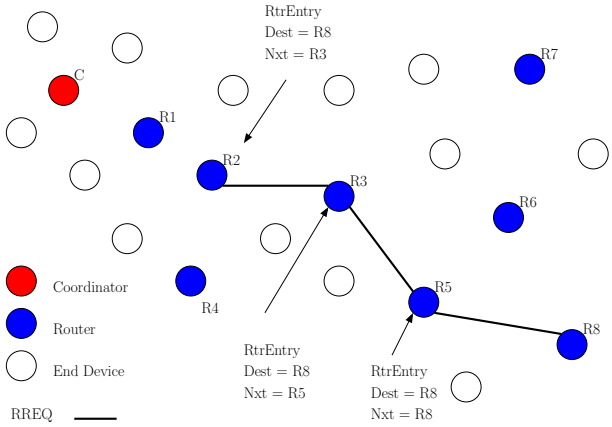


Figure 3.15: AODV routing establishment mechanism. Route reply requests are unicast through the nodes listed at the discovery table.

self healing mechanism by tracking transmission failures along a path within the network. A Route Error (RERR) command frame is originated by a node that loses the connection to the next hop in its routing table. The RERR is broadcasted back through the routing path, thus warning all nodes about the invalid routing table entry. A rediscovering of the route is executed when a data packet arrives for the destination address associated with that route. If the rediscovery mechanism fails, a RERR is sent back the path to the

requesting node that is in turn responsible for a new RREQ process.

Stochastic Addressing presents a number of advantages over Tree Addressing, but entails the use of tables to keep track of network and discovery routes. Though routing tables are expensive in terms of memory use, a number of solutions may be implemented to reduce this memory dependency. A route expiry mechanism is implemented so that an unused table route entry may be tagged as such. An expired table route entry is elected for deleting in case of device memory exhaustion. If a network is to be used as a data collector for a number of nodes, a central concentrator node is the recipient of all data. The concentrator may be used as a gateway for sharing data with other systems (e.g Internet). This topology, addressed in Zigbee as a Many-to-One routing scheme, is extensively used. A RREQ with a Many-to-One option is implemented so that a concentrator node may broadcast such a command. A router that receives this route request creates a route table entry to the concentrator and stores the address of the message sender as the next hop in the route. Also, source routing may be used to prevent the use of routing tables. In this mechanism, the concentrator issues a message with full path information to the destination. In this case the routing process is managed by the application layer. Moreover, if the ZC is the data sink, all data requests are routed to the only fixed network address - address 0x0000.

3.6 Data Profiles and Application Layer

The previously described layers create and maintain the communication network between physical devices through a wireless link. The application layer is used to implement connections between logical objects that co-exist within the physical devices. The application layer implements the Zigbee purpose of data transfer between those logical objects by defining data containers. A data container is implemented by a descriptor that is defined by a common structure known to all nodes. Descriptor exchange between nodes is the key to understand Zigbee operation.

One may illustrate Zigbee as being defined by computer programmers so that other computer programmers are able to understand it. A descriptor exchange between two nodes can be compared to a memory copy between two memory blocks within a microprocessor. Fig.3.16 illustrates an example of this process. In this example, the memory block A is copied to memory block B. The information this process requires is the initial address of the original block memory, the initial address of the destination block and the

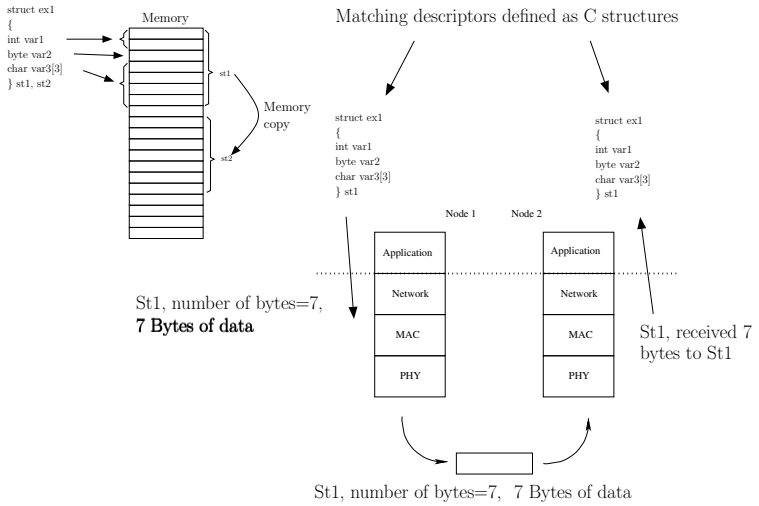


Figure 3.16: Zigbee descriptor exchange compared to memory block copy of C programming language.

number of bytes being copied. Data contents are therefore irrelevant. If block memory A is defined by a C programming language (C) structure data type, a meaningful copy may be processed if memory block B is defined by the same structure data type and if the number of bytes copied is the same as the size of the structure. Descriptors in Zigbee are defined by structure data types in C, and data exchanges between nodes identify the destination structure and the number of bytes being transferred. This process is also illustrated in Fig. 3.16. The destination node application layer receives a number of bytes to a structure St1. The received data is copied to the destination structure memory space thus completing the data transfer. The protocol definition is based on a common data structure that is replicated as many times as there are logical objects in a device. Each Application Object identified in Fig. 3.1 is a communication entity within a device that receives the data structures. These Application Objects are known as Endpoints and are conceptually similar to TCP ports in TCP/IP networks. As illustrated in Fig. 3.17, an application sends its data identified by the originator endpoint and by the destination address. Node short address, long address and Endpoint are defined within the destination address structure. Within each Endpoint several data structures may be defined. Zigbee names the defined data structures as Clusters. The SendData() command must therefore also identify, within the Endpoint, to which of these structures is the data addressed to. This information is specified by the sender using the ClusterID. Fig. 3.18 illustrates the

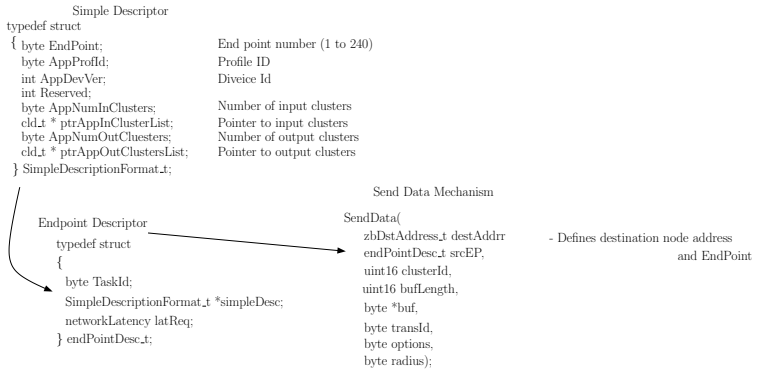


Figure 3.17: Zigbee Simple Descriptor definition.

communication between nodes with matching Endpoints and Clusters. In this example, a FFD (Node A) is implemented with a number of Endpoints. Each of Nodes A Endpoint (EP)s is associated with an EP of Node B, C and D. If a message from Node A EP1 to Node B EP4 is transmitted, the matching Cluster is transferred. Also, a Zigbee Profile is a group of clusters with a logical and/or functional correlation. Before start-

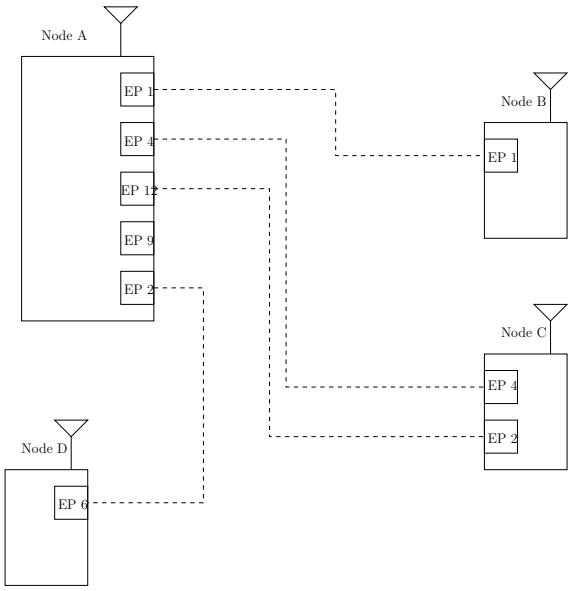


Figure 3.18: Binding example with a complex node (FFD) and three simple nodes (RFD).

ing communications, each node must know the physical device address of the destination node. This requirement is shown by the SendData() function example in Fig. 3.17 whose first parameter is the destination node physical address. The process of identifying the

node address within which a matching cluster is implemented is defined as Binding. Bindings establish logical connections between matching endpoints/clusters by implementing Binding Tables. Moreover, binding allows applications to send data without knowing the destination node physical address. The physical destination address is determined by the Application Sublayer from the binding table. The binding table can establish one to many links, thus allowing data transfer to multiple physical addresses without the Application Layer intervention.

Several mechanisms can be used to build a binding table. The process of adding a record to a device binding table is initiated by sending a specific message to the ZDO. A node can broadcast a specific message announcing its binding available clusters. All devices with a matching cluster respond also with a specific reply message that is handled and validated by the issuer ZDO. The initiator stores the novel binding record in its binding table. This process is named Automatic Binding. An intermediate node - the Coordinator - can store bind requests from two or more nodes. After receiving the second bind request, the coordinator matches it with the first request and, if the requests are compatible, issues a command so that the device that supports the client side of the cluster adds a record to its binding table. This process is known as centralized binding.

3.6.1 The Zigbee Device Object and the Zigbee Device Profile

The ZDO is a special endpoint that is common to all Zigbee devices and numbered endpoint zero (0). Within the ZDO a number of clusters are defined that provide support for node and network management. This special group of clusters is named Zigbee Device Profile (ZDP). Based on the ZDP, the ZDO provides a number of services that operate in master slave mode. Both master and slave are implemented in the same device as the application layer can request a service to a ZDO from another device or a device may be queried by other device and its ZDO must mandatorily provide a response. All ZDO must implement a response service to an identification request, meaning that all devices are masters for the identification service. An identification request is answered with a node descriptor.

These services are provided by the Zigbee Device Profile and are associated with discovering, maintaining and configuring the network. A client-server mechanism is implemented where server side services are mandatory. All devices must respond with their short and long addresses and their node and power descriptors when requested, thus exposing device

level characteristics. Moreover, implemented services (Endpoints) must also be exposed upon request by a node with authority. In this case, simple descriptors and active endpoints are transmitted when queried. Table 3.4 shows the Zigbee services with mandatory response from all network nodes. The number of bytes required by each service is identified in this table as well as the type of service and the node which is the responder. For sleeping nodes the parent node can assume the child's role and respond on its behalf.

Table 3.4: Implemented ZDO services with number of bytes and service type.

Service	Number of bytes	Type	Responder
Network address	12	Device Discovery	Parent
IEEE address	12	Device Discovery	Node
Node descriptor	17	Device Discovery	Node
Power descriptor	6	Device Discovery	Node
Device announce	12	Device Discovery	Parent
Simple descriptor	15	Service Discovery	Node
Active end point	8	Service Discovery	Node
Match descriptor	6	Service Discovery	Node

3.6.2 The Zigbee Cluster Library

The Zigbee Cluster Library (ZCL) is a list of clusters and cross-cluster commands that are used by Zigbee to direct device standardization. The clusters within the library can be used to build private profiles and are the building blocks of the Zigbee Public Profiles. An example of a public profile is the Home Automation profile that implements a number of logical devices that enables automatic home control.

3.7 Proposed Device Details

The measurements for this chapter have been obtained with a printed circuit board which includes a TI MSP430FR5739 microcontroller, a TI CC2530 microcontroller and a radio frequency module for 2.4-GHz IEEE 802.15.4. In this case, the MSP430 is programmed to remain in low power. The CC2530 has been programmed using Texas Instruments Zigbee protocol (Z-Stack [173]). Z-Stack is a Texas Instruments IEEE 802.15.4/ZigBee compliant protocol stack. Z-Stack is a highly parameterizable solution to implement Zigbee networks;

several operation options are available in Z-Stack.

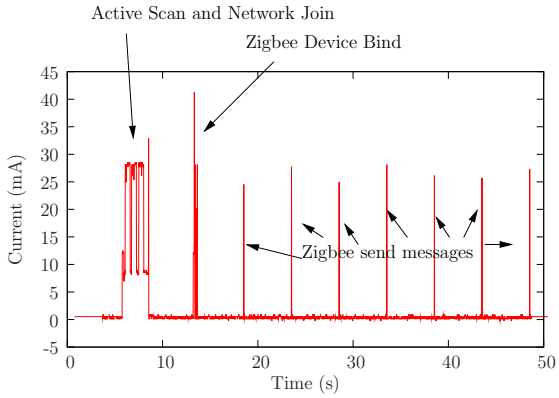


Figure 3.19: 50-s measurement of a ZED device operating a Zigbee compliant stack illustrating absorbed current. Obtained with an experimental setup, measuring drawn current of a ZED in a simple two node network. The setup measures the voltage drop on a 1Ω resistor in series with the system voltage line.

A ZED node full operation with Active Scan/Join, Bind and data transfer actions is illustrated in Fig. 3.19. The shown operation is executed without superframe because it is not currently supported by Z-Stack. Fig. 3.19 has been obtained by measuring current consumption of a ZED in a simple experimental setup with one ZC. Measurement has been obtained with a stable power supply provided by a 3 V battery. As illustrated, a data request leads to less energy consumption than the Join/Active Scan procedures. Moreover, within the shown 50-s test, energy consumption required for joining the network using an active scan procedure is the system main energy constraint.

Chapter 4

System Architecture and Simulation Model Description

Electromagnetic theory has been applied for a long time in a number of devices. Voltage transformers and current sensing are examples of this widespread use [155, 156]. In this work, a magnetic power generator is implemented as a contact-less energy harvesting system for a Zigbee wireless module. In this chapter the device architecture and design options are presented. The proposed device is illustrated in Fig. 4.1. As described in [126],

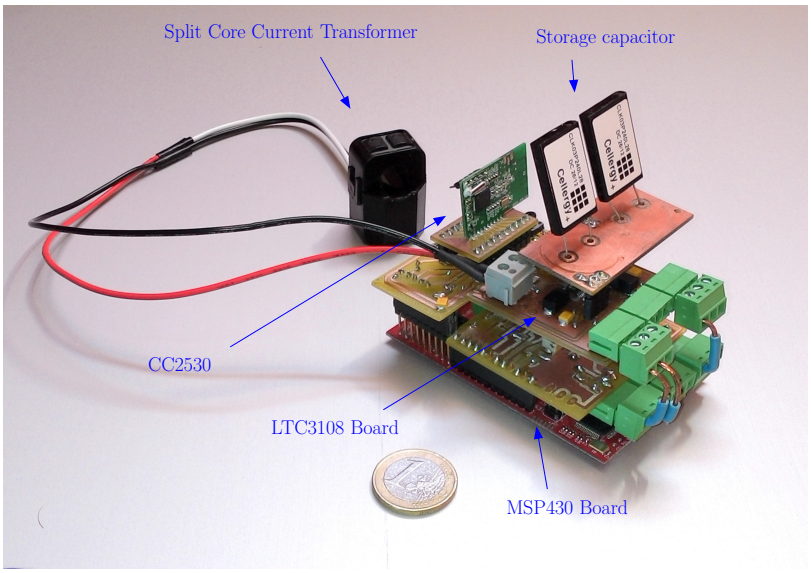


Figure 4.1: Implemented device photo illustrating component relative dimensions and identifying its modules.

a Harvester-Store-Use device is implemented.

4.1 System Architecture

The magnetic power generator is implemented with a well known SCCT, whose main applications are related with AC current measurements. The magnetic field generated by alternating current flowing through a power line is harvested to power a device. The SCCT generates a current from the induced electromotive force. A step-up converter circuit with integrated power manager is able to scavenge energy from this induced current. Moreover, the proposed device implements a number of energy management strategies, so that a complex wireless communications protocol is sustained. The implementation of a Simulation Program with Integrated Circuit Emphasis (SPICE) [203] model to simulate the SCCT generated output voltage is described. This model is used with the step-up circuit model to determine the implemented device behavior. The SCCT model has been obtained through an experimental setup while the step-up model is obtained from the manufacturer. The toroidal coil scavenges energy from power cables connecting electric charges (e.g. electric motors, lighting, computers). All system components use mature technology thus providing a market ready technological solution to be implemented. The system architecture is illustrated in Fig. 4.2.

Powering a WSN node directly from the SCCT is not a feasible solution because a non-stable AC current is generated. A conversion circuit is therefore required. Also, a capacitor storage management solution has been implemented. The SCCT generated current from the induced electromotive force is rectified, stored and regulated to a MCU suitable voltage operation. Following the harvesting device requirements identified in Chapter 2, the power management unit is able to handle very low feeding power and it is self-starting. The implemented AC/DC conversion circuit is based on the LTC3108 [204] from Linear Technology. This device main uses are related to energy harvesting from voltage sources such as TEGs (thermoelectric generators), thermopile and small solar cells. A study of the proposed harvesting device is presented in [32] and its implementation described in [29]. For the harvesting system development, a SPICE model for the SCCT has been implemented and its results confronted with the device measurements, thus demonstrating system viability [29], [32]. The proposed SCCT SPICE model has been used together with the Linear Technology LTC3108 SPICE model to simulate full system operation. The system is designed so that it is able to execute sensing and communication

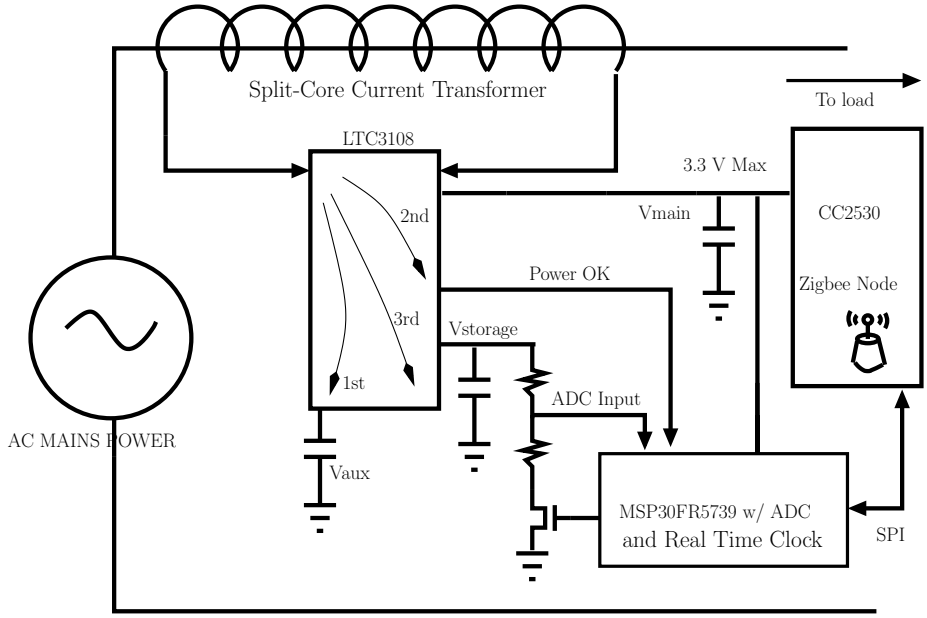


Figure 4.2: Magnetic power generator with AC/DC conversion and storage systems architecture.

tasks.

The proposed system uses the LTC3108 as a bootstrap charge manager for the device energy storage capacitors. The capacitors are identified by their line names V_{aux}, V_{main} and $V_{storage}$. The energy thus stored is thereafter used to power a radio frequency (RF) module, a low power device manager and two MCUs. The LTC3108 implements a charge pump from the AC SCCT rectified voltage. The rectified voltage operates a booster circuit that regulates the desired stable output value. LTC3108 operation voltage uses the stabilizing capacitor connected to V_{aux} which as it reaches 2.2 V allows output capacitor connected at V_{main} to begin charging. This operation initiates energy transfer from input voltage to the MCU powering capacitors. V_{main} charges to a maximum design defined voltage which for this system is optionally 2.5 V or 3.3 V. Once reached maximum V_{main} voltage, $V_{storage}$ is allowed to start charging up to 5.25 V. $V_{storage}$ can be used as an energy reservoir to power V_{main} when source energy is missing.

The behavior shown by the lines V_{aux}, V_{main} and $V_{storage}$ can be used to characterize the harvesting system operation. System characterization may be related to the mains current value. The SCCT source must first be able to power V_{aux} line and therefore start LTC3108 operation, which corresponds to the first mains current characteristic level.

During the charging phase, the device scavenges energy from load power cables until device manager operation is possible. Energy stored at V_{aux} is used to power LTC3108 internal circuitry. The first operation charges V_{aux} up to 2.2 V, where the capacitor V_{main} starts charging. V_{main} voltage is hardware configured and for this work is set to 3.3 V. A mains current under this first characteristic value doesn't allow system startup. Higher mains current level allows the V_{main} capacitor to charge. A second characteristic mains current level is therefore determined by V_{main} full charging to a specified value. Once V_{main} is fully charged, the excess energy is used to charge $V_{storage}$ that in turn may be used to recharge V_{main} if harvested energy fails. $V_{storage}$ may be allowed to charge up to 5.25 V, grounding all excess harvested energy. A third characteristic level for mains current can be identified as $V_{storage}$ full charging ability. Fig. 4.3 illustrates the successive capacitor charging system operation.

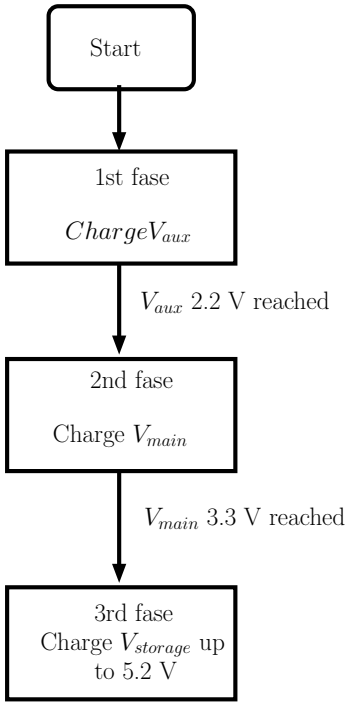


Figure 4.3: Flow chart of successive capacitor charging system operation.

The RF module is implemented with a CC2530 [56] Zigbee device, while the system power manager uses a MSP430FR5739 [205] MCU, both from Texas Instruments. A modular device is proposed which allows diverse implementations architectures (e.g. different radio frequency modules). System architecture is therefore implemented with two MCUs,

thus following the Zigbee Network Processor (ZNP) concept described in [206]. ZNPs are important development tools that allow small time to market for Zigbee devices as they enable designers to implement the code in an environment that is free from the the protocol complexities. With this option, the developer code is implemented in a MCU while the Zigbee operation is executed in a different device. A set of commands implement an on-board communication interface (e.g SPI, RS232). The proposed device has been designed with a SPI interface and the set of commands shown in Table 4.1. The

Table 4.1: Implemented Monitor end Test protocol commands.

Command	Operation	Direction
Start device	Initiates Z-Stack	MSP430 → CC2530
Connect to the network	Initiates network join	MSP430 → CC2530
Transfer measured data	V_{main} and $V_{storage}$ are transferred	MSP430 → CC2530
Active Scan	Executes an Active Scan operation	MSP430 → CC2530
Sleep mode	Enable Deep Sleep mode	MSP430 → CC2530
Sample rate	Change power manager sample rate	CC2530 → MSP430

commands are transferred with the transport protocol defined in [207]. The Monitor and Test protocol has been developed by Texas Instruments to exchange messages between Zigbee modules using a RS-232 interface. The proposed device implements the transport layer of this protocol over the SPI interface.

4.1.1 Capacitor Choice Issues

Capacitor choices have critical influence on system behavior. Both capacitor value and leakage current are relevant design options with critical influence on charge times. Moreover, in the absence of mains current, capacitor leakage current is relevant for system lifetime even without radio frequency activity. The choice of capacitor devices is therefore limited to those with low leakage characteristics. Devices presenting high capacitance values are usually associated with electrolytic or tantalum capacitors that present high leakage currents. Leakage capacitor current is addressed in [160],[208]. As a design option, Vishay Aluminum capacitors [209] are used as recommended in LTC3108 data sheet. Large capacitance values are obtained from Cellergy [210] that presents the desired characteristics of high capacity and low self discharge current. The capacitance choice value is

justified by device operation requirements. These values are addressed in the subsequent sections and chapters.

4.2 Simulation Model

Applying Ampere's Law allows to determine the current generated by the magnetic generator, given by:

$$\oint H.dl = i_p - Ni_s, \tag{4.1}$$

where H is the magnetic field density, i_p is the primary current (in this case the input current of the load), N is the number of turns in the transformer secondary circuit and i_s the current generated in the secondary coil. If B is the magnetic flux density inside a core with permeability μ, Equation 4.1 can be written as

$$\phi = BA = \mu HA = \mu A(I_p - nI_s)/\ell, \tag{4.2}$$

where ϕ is the magnetic flux and A is the cross-sectional area of the core. According to Faraday's law, the voltage at the load R terminals, connected to the secondary winding, is approximately equal to the induced electromotive force (EMF) which, in the frequency domain, is given by:

$$|i_s R| \approx |E| = (N^2 \mu A/\ell)2\pi f(I_p/N - I_s), \tag{4.3}$$

where ℓ is the path length for the magnetic flux, A the cross sectional area of the secondary core, μ the magnetic permeability and R is the secondary charge resistor.

Secondary coil current for a load R can therefore be obtained and thus determine the energy supply to the wireless node. Even though (4.3) analytical resolution may be obtained, a more practical solution can be found by circuit simulation. For this work, a SPICE circuit simulator is used and the SCCT is modeled by a pair of mutually coupled inductors. Mutual magnetic coupling is described in SPICE simulations by coefficient parameter k defined between primary and secondary coils. Modeling the SCCT requires however circuit parameters that are not provided by the SCCT manufacturer. Moreover, SCCT are commonly used in current sensing applications. Their data sheets are clearly made for different uses than those pursued in this work. As shown in Fig. 4.4, that illustrates the manufacturer announced performance for the used device, SCCT applications are directed to high power solutions. Low power (low current and low voltage) behavior

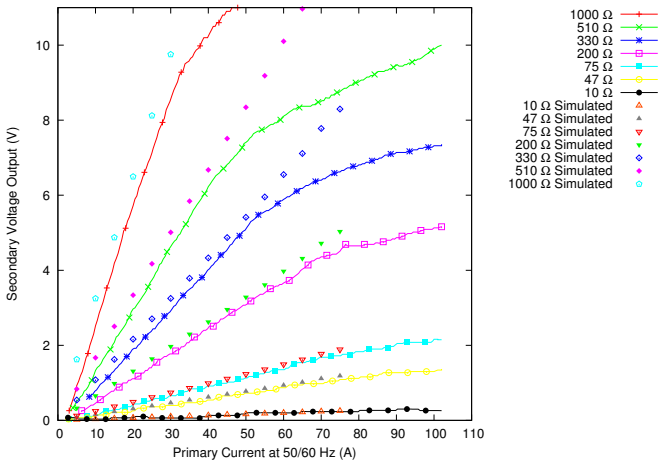


Figure 4.4: Datasheet Split-Core Toroidal Coil I-V curves vs Simulated curves for coupled inductor model.

Table 4.2: Manufacturer given values for SCCT characterization.

Core area	Magnetic flux length	Secondary turns	Core type	Cross section area
5 mm^2	5 cm	3000	Ferrite (Ui:7000 Bs:4300mT)	50 mm^2

is not clearly shown. Also, secondary inductor values are missing from manufacturer data sheet. The use of the SPICE mutual induction model requires primary and secondary induction knowledge. These values are obtained through an experimental setup to determine a first estimate for coil values followed by simulation for fine tunning. The obtained SPICE model has been used in simulated environment for system design and implementation.

The known values prior to experimental setup measurements are shown in Table 4.2. The first estimate for primary and secondary inductor values is obtained analytically from values delivered by the manufacturer and read from the experimental setup. Fine-tuned inductor values are obtained by confronting simulation results with SCCT manufacturer data sheet curves. The coil was placed around a power line with a fixed current thus generating an output voltage. Considering that the transformer universal EMF equation

Table 4.3: Determined values for split-core toroidal core spice model.

First estimate for primary induction value	First estimate for secondary induction value	Model value for primary inductance	Model value for secondary coil inductance
5 μ H	66 H	5 μ H	45 H

for a single frequency circuit can be expressed as

$$E_{rms} = f \cdot N \cdot \phi_{max} \cdot 4,44, \tag{4.4}$$

where E_{rms} is the voltage generated at the SCCT output, f is the circuit operating single frequency, ϕ is the generated magnetic flux and N is the number of secondary turns, the open circuit SCCT output voltage has been measured, allowing the determination of the magnetic flux. Magnetic flux is the link between primary current and open circuit secondary voltage. This link may be expressed by the $N\phi = LI_s$ relation that used with (4.4) allows to determine a first estimate for the primary coil inductance. For the secondary coil induction, (4.5) and (4.6) have been used

$$L = N^2/\mathcal{R}, \tag{4.5}$$

where $\mathcal{R}$ is the magnetic reluctance and L is the secondary coil inductance. Coil reluctance is obtained from Equation 4.6 that relates its value with toroid geometry and core materials.

$$\mathcal{R} = \frac{\ell}{\mu_0 \cdot \mu_r \cdot A}, \tag{4.6}$$

where μ_0 is the magnetic permeability of the air and μ_r is the relative magnetic permeability of ferrite.

Table 4.3 shows the first approach to both primary and secondary inductance coils. The first approach produced a result that is near to actual model values. Fine-tuning for SCCT model inductor values are indicated in Table 4.3 whereas Fig. 4.4 shows datasheet values confronted with simulated values. The model accounts for the V_{aux}, V_{main} and $V_{storage}$ capacitors leakage currents by implementing current sources associated to each capacitor. Also, the current drawn by the charge is addressed by the I_{load} current source. The circuit model is illustrated in Fig. 4.5.

Fig 4.6 illustrates both simulated and measured device operation for 800 mA and 1-A mains power line currents and 330 μF V_{main} and $V_{storage}$ capacitors. Full charging is

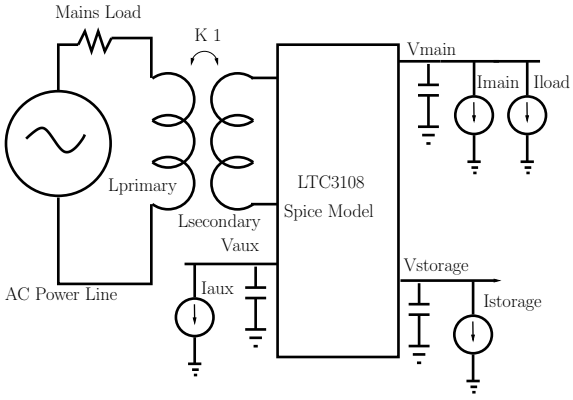


Figure 4.5: Implemented SPICE model for node behavior simulation. Capacitor leakage currents (I_{aux}, I_{main} and $I_{storage}$) are addressed as well as the load absorbed current I_{load}.

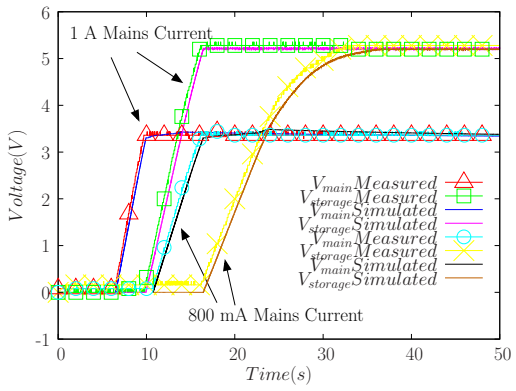


Figure 4.6: Simulated and measured LTC3108 operation for mains currents of 1A and 800mA.

possible for mains currents as low as 800 mA. After $V_{storage}$ full charging, excess current is shunted to ground if no load exists at the device output or the input source is generating more power than what is required by the load. Table 4.5 shows system behavior for several mains currents.

Fig 4.7 illustrates system V_{main} voltage line running Z-Stack and executing 5 s spaced data transfers. Both simulated and measured plots are shown for a system with 330 μF V_{main} and $V_{storage}$ capacitors. These results are obtained for a ZED data request followed by a data transmission of 10 bytes that in [31] are shown to be sufficient to transfer motor operational data. The implemented SPICE model has been successfully used to predict system behavior and justify capacitor choice. Also, the minimum voltage V_{main} is

Table 4.4: Leakage currents for V_{aux}, V_{main} and $V_{storage}$ capacitors.

V_{aux}	V_{main}	$V_{storage}$
5 μ A	20 μ A	20 μ A

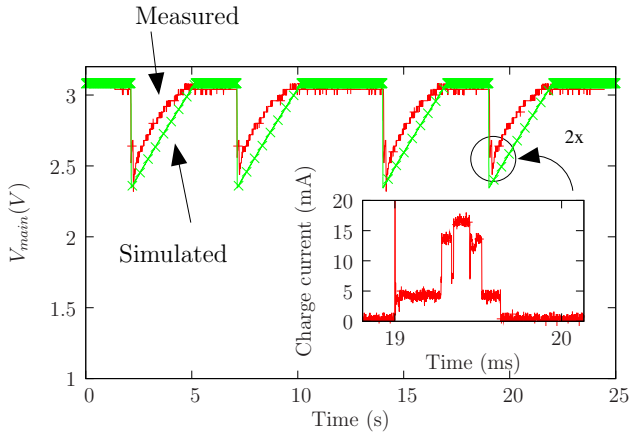


Figure 4.7: CC2530 operating a Zigbee compliant stack voltage line level. Obtained with an experimental setup, measuring current consumption of a ZED in a simple two node network. This setup measures the voltage drop of the CC2530 microcontroller voltage line. Mains current is 1.2 A

maintained over 2 V thus preventing MCU shutdown.

It was possible to maintain the system regular operation for a ZED based only on capacitors. Routine Zigbee data exchange is shown to be possible with the proposed harvesting device. RF communications are, however, constrained by the a number of factors like PCB printed antennas, possibly poor RF design and thick concrete walls. In this context, node radio links to their parents are frequently interrupted. In this event ZED may be programmed with protocol dispositions to Rejoin the network and/or issue an Orphan condition. Also, network reconfiguration or redeployment requires that Join and Bind procedures are executed. The proposed device architecture is not able to handle these events as the charge stored in 330 μF capacitors V_{main} and $V_{storage}$ is insufficient for a long period of radio module turn on time. Sustaining exception procedures is possible by increasing V_{main} and $V_{storage}$ capacitance with supercapacitors. The main consequence of the introduced change is related to the necessary time to charge a large capacitor. Chapters 5 and 6 address this issue.

Table 4.5: Simulated and measured contact-less energy harvesting device. Charging times for different mains current and capacitor values.

Mains Current		V_{main} (3.3 V)	$V_{storage}$ (5.2 V)	V_{main} (3.3 V)	$V_{storage}$ (5.2 V)	$V_{storage}$ (5.2 V)
-	-	330 uF Capacitor		240 mF Capacitor	120 mF Capacitor	120 mF Capacitor (from 4 V to 5.27 V)
400 mA	Measured	Nr*	Nr*	Nr*	Nr*	Nr*
	Simulated	Nr*	Nr*	Nr*	Nr*	Nr*
500 mA	Measured	15.2	42 (3.12 V)	17:30.20	(0.34 V)	(0.34 V)
	Simulated	17	52 (3.2 V)	-	-	14:00:00
600 mA	Measured	7.2	32 (3.76 V)	07:02:35	17:00:00 (4.3 V)	01:20:04 (4.0V - 4.3V)
	Simulated	9.1	36 (3.8 V)	-	-	-
700 mA	Measured	9.0	23 (4.16 V)	05:08:40	10:37:00 (4.7 V)	01:10:20 (4.0 - 4.7 V)
	Simulated	8.9	29 (4.9 V)	-	-	-
800 mA	Measured	4.8	17	03:07:35	05:45:59	01:03:20
	Simulated	5.6	22.9	-	-	-
900 mA	Measured	3.8	19.2	01:36:20	03:19:40	00:41:20
	Simulated	4.9	20.8	-	-	00:43:12
1000 mA	Measured	3.0	13.6	01:30:00	03:10:00	00:35:40
	Simulated	4.2	14.4	-	-	00:32:56
1200 mA	Measured	2.8	10.1	01:09:20	02:18:40	00:24:00
	Simulated	3.23	11.78	-	-	00:23:21

* Neither capacitor reached maximum voltage

Chapter 5

Powering IEEE 802.15.4/Zigbee Compliant Nodes on Harvested Energy

Powering electronic devices is usually associated with a reasonably stable power supply. Portable or other independent devices are commonly powered by batteries that provide a reasonably stable voltage level. The proposed battery-less device relies exclusively on capacitors to power a radio frequency module and a microcontroller operation. Unlike a battery operated device, once current is drawn from the capacitors, voltage may drop thus failing microcontroller powering. In one such system, capacitor discharge control is mandatory. In this chapter the battery-less energy harvesting system operation is analyzed. Capacitor charge is managed so that protocol operation is possible and sustainable. The proposed system operation should meet energy constrains to prevent capacitor V_{main} from discharging bellow the microcontroller minimum operating voltage of 2.0 V. System operation is controlled by the Texas Instruments software stack. Texas Instruments Z-Stack implements an operative system that is referred to as board support package (BSP) [211]. This BSP consists of a hardware abstraction layer (HAL) and an operating system abstraction layer (OSAL). OSAL is a mechanism for task allocation of resources, implementing a cooperative, round-robin task servicing loop where each operation in Z-Stack runs as a task that is capable of communicating with other tasks through a message queue [105]. Original OSAL implementation is developed for battery operated nodes with no short term energy limitations. The Out-Of-The-Box implementation executes Z-Stack

initialization, IEEE 802.15.4 Join and Association procedures as fast as possible without entering low power mode and only then returning to processor sleep mode. This is a natural and desirable behavior for battery operated nodes. For such power supplies, short term energy exhaustion is not a problem and is therefore indifferent to remain on full power mode for one time 1 or 10 seconds. Operating on harvested energy poses different challenges. For one such system, prolonged (10+ seconds) full power operation is impossible because only the energy storage in capacitor banks is available. Unlike battery powered devices, this system is unable to power the Zigbee protocol tasks for prolonged periods of time. To prevent uncontrolled device power down, a return to sleep mode is mandatory within a limited time frame. Energy harvesting for WSN requires a meaningful change in node behavior when compared to battery operated nodes. Nodes running only on capacitor charge must be programmed so that short term energy constraints are met while battery operated nodes have strong long term energy limitations. In this system, short time energy availability is small. On the other hand, in the long run, energy is harvested allowing long term device powering.

In this chapter a change to the original Z-Stack implementation is presented, so that a full Zigbee compliant node is possible to run on harvested power. Changes to the original Texas Instruments software are described. A model that accounts for system operational and environmental parameters is proposed and analyzed. An analysis on the limitations and advantages of the proposed system is presented. The proposed model may be related to the energy neutral operation introduced in [126]. This analysis accounts for the capacitors stored energy. The stored energy is quantified by the amount of voltage variation the V_{main} capacitor suffers. This variation limits the node available energy. Z-Stack operation is tuned so that energy limits are maintained under node operational limits.

5.1 IEEE 802.15.4/Zigbee Task Scheduler for Nodes Running on Harvested Power

Figs. 5.1 and 5.2 illustrate original and proposed OSAL operation, signaling proposed changes with the dotted/red flowchart boxes. The software executes system initialization (see Fig. 5.1) before entering into the stack main loop (see Fig. 5.2). Changes are proposed to the initialization procedure as well as to the main round-robin operation. The system initialization executes start-up to memory allocation system, OSAL basic timer, message queue, power management and task scheduler system. With the proposed mech-

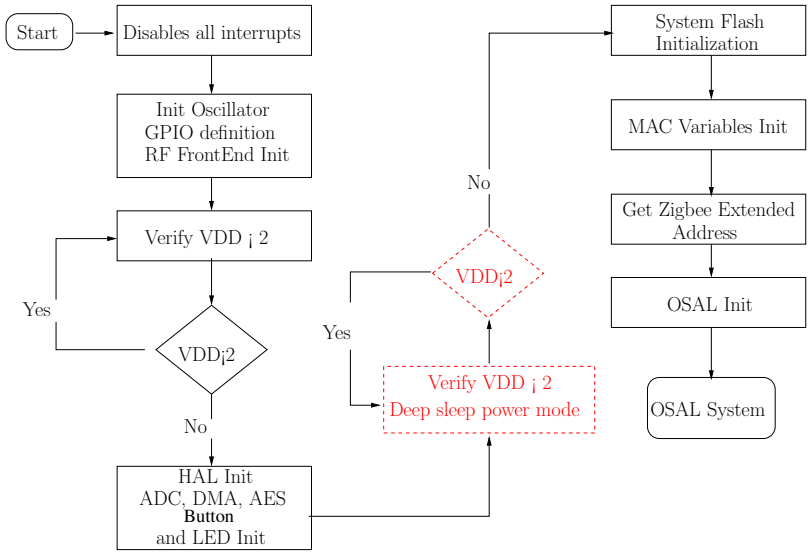


Figure 5.1: Z-Stack Operating System Abstraction Layer initialization procedure.

anism, tasks are executed only if enough energy is available on V_{main} capacitor and the capacitor voltage level is used to determine operation capability. Fig. 5.1 illustrates the initialization procedure where energy availability is tested before system flash initialization because these memory accesses are, together with radio module operation, the CC2530 main current sinks. If the illustrated test returns a valid voltage level for V_{main} capacitor the system is allowed to start OSAL operation.

In OSAL, application processes run based on tasks that must be created in software. This is accomplished by creating an array of tasks and adding entries for each new application process. The OSAL searches for pending events created by tasks and only after they all agree the processor is allowed to sleep. As task execution is done by OSAL without user interference, the task scheduling events must be controlled to prevent processes from running. All tasks are non preemptive and once started run to completion. A typical operation sequence is initiated by an event set by hardware or by the application related tasks. These tasks implement the system main purpose of transferring user data with a defined periodicity reacting also to hardware changes. The remaining tasks get events that have been created by these parent tasks. Moreover, tasks that entail large energy costs are related with data transfers that are all application defined. Users must know at programming time the worst case sink current for each scheduled task. Application level tasks are allowed to schedule events only if energy availability exists. If a communication

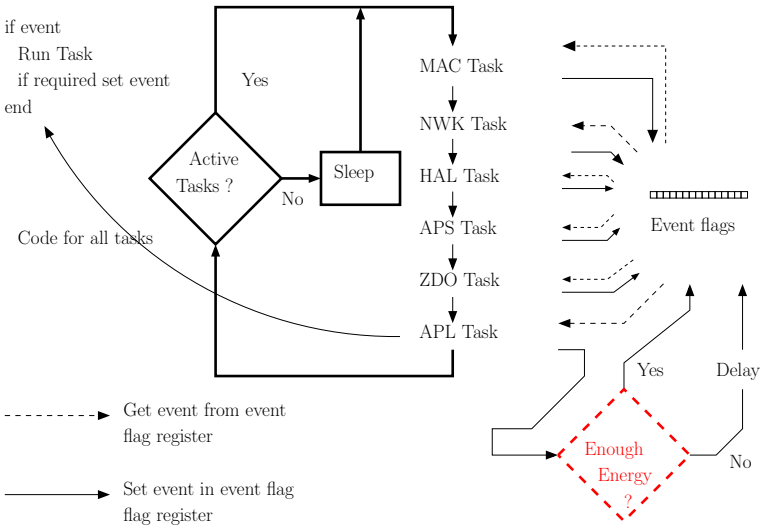


Figure 5.2: Z-Stack OSAL task scheduling mechanism for an example task array. The OSAL reads the event flag register to determine whether a task should be launched. The event flag register is set by each task if required. The processor is allowed to sleep if no tasks have pending events. Application level tasks prevent event generation if no energy is available.

event is not possible, a delay event is set, thus allowing OSAL to place the processor in low power mode.

Fig. 5.2 illustrates the round-robin task servicing scheme, the event scheduling process and the proposed blocking mechanism. Each task searches the event flag register to determine if an event has been generated. If an event is detected the task runs to completion and generates its own events if so required. Further transactions between tasks can be addressed using the messaging scheme also implemented in Z-Stack. Fig. 5.2 shows the implemented task array with Z-Stack. The stack has been configured so that all events are generated by the application defined task and hardware events disabled. The implemented mechanism operates therefore by blocking application initiated task events. The analysis of the implemented algorithm is presented in the next section. Task scheduling is based on a model that accounts for both the harvested and the dissipated energy.

5.2 Model Description and Assumptions

The proposed mechanism operation may be characterized by two operation modes:

Before network join - In this mode at least one Active Scan is required for operation. Because of its impact on V_{main}, Active Scan scheduling is a simple mater of allowing it or not. No parameters are required to evaluate if this action is executed. Active Scan and Join actions are only executed with V_{main} full charge. Moreover, these actions are usually associated with non periodic events. The proposed system has been designed so that capacitors are able to power three Active Scan procedures.

After network join - In this mode, Zigbee data transfers entail a relative small impact on V_{main} voltage level when compared to Active Scan/Join procedures. This mode allows data transfer management to be executed with a number of operational options, namely related with communication periodicity, data size and CSMA-CA parameters. The task scheduler decision generates a new event, based on the possibility of a node power loss.

The event scheduler management operation must know beforehand the worst case scenario of a new task generated sink current. System parameters can be defined at hardware level, software compilation time and run time. The scheduler must also account for external parameters. The capacitance value is defined at hardware level and is therefore out of the scheduler control. In this system the mains current behavior is the external parameter.

The compile time parameters are:

Worst case CSMA/CA - The parameters aMinBE, aMaxBE, aMaxBO (defined in Chapter 3) as well as aMaxFR (described in sequel), represent the worst case communication time with the RF module turned on. These parameters are not changed during runtime. The scheduler must therefore account for their effects in current consumption, always assuming that maximum values are reached.

Number of bytes The number of transfered bytes can be regarded as a compile time parameter. If the node is to maintain Zigbee Cluster Library [212] compatibility it is reasonable to assume that the clusters have fixed length payloads. Data transfers are achieved by Endpoint matching between two nodes. This matching is implemented by defining on both sender and destination nodes a common data structure known as simple descriptor. The contents of this data structure are transfered along with its identification that includes endpoint number and profile identification. The architecture of the Simple Descriptor allows all possible data lengths in bits, thus

enabling all sorts of data transfers. The fact that this structure must be known by both source and destination nodes implies that, for each particular case, the data size parameter is known at compilation time.

Network message delivery timeouts - In a Zigbee network with sleeping end devices, data delivery is dependent on ZED poll rate and the parent defined indirect message relaying timeout. The indirect message timeout parameter should be defined so that messages are not discarded before the ZED defined poll rate. In Z-Stack, the indirect message timeout is a parent compile time defined parameter that the network must be aware.

The communication/poll period is a run time parameter and is therefore under the scheduler full control. Moreover the scheduler must decide if the node is running out of energy and issue a network leave command, announcing its absence to the network.

A model is proposed that addresses all mentioned parameters, allowing the task scheduler to decide if a communication event can be generated. The task scheduler manages the energy balance between node consumption, and harvested and capacitor storage energies. Equation 5.1 describes this required energy balance.

$$E_{Cmain} = -E_{Active} - E_{sleep} + E_{harvesting} \tag{5.1}$$

where E_{Cmain} is the capacitor V_{main} available energy, E_{Active} is the energy absorbed by the node when transmitting, receiving or processing data, E_{sleep} is the energy absorbed by the node when in power down mode and $E_{harvesting}$ is the energy supplied by the SCCT harvester. Sleep mode energy is accounted for as constant. Note that the sleep energy includes capacitor self-discharge current as well as the CC2530 and MSP430 currents in low power mode. The total value for the sleep current is considered to be 22.5 μA, where 20 μA is accounted for the Cellergy capacitor [210] and 1 μA for CC2530 [56] Power Mode 2 in which only Sleep Timer is running and 1.5 μA for MSP430 real time clock mode.

The node active power consumption time has variable length and is highly dependent on communication environment. Understanding worst case active mode current consumption is required to determine E_{Active} value. Active time can be described by([194])

$$t_{Active}(n) = t_{swt} + t_{TxRx}(n) + t_{csma} + t_{ack} \tag{5.2}$$

where t_{Active} is the total time the node is transmitting or receiving n bytes, t_{ack} is the acknowledgment procedure time, t_{swt} is the radio module receive/transmit or transmit/receive switching time, $t_{TxRx}(n)$ is the communication time and t_{csma} is the CSMA-CA

algorithm running time. In the case of t_{ack}, transmission is executed without CSMA-CA mechanism thus entailing a fixed time length (see Chapter 3).

The CSMA-CA mechanism is used to avoid communication collision and energy consumption in noisy RF environments. As described in Chapter 3, nodes execute carrier sensing to assess radio channel availability based on a randomly distributed fixed length medium sampling. CSMA-CA is not however fail proof and collisions occur. In this case, when CSMA-CA informs upper layers that the radio channel is clear to send, a communication is attempted. If other node is also communicating or the radio channel is otherwise occupied, data transfer fails and the destination node does not send MAC level acknowledgment. A node that fails to receive data acknowledgment may retry communication. From (3.2) the maximum time a node tries to send data can be written as

$$t_{TxRxmax}(n) = (aMaxFR + 1) \cdot t_{TxRx}(n) \tag{5.3}$$

where $aMaxFR$ is the maximum number of transmit retries a node executes on each communication attempt. The number of transmit retries in both Z-Stack and IEEE 802.15.4 definition is set by default to 3. As communication retries require a new CSMA-CA procedure execution, from (3.1) the maximum time for the CSMA-CA mechanism $t_{csmamax}$ may be written as

$$t_{csmamax} = (aMaxFR + 1) \cdot t_{CSMA} \tag{5.4}$$

The CSMA-CA listening time expressed in (3.1), is dependent on software parameter values $aMinBE$, $aMaxBE$ and $aMaxBO$ that are respectively set by default to 3, 5 and 4 in both IEEE 802.15.4 definition and Z-Stack implementation. The unit backoff period $t_{aUnitBO}$ of (3.1) is 320 μs for 2.4 GHz frequency band. Also in (5.2), t_{swt}, t_{ack} entail constant periods of time with 192 μs and 864 μs, respectively.

The energy balance in (5.1) can be described as a function of the V_{main} capacitor voltage as:

$$V_C(T) - V_C(0) = \frac{1}{C} \cdot \int_0^T \left[-i_{active}(t,n) - i_{sleep}(t) + i_{harvested}(t)\right] dt \tag{5.5}$$

where $V_C(T)$ is the V_{main} capacitor voltage after a charge/discharge period T measured between two consecutive data transfers, $V_C(0)$ is the initial V_{main} capacitor voltage, $i_{active}(t,n)$ is the node instantaneous active current to transfer n bytes at time instant t, i_{sleep} is the node instantaneous current when in sleep mode, and $i_{harvested}$ is the instantaneous current the AC/DC circuit is able to harvest.

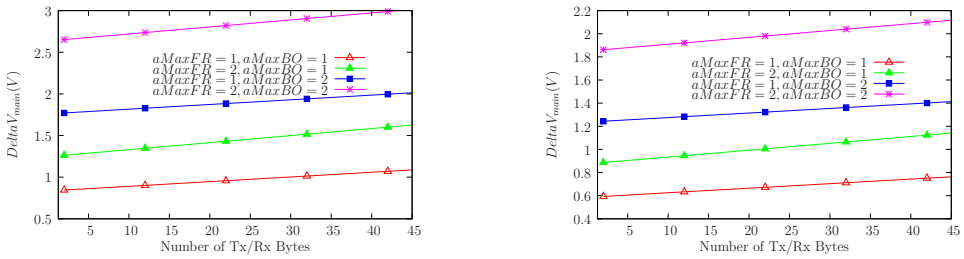
Equation (5.5) can be written as

$$\Delta V_{Cmain} = \frac{I_{Active}(n)}{C} \cdot t_{active} + \frac{I_{Sleep}}{C} \cdot (T - t_{active}) - \frac{I_{harvested}}{C} \cdot T \tag{5.6}$$

where ΔV_{Cmain} is the voltage variation of V_{main} when an I_{Active} current is sinked during t_{active} time within a period T and n bytes are transfered. I_{sleep} is the capacitor's self discharge current plus the CC2530/MSP430 power down currents and $I_{harvested}$ is the harvested current for time interval T, where $T > t_{active}$. Harvested energy is dependent on mains current behavior. The model addresses harvested current with a discrete behavior $I_{harvested} = \{700mA, 800mA, 900mA, 1A, 1.1A, ..., 2A\}$ and $I_{harvested} = \{3A, 4A, ..., 10A\}$. Current I_{active} is the average piecewise value of i_{active} during the time interval of (5.2).

5.3 Model Analysis

The V_{main} voltage level is the main constraint the operational mechanism must address. Fig. 5.3 illustrates V_{main} voltage drop for n bytes data transfers for different values of frame retries and for a number of backoff periods with 470 μF and 330 μF capacitors. These

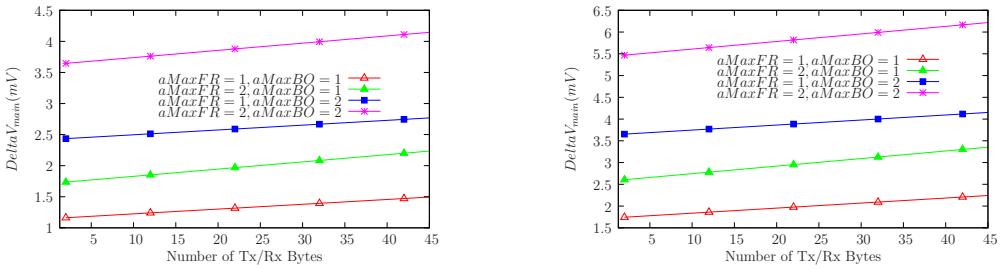


(a) (b)

Figure 5.3: V_{main} voltage drop for n bytes communication as a funtion of frame retries and backoff periods for mains current of 1.2 A. (a) The V_{main} capacitor has 330μF. (b) The V_{main} capacitor has 470μF.

capacitor values have been used in the system proposed in [29]. For the proposed device, the maximum V_{main} supported variation is 1.5V. Fig. 5.3 shows that 330 μF capacitor values can only operate with $aMaxBO$ value of 1 and 10 transfered bytes. A different number of maximum backoff procedures entails a V_{main} variation larger than the supported by the capacitor charge. For a 470 μF, a V_{main} drop outside device operational limits is achieved if $aMaxBO$ and $aMaxFR$ are both 2. For these capacitor values the system

has limited operational capabilities but can be used with the scenario proposed in [31]. In this case 10 bytes are shown to be enough to monitor induction motor parameters. Also the nodes are placed after network formation which is allowed by the contactless architecture. Fig. 5.4 shows system operation with V_{main} Cellergy available capacitor series of 240 mF and 160 mF. The effect of these capacitors is clearly shown by the fact



(a) (b)

Figure 5.4: V_{main} voltage drop for n bytes communication as a function of frame retries and backoff periods for mains current of 1.2 A. (a) The V_{main} capacitor has $240mF$. (b) The V_{main} capacitor has $160mF$.

that V_{main} voltage variation is measured in millivolt. With this hardware configuration the device is able to power the out-of-the-box Z-Stack with payload length illustrated up to 45 bytes. The system is able to sustain the maximum number of frame and backoff retries defined at compile time with the default Z-Stack distribution. Figs. 5.3 and 5.4 illustrate system behavior for the two implemented hardware versions. The analysis for a different hardware configuration (e.g different capacitor values) is illustrated in Fig. 5.5. This analysis shows how changing V_{main} capacitor values allows system communication with different period T. A ΔV_{main} value of zero shows that an energy balance exists between the harvested and the drawn energy. Negative values stand for a surplus of harvested energy as a positive variation illustrates that the node is consuming more energy than the one it scavenges. Moreover, a limit to the harvested energy is defined within the model so that real capacitor storage is accounted for. The limit to the harvested energy is illustrated by the minimum negative ΔV_{main} which means that the capacitor has reached charge limit and thus no more energy can be stored. Fig. 5.5 also illustrates that smaller capacitor values charge faster, by reaching maximum stored energy, and can therefore be better suited to enhance system availability and smaller communications period. Fig. 5.6 shows model behavior for periodic communications. It is shown that

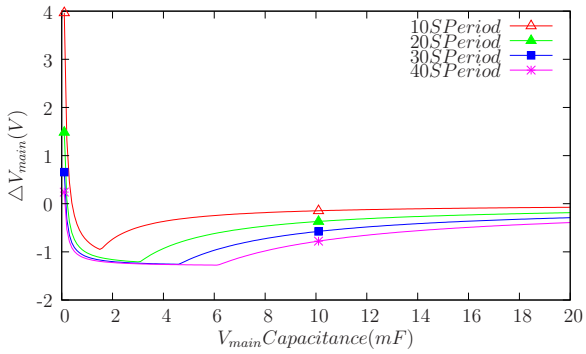
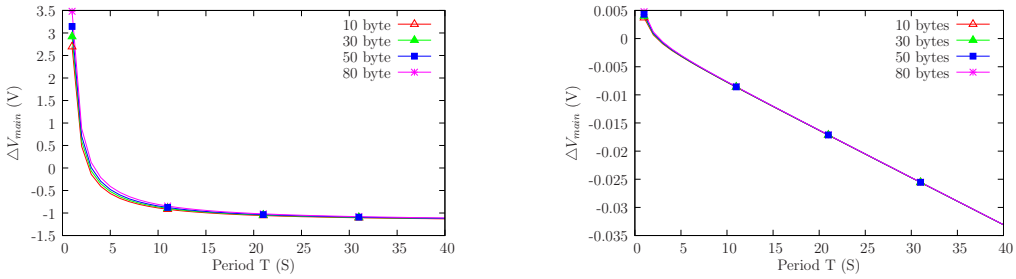


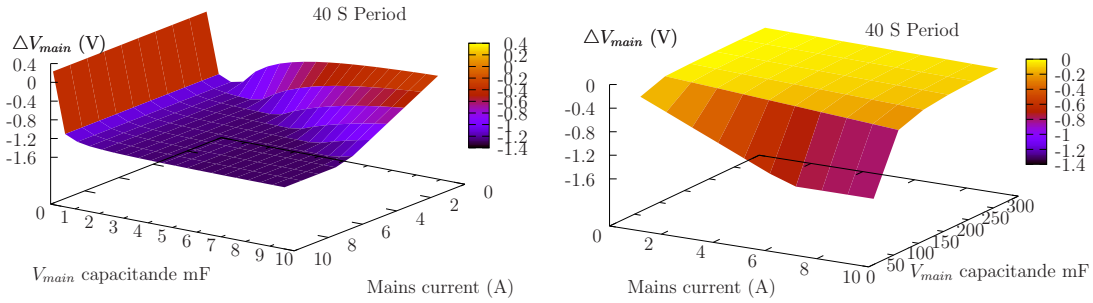
Figure 5.5: V_{main} voltage variation analysis for capacitance value and period T. Mains current is accounted as 1.2 A and MAC parameters as their default values.



(a) V_{main} capacitande $330\mu F$ (b) V_{main} capacitande $240mF$

Figure 5.6: V_{main} voltage variation analysis for communication period (T) and number of transfered bytes. Mains current is accounted as 1.2 A and MAC parameters as their default values.

the number of bytes has little impact on V_{main} variation. Fig. 5.6a is consistent with the measurements presented in Section 4.2 where 5-second spaced communications are compatible with capacitor recharging as for this communication period a negative voltage variation is illustrated. Moreover, Fig. 5.6b illustrates that large capacitors present small V_{main} variations as well as large recharge times represented by the fact that no asymptote exists for these curves. Fig. 5.7 shows the model analysis of the mains current and the V_{main} capacitance for a 40-second communication period. Fig. 5.7a represents model behavior for μF range capacitance values and Fig 5.7b shows results for mF range capacitors.



(a) V_{main} capacitande $330\mu F$ (b) V_{main} capacitande $240mF$

Figure 5.7: V_{main} voltage variation analysis for mains current and V_{main} capacitante. Communication period (T) of 40 S, 40 transfered bytes and MAC parameters default values.

5.4 Model Applications and Mechanism Compliance

The proposed device is to be included in a Zigbee network as a ZED. As all defined ZEDs in Zigbee, it will not execute routing tasks nor data relay. The proposed device must however be able to receive data and implement all protocol functions mandated by the Zigbee definition. ZED operation requires that the device is able to respond to mandatory services. These services are provided by the Zigbee Device Profile and are associated with discover, maintain and configure the network. A client-server mechanism is implemented in which server side services are mandatory. All devices must respond with their short and long addresses and their node and power descriptors when requested, thus exposing device level characteristics. Moreover implemented services (Endpoints) must also be exposed when requested by some node with authority. In this case simple descriptors and active endpoints are transmitted when queried. Table 5.1 shows the Zigbee services with

mandatory response by all network nodes. The number of bytes required by each service is identified in this table as well as the type of service and which node is the responder. For sleeping nodes the parent node can assume the child's role and respond on its behalf. Table 5.1 shows that some services exceed the 10 byte comunication limit determined in

Table 5.1: Implemented device mandatory services with number of bytes, service type and responder in case of non powered devices.

Service	Number of bytes	Type	Responder
Network address	12	Device Discovery	Parent
IEEE address	12	Device Discovery	Node
Node descriptor	17	Device Discovery	Node
Power descriptor	6	Device Discovery	Node
Device anounce	12	Device Discovery	Parent
Simple descriptor	15	Service Discovery	Node
Active end point	8	Service Discovery	Node
Match descriptor	6	Service Discovery	Node

the previous section for the device with the μF range capacitors. The considerations of the previous section have been obtained for the worst case scenario in which the $MacHr$ (see Section 3.5) is computed for a value of 20 bytes. The implemented μF device uses a $MacHr$ of 7 bytes thus allowing it to support the mandatory services payload. The mF range capacitor devices suffer litle influence from the number of transfered bytes.

In a Zigbee network with sleeping end devices, ZED data delivery is dependent on its poll rate and the parent indirect message relaying timeout defined for router devices. The Zigbee Alliance defines Protocol Implementation Conformance Statement (PICS) to specific device profiles. PICS classifies parameters and functionalities for a determined profile. The poll rate and the indirect message timeout are two of these PICS defined parameters. As an example the Zigbee specification recommends polling at least once per 7.5 seconds but the Smart Energy profile allows this poll rate latency to be as large as 60 seconds. In either case the indirect message timeout parameter should be defined so that messages are not discarded before the ZED defined poll rate. The proposed system relies on the mains current availability and on its absence the node energy source is unable to power the node. Moreover, on the router side, message relay is a memory consuming

procedure that can be affected by ZED unavailability. An absent ZED is unable to poll its master and may therefore be discarded and removed from the controller association table.

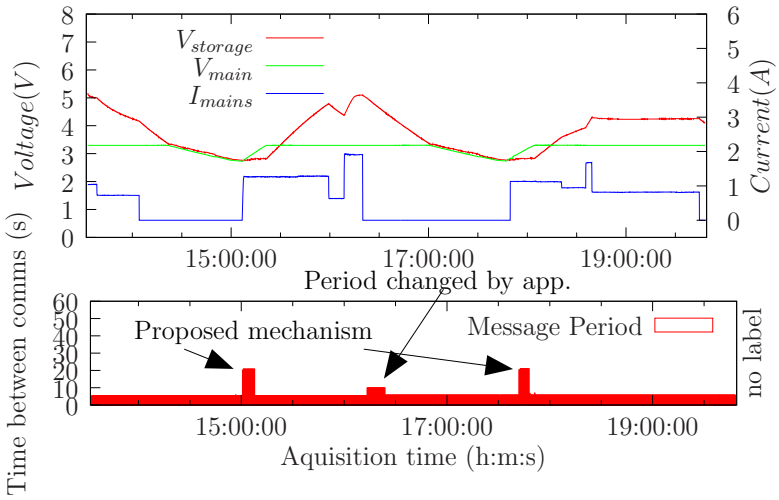


Figure 5.8: V_{main}, $V_{storage}$ voltage, I_{mains} (upper graph) and time between two data transfers (lower graph) illustrating capacitor charging with mains current. The proposed mechanism operation is identified by a change in the data transfer period when V_{main} reaches a low charge value. Also the transmission period changed by the network application software is shown.

A six node network experimental setup has been implemented in which one node is configured as a battery powered ZED, one node is configured as a SCCT powered ZED, three nodes are routers and a central sink node is the coordinator. All nodes implement a time synchronization service in which the coordinator broadcasts its date/time in Coordinated Universal Time (UTC) format, entailing a periodic 4 byte data broadcast. Routers receive the UTC time for themselves and relay it for their children if any. Both ZED transfer to the sink node, in unicast mode, a 16 byte data packet with their V_{main} and $V_{storage}$ values (2 bytes each), synchronized UTC time (4 bytes), short address (4 bytes), 2 bytes with a communication index and 2 bytes with data tags for parsing at the sink. This data packet is sent with adjustable periodicity that may be changed from any other node within the network in unicast mode. The period change command is implemented with a 2 bytes structure. Each node is therefore implemented with three endpoints. Fig. 5.8 illustrates V_{main}, $V_{storage}$ variation with time and different mains currents for the SCCT

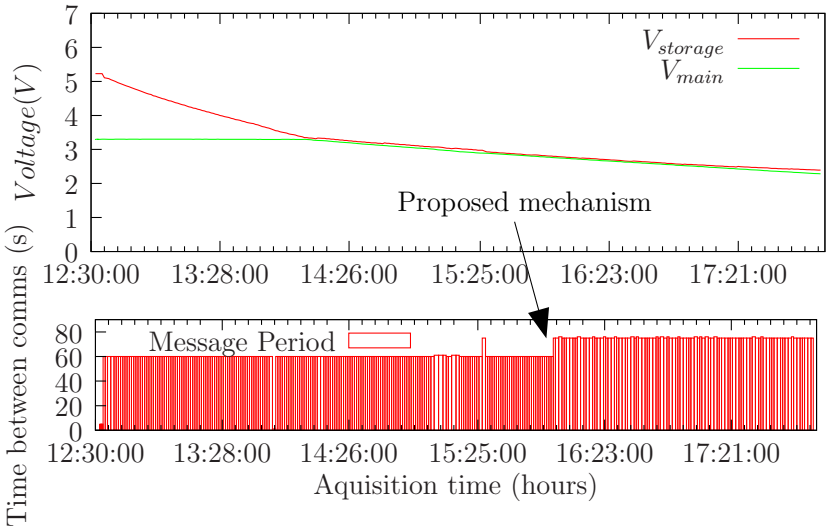


Figure 5.9: Device lifetime for best case communication period of 60s.

powered ZED. The proposed mechanism effect is also shown by changing the data transfer periodicity.

The capacitor is initially fully charged at 13:30:00h. A 5-s data transfer periodicity is thereafter initiated with the V_{main} and $V_{storage}$ readings. Mains current behavior is also shown. In this experimental setup the measurements are obtained by the node resulting that no data is available on a power failure. The proposed mechanism operates therefore by increasing the period of readings thus signaling a low voltage at V_{main}. Initial period communication is restored as V_{main} recharges over the mechanism activation threshold.

In [213] a study is proposed on the existing definitions of network lifetime. Also a generic definition of network lifetime is proposed. A number of lifetime definitions are based on network connectivity metrics. Connectivity is defined as the ability to transmit data to a particular destination. In the proposed system this ability can be defined as the time a node remains operational in the absence of mains current. Also the proposed device assumes the remaining network is fully operational and therefore able to relay data to a particular destination. The node availability is therefore discussed as the capability to poll data within the limits tolerated by the Zigbee definition.

Figs. 5.9 and 5.10 show measured best and worst case for device lifetime with the proposed mechanism. The measurements have been obtained with no mains current and communication period of 60s as the best case scenario. The worst case scenario entails a 1s period communications. For the 60s period, shown in Fig. 5.9, the device is able to

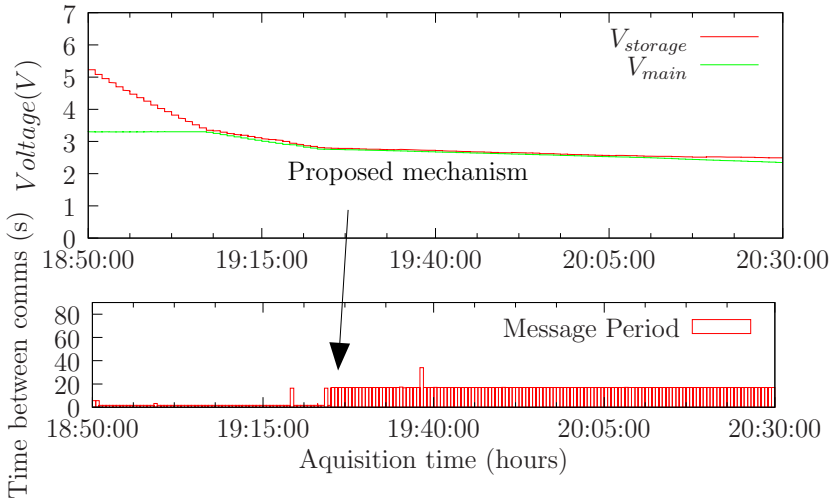


Figure 5.10: Device lifetime for worst case communication period of 1s.

comply with the Zigbee protocol for 04:30 hours without mains current energy supply. After this period the proposed mechanism is activated and two more operational hours are possible with communication periodicity of 70s.

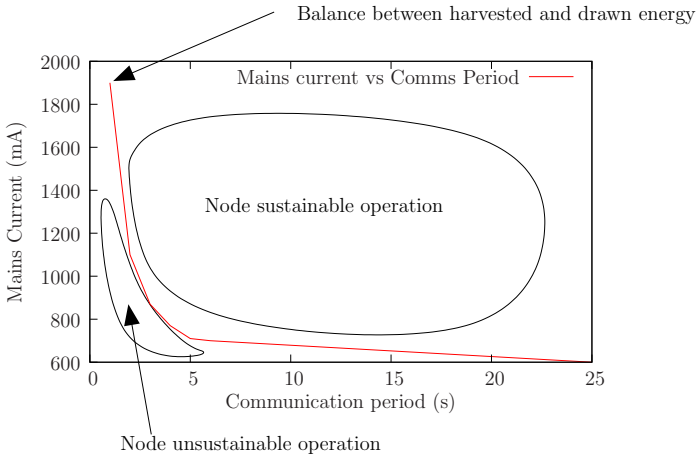


Figure 5.11: Communications period balance with mains current delimiting sustainable device operation.

In the worst case scenario, shown in Fig. 5.10, the device is powered for 01:45 hours and for one more hour with the activated mechanism. Fig 5.11 illustrates mains current values that balance node energy consumption with periodic communications. This figure shows a measure for device sustainability. Large mains currents allow the device to operate with

higher communication frequency. For small mains current lower communication frequency is required.

Chapter 6

Mains Current Estimation Using Nodes Powered on Harvested Energy

The previous chapter addressed the issue of node sustainability under several operational conditions. The focus of the previous analysis is placed on the Zigbee data transfers and the sustainability of the system under regular communications. In the context of a building electrical installation, powering WSN nodes through the exclusive use of harvested energy is an important development. This development is relevant if a number of nodes are able to estimate the Root Mean Square (RMS) value of the current flowing in the power cables they scavenge energy from. As illustrated in Fig.6.1, a WSN with these characteristics may be used to monitor user behavior, equipment efficiency, fault identification and contribute to the implementation of the Smart Grid concept.

In this chapter, a current estimation device powered by a contact-less electromagnetic source is presented. The device main contribution, is the ability to assess mains RMS current value with a self-powered device that runs a complex WSN protocol. Energy is scavenged from the mains current and the device runs exclusively on this harvested and capacitor stored energy. Moreover, the power cable current estimation mechanism is presented and discussed. A single SCCT device scavenges energy from loads electrical grid cables to power the Zigbee node and is used to estimate the current that runs in these electrical wires. This chapter, analyses the impact on capacitor charging of both regular and non regular node protocol operations. This impact is relevant for the mains current

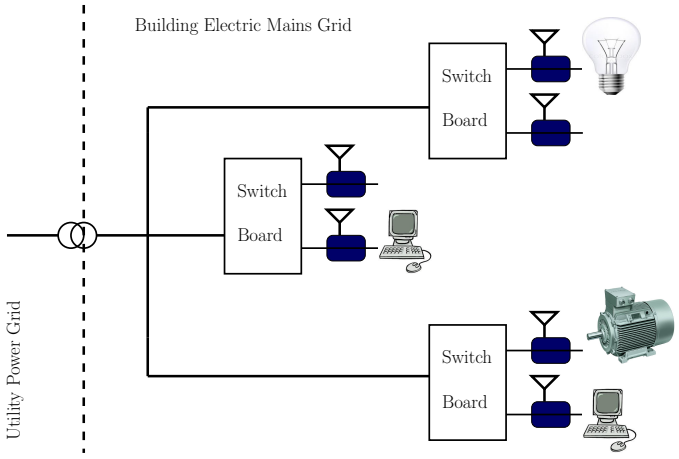


Figure 6.1: Disaggregated mains current monitoring within a building electric installation using Wireless Sensor Networks with nodes that are able to estimate Root Mean Square current.

estimation mechanism. Within the context of this energy neutral device, different operation profiles entail different charge profiles for storage capacitors. This impact is evaluated using the simulation model described in Chapter 4. Model analysis is thereafter used to characterize device relevant behavioral parameters for the mains current estimation.

6.1 Systems Sustainability for Energy Harvesting Operation

The system architecture is described in Chapter 4 and is illustrated by Fig. 4.2. The system operation is characterized by two modes: An active mode where controlled capacitor discharge is executed and both microcontrollers are active; A low power mode where both processors sleep, only the MSP430 real time clock is continuously operating and ADC periodic reads are executed. In this case, the system periodically executes Zigbee data transfers after acquiring $V_{storage}$ and V_{main} levels. After $V_{storage}$ full charge, the system is able to estimate mains current average value by promoting capacitor discharge and measuring the slope of the recharging voltage.

Fig. 6.2, illustrates the described mechanism where a V_{main} and $V_{storage}$ controlled discharge is executed. In this scenario, the $V_{storage}$ capacitor energy is used for immediate V_{main} recharge and $V_{storage}$ starts recharging from the harvested energy. Fig. 6.2, shows

both measured and simulated $V_{storage}$ variation for 1.2-A mains current. $V_{mainSim}$ and $V_{storageSim}$ show simulated results based on the model presented in Chapter 4. $V_{storageMea}$ illustrates measured data.

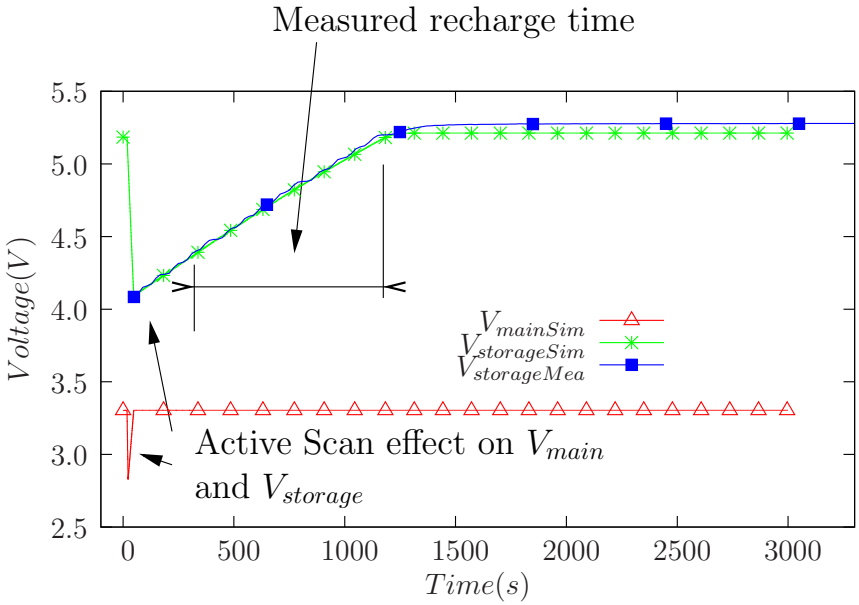


Figure 6.2: $V_{storage}$ and V_{main} voltage levels for Active Scan, association and data communication for ZED over one hour. $V_{storageSim}$ and $V_{mainSim}$ are simulated values. $V_{storageMea}$ presents measured results.

The estimation process identifies a discrete value for I_{mains} that is close to its average value. The instant $V_{storage}$ slope between two consecutive samples is computed and accumulated by a curve fitting algorithm thus obtaining an average slope. The average value is compared against a look up table to estimate mains current. The proposed system operation meets energy constraints preventing V_{main} from discharging bellow minimum voltage required for microcontroller operation. V_{main} controlled discharge is achieved by executing Active Scan procedures. The current drawn by this procedure is software controlled and may therefore be used without capacitor over-discharging. In this system, CC2530 minimum operation voltage is 1.8 V and the MSP430 operates with line voltages as low as 2.0 V.

A test is proposed to evaluate the impact of the current drawn by the Zigbee Active Scan and Join procedures on V_{main} and $V_{storage}$ capacitors. The regular Zigbee data transfers are also tested thus assessing its impact on the estimation process. Fig. 6.3

describes the proposed test sequence. For this test, after joining the network, the ZED is

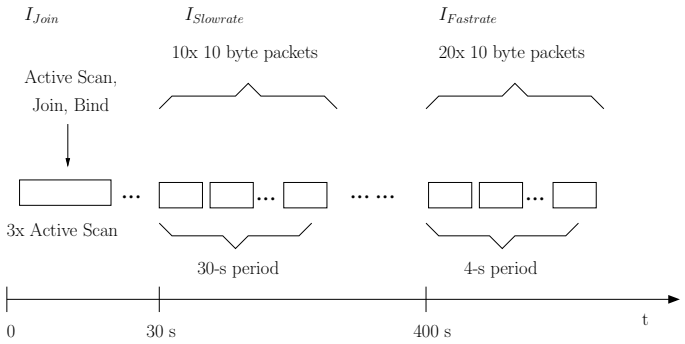


Figure 6.3: Data transfer executed test description.

programmed to send 10 bytes of data with MAC level acknowledgment. These frames are transmitted with a time period of 30 *seconds* ending 300 *seconds* after the first sent frame. The current associated with these frames is named $I_{SlowRate}$. ZED's parent is programmed to start returning data at 400 s. Data are returned with 4-s period, application level acknowledgment and each data frame is sent after a ZED data request command. The current associated with these frames is referenced as $I_{FastRate}$. In this case, last data reception is executed at time 480 s. Test results are shown in figures 6.4, 6.5 and 6.6. Fig. 6.4 shows a 70-s detailed system operation after ZED joining the network to illustrate the impact on both V_{main} and $V_{storage}$ capacitors. The absorbed current by the Active Scan procedure is named I_{Join}.

Fig. 6.4 also shows the first two $I_{SlowRate}$ occurrences, as well as its relative low influence on capacitor voltage levels. Fig. 6.5 details the impact of $I_{SlowRate}$ and $I_{FastRate}$ series of data communications on V_{main} and $V_{storage}$ voltage levels. This detail shows that the system is able to maintain a stable V_{main} voltage level for both communication data. However, for the $I_{FastRate}$ data series, $V_{storage}$ capacitor is required to charge V_{main} thus losing its own charge, demonstrating that $I_{FastRate}$ is no long term sustainable by the device. Fig. 6.6 illustrates the impact of two successive 4-s period $I_{FastRate}$ drawn current on $V_{storage}$ and V_{main}. The impact on V_{main} has an amplitude of 2 mV that is not visible in Fig. 6.5. However, the $V_{storage}$ capacitor is not able to recover and the detail of its descending voltage level is observed.

This test shows, that the proposed system is able to operate without microcontroller shutdown. Data exchanges are executed in a noiseless environment with low probability of collision or occupied channels. The system main constraint is shown to be the Active Scan

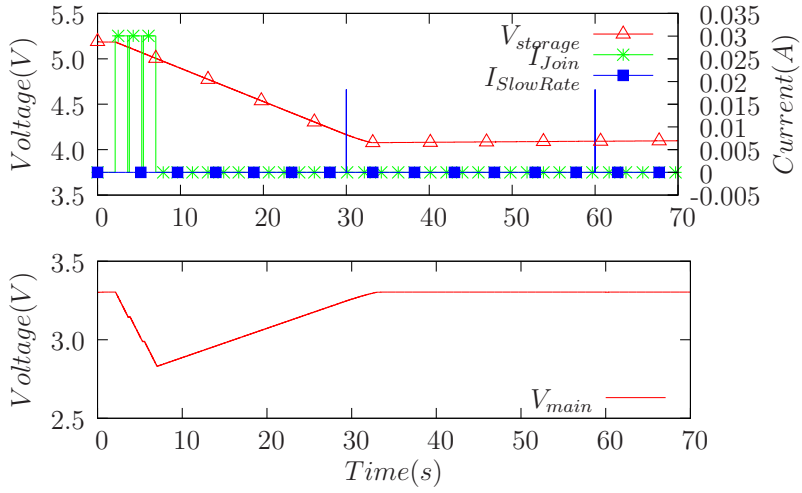


Figure 6.4: Active Scan and association Zigbee device related drawn current and node voltage variation. $V_{storage}$ is the simulated $C_{storage}$ voltage, $I_{SlowRate}$ is the simulated current drawn by a data request and I_{Join} is the simulated drawn current from an Active Scan procedure.

procedure, which requires the use of mili-Farad capacitors, having impact on charge times. Currents I_{Join}, $I_{FastRate}$ and $I_{SlowRate}$ are modeled as square pulses whose duration and amplitude are obtained from the averaging of current by time slot presented in Chapter 3. The variable time slots that are Zigbee operational options dependent are considered for their default value of both IEEE 802.15.4 and Z-Stack. Fig. 6.4 also shows that the Active Scan procedure is an important energy consumer when compared to data transfer actions. In contrast, $I_{SlowRate}$ influence is shown to be small to both V_{main} and $V_{storage}$ charge values. The chosen V_{main} capacitance of 240 mF is thus justified by its ability to sustain out-of-the-box Z-Stack operation of three Active Scan procedures. The same capacitors are serially connected to form a single $C_{storage}$ device of 120 mF due to higher operation voltage.

6.2 Mains Current Estimation Mechanism

The system sequence of operations is controlled by the MSP430 system power manager. Fig. 6.7 illustrates the system work-flow diagram. The MSP430 ADC is used, through the voltage divider illustrated in Fig 4.2, to determine the lowest voltage level reached by $V_{storage}$. The power manager waits for V_{low} level to be reached in $V_{storage}$. V_{low} level

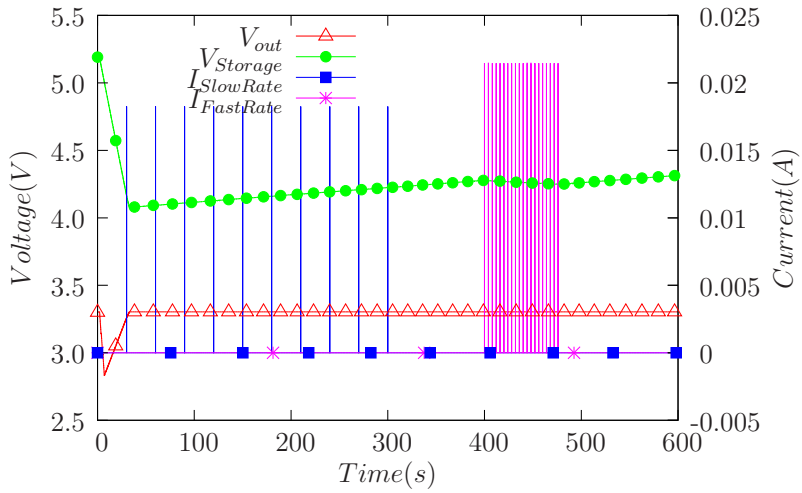


Figure 6.5: Active Scan and association Zigbee device drawn current and node voltage variation. $V_{storage}$, V_{main}, $I_{SlowRate}$ and $I_{FastRate}$ levels for Active Scan, association, bind and data communication 600-seconds detail.

is set to 3.0 V. Once $V_{storage}$ reaches V_{low} the recharge time is started for mains current estimation process. When the recharge process starts, all microcontroller are halted and system placed in low power mode. The system activity is related to ADC periodic $V_{storage}$ and V_{main} measurements to determine charge state. Periodic readings are obtained and transfered to the Zigbee sink node with a one minute interval.

The previous section shows that this 1-minute interval entails a small impact on system behavior while maintaining the Zigbee PICS compliance.

The charge/discharge mechanism operates within fixed voltage boundaries. This voltage bounds are implemented due to the exponential behavior of the capacitor charging process. The charge operation is implemented between 3.0 V and 4.5 V thus obtaining a near linear evolution of the charge $V_{storage}$ voltage. These limits are addressed as useful voltage bounds.

Figs. 6.8 and 6.9 shows V_{main} and $V_{storage}$ full charge time also indicating useful voltage bound. As observed in Figs. 6.8 and 6.9 different mains currents entail diverse $V_{storage}$ voltage rise slopes. The instantaneous $V_{storage}$ rise slope between two consecutive readings may be determined by

$$Vst_{InstSlope} = \frac{V_{storage}(i) - V_{storage}(i-1)}{t(i) - t(i-1)} \tag{6.1}$$

where $Vst_{InstSlope}$ is the calculated instantaneous slope between two consecutive $V_{storage}$

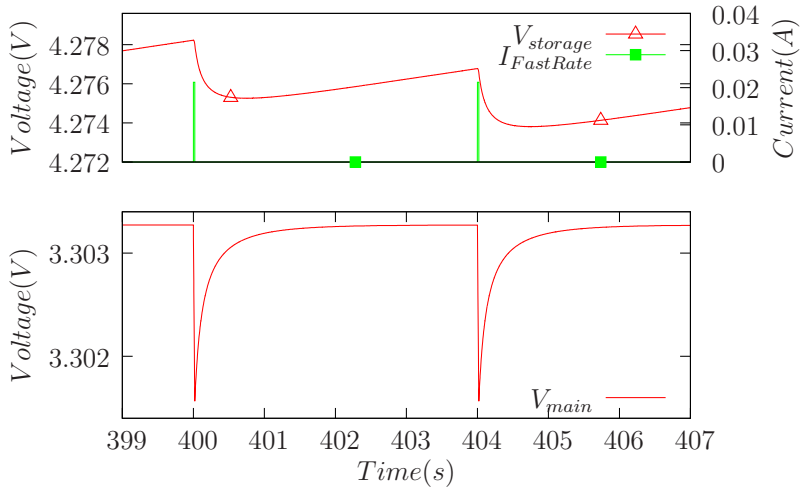


Figure 6.6: $I_{FastRate}$ drawn current on $V_{storage}$ and V_{main} voltage detail impact.

readings, $V_{storage}(i)$ and $V_{storage}(i-1)$ are two consecutive $V_{storage}$ readings, $t(i)$ and $t(i-1)$ are the timestamps of two consecutive readings. Computed instantaneous slopes present however scattered values that prevent its direct use to estimate mains current. A Least Squares Method (LSM) [214], [215], [216] is computed to determine to which curve shown in Figs. 6.8 and 6.9, a calculated slope belongs. For a given set of instantaneous $V_{storage}$ slopes $\{(t_0, Vst_{InstSlope}(0)), ..., (t_N, Vst_{InstSlope}(N))\}$ with N samples the line $y = Mx+B$ that best fits these samples may be given by

$$\begin{pmatrix} \sum_{n=1}^{N} t_n^2 x & \sum_{n=1}^{N} t_n \\ \sum_{n=1}^{N} t_n & \sum_{n=1}^{N} 1 \end{pmatrix} \begin{pmatrix} M \\ B \end{pmatrix} = \begin{pmatrix} \sum_{n=1}^{N} t_n Vst_{InstSlope(n)} \\ \sum_{n=1}^{N} Vst_{InstSlope(n)} \end{pmatrix} \tag{6.2}$$

Fig. 6.10 illustrates the $V_{storage}$ readings, the computed instantaneous slope $Vst_{InstSlope}$ and the slope M. B is the model intersection at line $V_{storage} = 0$ and it is not used in this mechanism. Fig. 6.10 also illustrates a set of defined threshold bands such that,

$$M_{min_I mains} Di < M < M_{max_I mains} Di, D < 1 \tag{6.3}$$

where $M_{min_I mains}$ and $M_{max_I mains}$ are the minimum and maximum M values defined for a given I_{mains}, D is a multiplying factor and i is the i^{th} index of a reading within a discharge process. One threshold band is defined for each discrete estimated mains current. Fig. 6.10 shows that, during one recharge process, the first computed M value for one I_{mains} is within a specific threshold band. Fig. 6.11 shows the same data as Fig. 6.10. The proposed estimation algorithm identifies the current mains average value if

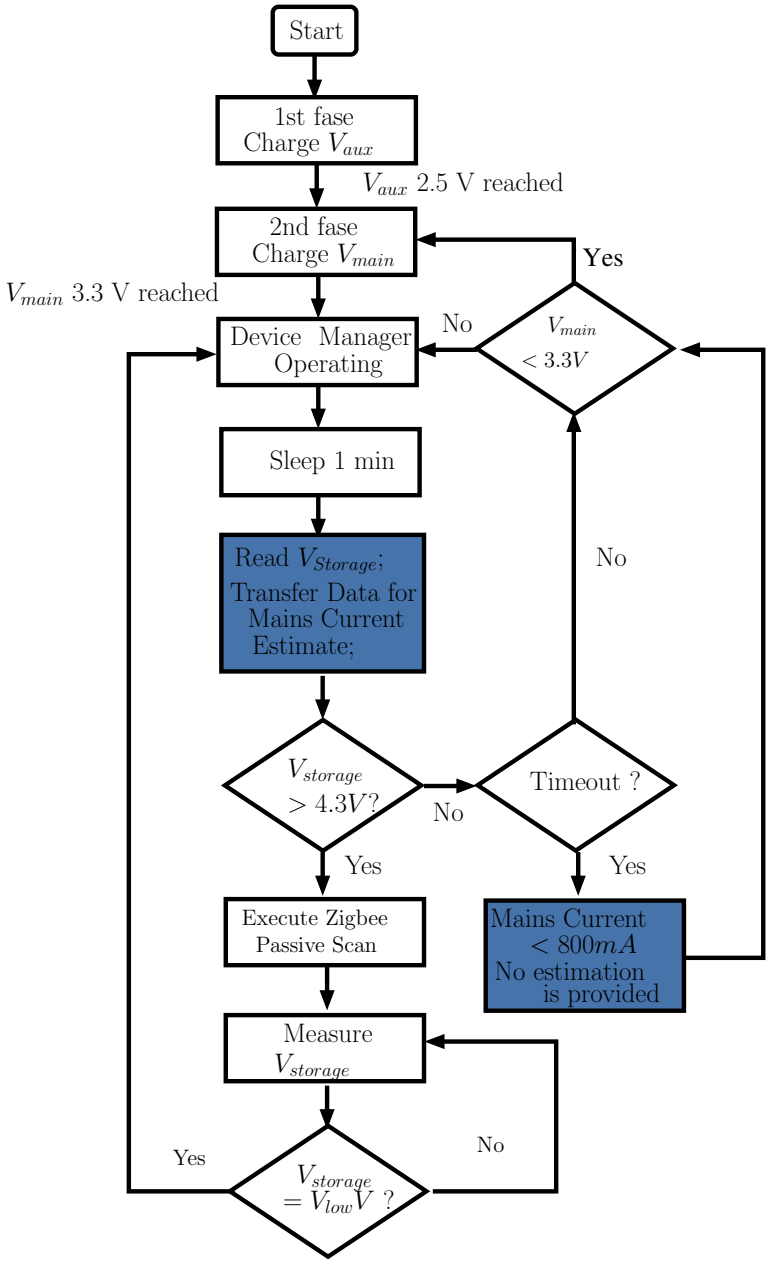


Figure 6.7: System power manager work flow diagram.

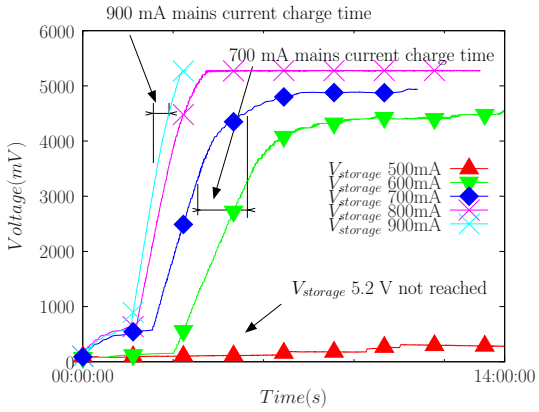


Figure 6.8: Cellergy 120 mF $V_{storage}$ capacitor full charge for mains current from 500 mA to 900 mA.

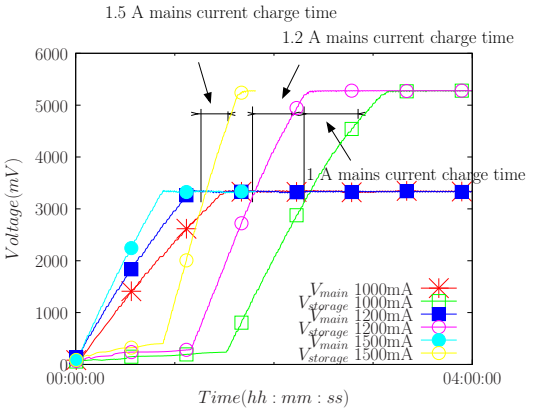


Figure 6.9: Cellergy 120 mF $V_{storage}$ capacitor full charge for mains current from 1 A to 1.5 A.

M is computed within the respective $M_{min_I mains}$ and $M_{max_I mains}$ limits. If I_{mains} remains constant for the subsequent data readings smaller M values are computed. D is therefore introduced so that smaller M values are within the respective threshold band. On I_{mains} increase the computed M value falls between the bounds of a higher threshold band. In this case the estimated mains current is increased. A smaller mains current is estimated if computed M value is smaller than the initial threshold band.

Fig. 6.12 illustrates mains current estimation obtained with this technique. Fig. 6.12 illustrates mains current estimation obtained with this technique. During one recharge process, the first computed M value for one I_{mains} is within its specific threshold band. The

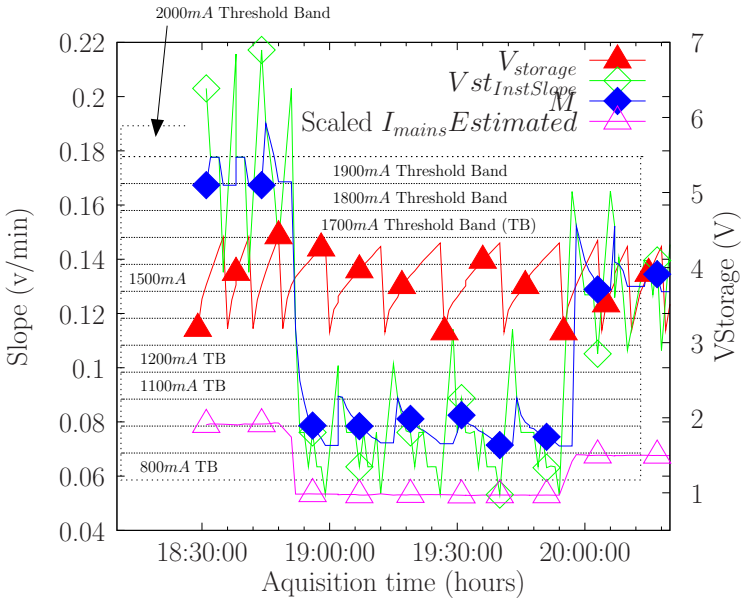


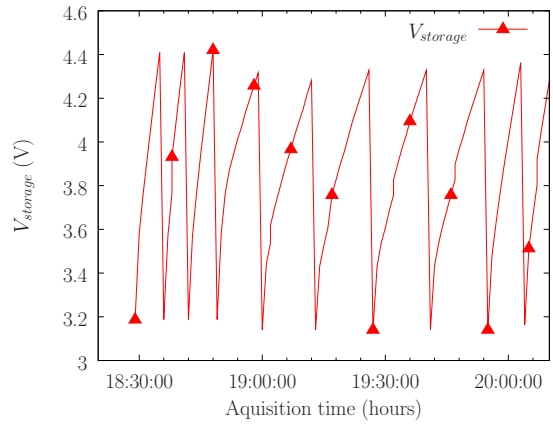
Figure 6.10: $V_{storage}$ readings, computed instantaneous slope $Vst_{InstSlope}$ and the slope M over a 2 h period. I_{mains} is also illustrated.

proposed estimation algorithm identifies the current mains average value if M is computed within the respective M_{min_Imains} and M_{max_Imains} limits. Parameter D is introduced to cope with the fact that if I_{mains} remains constant within a recharge process, smaller M values are computed as time goes by. If I_{mains} increases the computed M value falls within the bounds of a higher threshold band. A smaller mains current is estimated if M falls within the bounds of a lower threshold band. D is set to $D = 0.0015$.

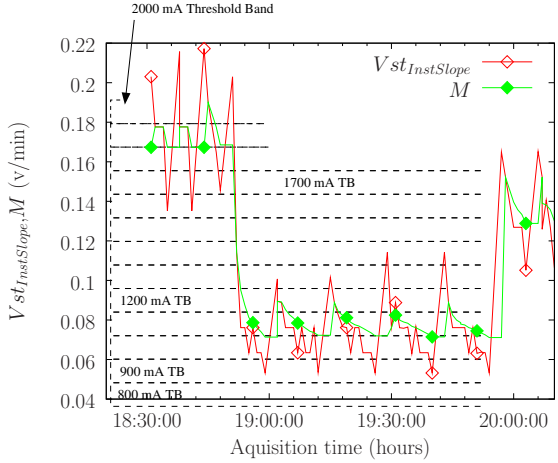
Fig. 6.13 illustrates the estimation error obtained by the proposed device. The spikes that show large errors are introduced by the rapid mains current change that are sensed by the system only after the one minute sleep time. The system behavior is determined by the small sample rate and its large capacitors. The system behaves therefore as a low-pass filter. Fast mains current variations are not directly sensed but accounted as a RMS value.

6.3 Development Tools and Measurement Framework

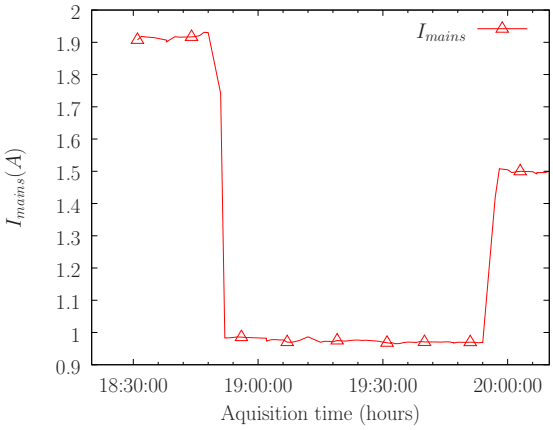
Mains current is measured and compared with the obtained estimate. The measured results have been obtained by the implementation of a digital ammeter. The ammeter is built with the well-known circuit of a SCCT connected to a microcontroller through a



(a) $V_{storage}$ Measurements illustrating variable charge slope.



(b) Instantaneous and Least Mean Square determined slope M.



(c) I_{main} current over the illustrated period.

Figure 6.11: $V_{storage}$ readings, computed instantaneous slope $Vst_{InstSlope}$ and the slope M over a 2 h period. I_{mains} is also illustrated. All values are shown in a separate chart.

Table 6.1: M Threshold bands look up table for the proposed estimation mechanism.

$M_{min_{I}mains}$ V/min		Estimated curret mA		$M_{max_{I}mains}$ V/min
-	<	700 mA	<	0.030
0.030	<	800 mA	<	0.058
0.058	<	900 mA	<	0.069
0.069	<	1000 mA	<	0.080
0.080	<	1100 mA	<	0.091
0.091	<	1200 mA	<	0.102
0.102	<	1300 mA	<	0.113
0.113	<	1400 mA	<	0.124
0.124	<	1500 mA	<	0.134
0.134	<	1600 mA	<	0.144
0.144	<	1700 mA	<	0.154
0.154	<	1800 mA	<	0.164
0.164	<	1900 mA	<	0.174
0.174	<	2000 mA	<	0.184

voltage divider. The voltage divider adds a Direct Current (DC) voltage level to the AC SCCT output voltage. This conditioned signal is connected to the ADC input of a second MSP430FR5739 Board. In this case the current output is converted to a voltage signal through a load resistor. The microcontroller implements an Infinite Impulse Response (IIR) filter to remove the DC component and computes the RMS value of the current as described in [217, 218]. This measurement device transfers data to a Graphical User Interface (GUI) framework through an Universal Serial Bus (USB). Moreover, mains current measurements are corroborated using an hand-held Tektronix DMM249 Multimeter that is used for mains current visual verification as well as to adjust the IIR filter characteristic constant. The implemented framework GUI is shown in Fig. 6.14. The framework is able to process data from the Zigbee network sink node as well as from our ammeter. The framework has been implemented in Visual Studio versions 10 and 12 using the C# programming language. Fig. 6.15 illustrates the implemented test setup showing the Zigbee nodes, the estimating and the measurement devices. The current source, simulating the mains current source, is a Metrel 3.38 KW HSN-0303 Variable Autotransformer connected to a resistive load.

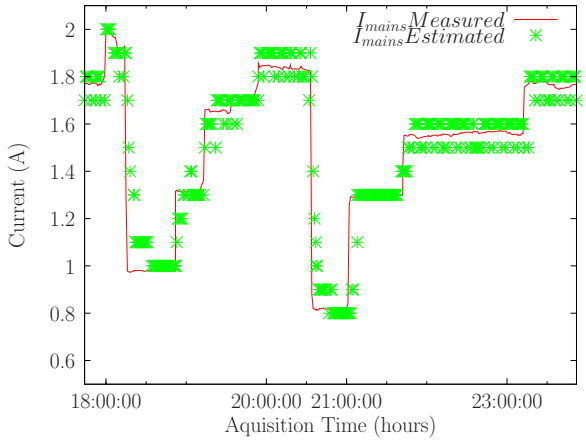


Figure 6.12: System mains current estimations over a 6 h period.

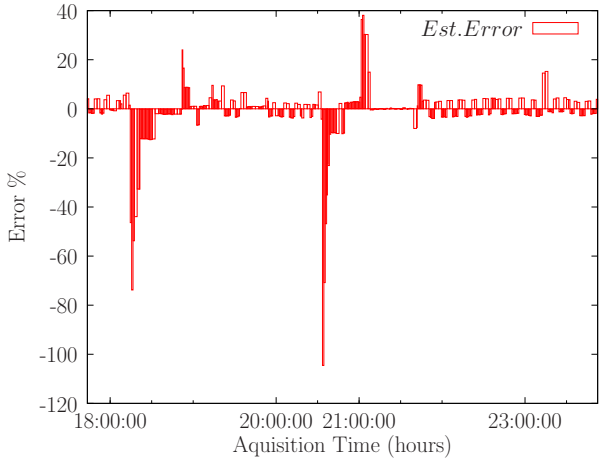


Figure 6.13: System mains current estimations error over a 6 h period.

As described in the previous chapter, a number of bytes are transfered from the proposed ZED to the sink node. The measurements obtained from MSP430FR5739 ADC use a 10 bit data length. The ZED transfers measurement data using the structures identified in Chapter 5. This structure contains both V_{main} and $V_{storage}$ readings as well as the node local time in UTC format. As stated in the previous section, data is transfered to the sync node (and the Windows PC application) with a 1 minute frequency. The power manager implements however a different period reading of the ADC. V_{main} is first read to determine energy stored level. If a low voltage level is measured, the manager doesn't execute the $V_{storage}$ readings and sends a low energy command to the CC2530. If V_{main} is higher than the minimum required energy $V_{storage}$ measurements are executed. Every

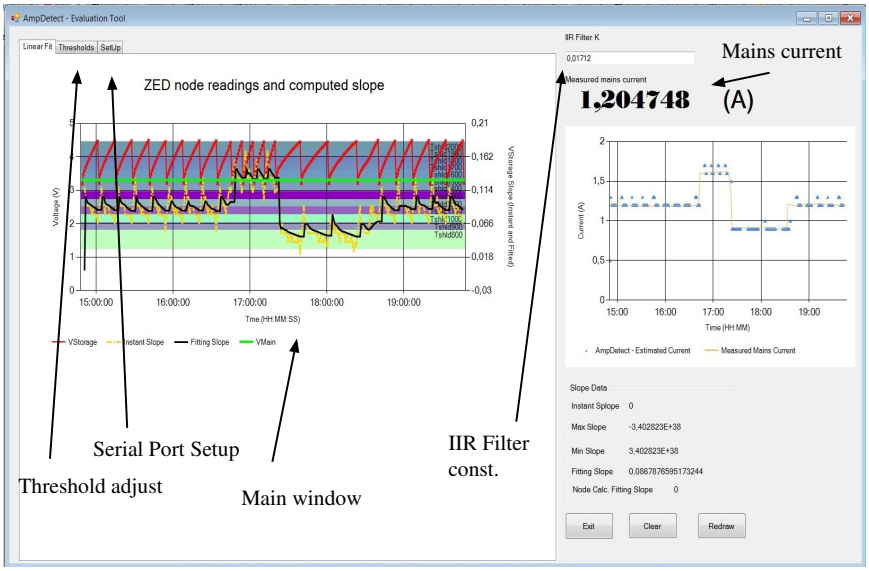


Figure 6.14: The Visual Studio/C# implemented Graphical User Interface (GUI). The GUI collects data from the estimation device Zigbee network and the implemented ammeter.

7.5-s interval the power manager executes 16 ADC consecutive readings executed at the ADC maximum conversion frequency. Each group of 16 samples is averaged and stored. After a 60-s interval the device has 8 samples of $V_{storage}$ that are again averaged and sent to the CC2530 and to the sink node. At each sample the voltage divider illustrated in Fig 4.2 is turned on so that $V_{storage}$ may be read.

Computation within the sync node is done with these data including the transferred timestamps. The measurement activity is a justification for the use of the MSP430 instead of the CC2530. Table 6.2 illustrates a relevant comparison of CC2530 and MSP430FR5739 operational characteristics.

Table 6.2: MSP430FR5739 vs CC2530 relevant characteristic for the measurement operation.

Characteristic	MSP430FR5739	CC2530
ADC conversion time	4 μS	132 μS
ADC sinked current	100 μA	1.2 mA

The slope calculus is implemented on a personal computer running Microsoft Windows 7. The calculus framework has been compiled on a Core i7 and a Centrino 2 microprocessors. The numbers are stored in the ZED as 16 bit unsigned integers. All Zigbee communications are executed with bytes until their arrival at the serial port. A cast to a float is implemented and slope calculus are implemented with this precision. The software for both the MSP430FR5739 and the CC2530 have been developed in C using the IAR Workbench Integrated Development Environment (IDE).

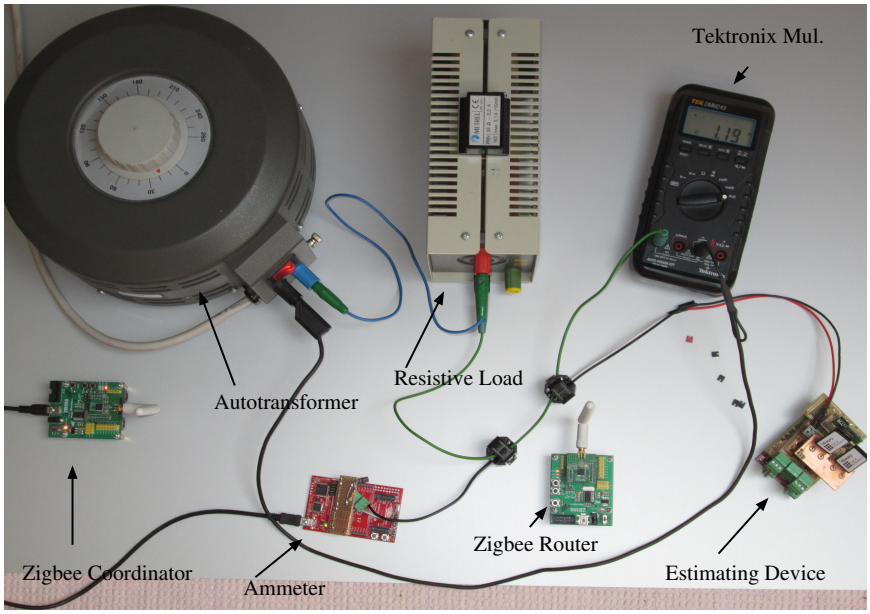


Figure 6.15: The implemented test setup with the Zigbee nodes, the estimating and ammeter devices, the charge cables, the HSN-0303 Autotransformer, the Tektronix multimeter and the resistive load.

Chapter 7

Conclusions

This research work investigates and proposes a number of contributions to improve the operation of large scale wireless sensor networks. The operation of one such system, is dependent on a number of conditions that include easily installable nodes, independence from power sources and the ability to run a complex communication software. The work addresses the main obstacle to the widespread use of wireless sensor networks: Their installation, maintenance and operation costs. A WSN, that requires no skilled workers for node placement and replacement and whose nodes may easily be moved, presents a significant evolution towards low cost and small return of investment time networks. The network operational costs have been addressed by replacing batteries with an energy harvesting solution, without losing operational functionalities. Also, the installation costs are addressed by the proposed solution, that is easy to install in the context of an electrical power facility. Moreover, the proposed solution is able to estimate disaggregated current consumption of electrical loads. This chapter summarizes the work performed during this research dissertation and highlights significant achievements to the existing body of knowledge in wireless sensor networks.

7.1 Summary of Research

The energy source has been identified as the main constraint when operating a WSN. The most obvious energy source for a WSN node is a battery, being these rechargeable devices or non-rechargeable ones (i.e. primary or secondary). However, batteries have a number of drawbacks that seriously limit the implementation of WSN, and are in fact the main constraint to its widespread use. The WSN implementation/operation costs may be

significantly reduced if node battery utilization is substantially reduced or eliminated. Energy scavenging, also known as energy harvesting, is the alternative to the use of batteries. Battery limitations are not exclusively related to node energy depletion. Namely, the need for battery replacement and/or network reconfiguration may entail added cost to WSN operations. A system is proposed that implements a contact-less energy harvesting system for an IEEE 802.15.4/Zigbee wireless module. The device includes a magnetic power generator, a harvesting circuit and a communication interface, which has been designed, tested and implemented. A toroidal coil scavenges energy from power cables connecting electric charges (e.g. electric motors, lighting, computers). The proposed device is a WSN node that runs a complex communication protocol and is exclusively powered on harvested energy. Within the context of a large building electrical installation, an algorithm has been implemented to estimate mains current using only one SCCT, that at the same time is the nodes' single energy source. The estimation is implemented while maintaining the scavenged energy operational constraints as well as using a complex protocol. A SCCT, commonly used for current sensing, implements the source of the micro watt power. The magnetic field generated by alternate current flowing through the power lines is harvested to power the WSN end node. The SCCT that generates a current from the induced electromotive force and a step-up converter circuit with integrated power manager, are able to scavenge energy from this induced current. The harvested energy is stored and accumulated so that a milliwatt device is powered thus allowing WSN nodes operation. The main advantage of this energy harvesting solution is that it can be market ready for industrial or home environments. Built from mature devices, the proposed system doesn't include new materials nor emerging technologies. Such emerging technologies, even though extremely promising, are still in its early steps of development and are therefore untested for full operation conditions. Being built with mature and fully tested components (SCCT, LTC3108), the proposed system is less prone to childhood related problems. Moreover, the proposed device implements a number of energy management strategies, so that a complex wireless communications protocol may be sustained. The presented work demonstrated that the developed system is able to sustain a communication protocol such as IEEE 802.15.4/Zigbee. The developed device is able to power a Zigbee node operating as an End Device that is able to transfer sensor readings while maintain a time synchronization with the network.

Also a SPICE model has been implemented, thus allowing the simulation of the SCCT output voltage. This model is used with the power manager circuit model to determine

the implemented device behavior. The SCCT model has been obtained through an experimental setup while the step-up model is obtained from the manufacturer.

Within the context of WSN operation powered on harvested energy, the protocol management is a fundamental concern. As in the out-of-the-box operation mode, data is identified and related to its origin and a network topology is able to operate. Nodes execute bidirectional communications and follow a protocol to be part of the network. Protocol operations such as registering with the network or joining a group of nodes within the network have been addressed in the context of a battery-free node. Moreover, nodes must be able to listen before transmitting thus requiring more energy than if they were only sending a small number of bytes. The communications protocol requires energy availability that is strongly correlated with its complexity. Within the context of battery-free nodes, IEEE 802.15.4/Zigbee active scan and join activities were found to be critical for node operation. Not only are these activities required for node operation within the network, but they may be required more often than just once in node lifetime/operation. An energy management strategy has been implemented based on the available tools. Texas Instruments IEEE 802.15.4/Zigbee protocol stack is a powerful implementation tool for WSN. It is nevertheless a complex implementation whose operation may be controlled through a large number of compile and run time options. Moreover the IEEE 802.15.4/Zigbee specifications provides a strong support for battery operated nodes but does not consider nodes that run on harvested energy. The proposed device is able to run an adapted Texas Instruments IEEE 802.15.4/Zigbee software stack (Z-Stack). The changes to the original Z-Stack implementation have been presented, so that a full Zigbee compliant node is possible to be run on harvested power. An analysis on the limitations and advantages of the proposed system has been presented.

The developments to the system allowed that the single Split-Core Toroidal Coil Current Transformer (SCCT) device that scavenges energy from load power cables is used to estimate the current that runs in the electrical wires. The proposed harvester device may thus be used for industrial applications and large building monitoring. The original Texas Instruments software stack was successfully adapted to operate on harvested energy as well as to sustain the mains current estimation mechanism. Performed adaptation allowed IEEE 802.15.4/Zigbee protocol compliance to be maintained while system operation is possible for a micro power scavenger.

7.2 Future Work

The proposed system has been presented to the "EDP Inovação 2012" competition. Having reached the final pool of six groups, the jury decision stated that a number of improvements should be addressed before further support from Energias de Portugal (EDP). These suggestion may provide an endorsement for future work. The system should be included in the Zigbee network and the full Smart Energy profile implemented within the harvested power node. Also a more accurate measurement may be achieved by lowering the capacitors value and including a small coin cell battery. The charge/recharge strategy should be adapted to the new power source. Improving the estimation technique may be a future work path as well. In this case, other estimation algorithms may be implemented (e.g. Kalman Filter) to provide better estimation results.

Chapter 8

Bibliography

[1] S. Lucyszyn, "Review of radio frequency microelectromechanical systems technology," *Science, Measurement and Technology, IEE Proceedings* -, vol. 151, pp. 93–103, March 2004.

[2] J. Marek, "Mems for automotive and consumer electronics," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International*, pp. 9–17, Feb 2010.

[3] M. Weiser, "Hot topics: Ubiquitous computing," *IEEE Computer, 38(4)*, October 1993.

[4] M. Weiser, "The computer for the twenty-first century," *Scientific American*, September 1991.

[5] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," in *Computer Networks, 38(4)*, pp. 393–422, Abril 2002.

[6] F.-J. Wu, Y.-F. Kao, and Y.-C. Tseng, "Review: From wireless sensor networks towards cyber physical systems," *Pervasive Mob. Comput.*, vol. 7, pp. 397–413, Aug. 2011.

[7] G. Kortuem, F. Kawsar, D. Fitton, and V. Sundramoorthy, "Smart objects as building blocks for the internet of things," *Internet Computing, IEEE*, vol. 14, pp. 44–51, Jan 2010.

[8] L. Atzori, A. Iera, and G. Morabito, "From "smart objects" to "social objects": The next evolutionary step of the internet of things," *Communications Magazine, IEEE*, vol. 52, pp. 97–105, January 2014.

[9] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury, "A survey on wireless multimedia sensor networks," *Comput. Networks*, vol. 51, no. 4, pp. 921–960, 2007.

[10] M. Dohler, "Wireless sensor networks: The biggest cross-community design exercise to-date," *Bentham Recent Patents on Computer Science, 2008, Vol. 1, No. 1, Opening Issue, to be published.*, vol. 1, pp. 9–25, 2008.

[11] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, "Wireless sensor networks for habitat monitoring," in *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*, WSNA '02, (New York, NY, USA), pp. 88–97, ACM, 2002.

[12] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, and D. Rubenstein, "Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with zebranet," *SIGARCH Comput. Archit. News*, vol. 30, pp. 96–107, Oct. 2002.

[13] G. Werner-Allen, P. Swieskowski, and M. Welsh, "Real-time volcanic earthquake localization," in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems*, SenSys '06, (New York, NY, USA), pp. 357–358, ACM, 2006.

[14] J. Burrell, T. Brooke, and R. Beckwith, "Vineyard computing: sensor networks in agricultural production," *Pervasive Computing, IEEE*, vol. 3, pp. 38–45, Jan 2004.

[15] E. Petriu, N. D. Georganas, D. Petriu, D. Makrakis, and V. Groza, "Sensor-based information appliances," *Instrumentation Measurement Magazine, IEEE*, vol. 3, pp. 31–35, Dec 2000.

[16] S. Stillman and I. Essa, "Towards reliable multimodal sensing in aware environments," in *Proceedings of the 2001 Workshop on Perceptive User Interfaces*, PUI '01, (New York, NY, USA), pp. 1–6, ACM, 2001.

[17] K. Lorinez, D. J. Malan, T. R. F. Fullord-Jones, A. Nawoj, A. Clavel, V. Shayder, G. Maniland, M. Welsh, and S. Moulton, "Sensor networks for emergency response: Challenges and oopportunities.," *IEEE Pervasive Computing*, pp. 16–23, 2004.

[18] M. Wang, L. Ci, P. Zhan, and Y. Xu, "Acoustic source localization in wireless sensor networks," in *IITA '07: Proceedings of the Workshop on Intelligent Information Technology Application*, (Washington, DC, USA), pp. 196–199, IEEE Computer Society, 2007.

[19] G. E. Rolader, J. Rogers, and J. Batteh, "Self-healing minefield," in *Proc. SPIE*, vol. 5441, pp. 13–24, 2004.

[20] Texas Instruments, "AN123 - Breaking the 400-Node ZigBee Network Barrier With TI ZigBee SoC and Z-Stack Software," pp. 1–14, November 2013.

[21] G. Strazdins, A. Elsts, K. Nesenbergs, and L. Selavo, "Wireless sensor network operating system design rules based on real-world deployment survey," *Journal of Sensor and Actuator Networks*, vol. 2, no. 3, pp. 509–556, 2013.

[22] B. Thorstensen, T. Syversen, T.-A. Bjørnvold, and T. Walseth, "Electronic shepherd - a low-cost, low-bandwidth, wireless network system," in *Proceedings of the 2Nd International Conference on Mobile Systems, Applications, and Services*, MobiSys '04, (New York, NY, USA), pp. 245–255, ACM, 2004.

[23] M. Malinowski, M. Moskwa, M. Feldmeier, M. Laibowitz, and J. A. Paradiso, "Cargonet: A low-cost micropower sensor node exploiting quasi-passive wakeup for adaptive asychronous monitoring of exceptional events," in *In Proceedings of the 5th ACM Conference on Embedded Networked Sensor Systems (SenSys â07*, 2007.

[24] K. Chebrolu, B. Raman, N. Mishra, P. K. Valiveti, and R. Kumar, "Brimon: A sensor network system for railway bridge monitoring," in *Proceedings of the 6th International Conference on Mobile Systems, Applications, and Services*, MobiSys '08, (New York, NY, USA), pp. 2–14, ACM, 2008.

[25] J. Amaro, F. Ferreira, R. Cortesão, N. Vinagre, and R. Bras, "Low cost wireless sensor network for in-field operation monitoring of induction motors," in *Industrial Technology (ICIT), 2010 IEEE International Conference on*, pp. 1044–1049, March 2010.

[26] F. Ferreira and A. de Almeida, "Novel multiflux level, three-phase, squirrel-cage induction motor for efficiency and power factor maximization," *Energy Conversion, IEEE Transactions on*, vol. 23, pp. 101–109, March 2008.

[27] J. Amaro, R. Cortesão, J. Landeck, and F. Ferreira, "Harvested power wireless sensor network solution for disaggregated current estimation in large buildings," *Instrumentation and Measurement, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2015.

[28] J. Amaro, R. Cortesão, J. Landeck, and F. Ferreira, "Device and operation mechanism for non-beacon ieee802.15.4/zigbee nodes running on harvested energy," *Ad Hoc Networks*, vol. 26, no. 0, pp. 50 – 68, 2015.

[29] J. Amaro, R. Cortesão, J. Landeck, and F. Ferreira, "Energy harvesting for zigbee compliant wireless sensor network nodes," in *IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society*, 2012.

[30] J. P. Amaro, J. Ferreira, R. Cortesão, and J. Landek, "Powering wireless sensor networks nodes for complex protocols on harvested energy," in *Procedia Technology, Volume 5, 2012,*, Sep 2012.

[31] J. P. Amaro, R. Cortesão, J. Ferreira, and J. Landek, "In-field operation monitoring of induction motors using wireless modules running on harvested power," in *Proceedings 37th Annual Conference of the IEEE Industrial Electronics Society*, 2011, November 2011.

[32] J. P. Amaro, J. Ferreira, R. Cortesão, and J. Landek, "A study on energy harvesting for wireless sensor networks using a split-core current transformer," in *Proceedings of Conference on Electronics Telecommunications and Computers*, Dec 2011.

[33] Zigbee Alliance, "Zigbee - specification protocol." http://www.zigbee.org, 2007.

[34] I. Akyildiz and M. C. Vuran, *Wireless Sensor Networks*. New York, NY, USA: John Wiley & Sons, Inc., 2010.

[35] H. Karl and A. Willig, *Protocols and Architectures for Wireless Sensor Networks*. John Wiley & Sons, 2005.

[36] D. Culler, D. Estrin, and M. Srivastava, "Guest editors introduction: Overview of sensor networks," *IEEE Computer*, vol. 37, pp. 41–49, August 2004.

[37] J. L. Hill, "System architecture for wireless sensor networks," tech. rep., University of Berkeley, California, 2003.

[38] CrossBow Inc, "Mica2 mote." http://www.tinyos.net/scoop/special/hardware#mica2, 2007. Last access in May 2015.

[39] G. Anastasi, A. Falchi, A. Passarella, M. Conti, and E. Gregori, "Performance measurements of motes sensor networks," in *MSWiM '04: Proceedings of the 7th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems*, (New York, NY, USA), pp. 174–181, ACM Press, 2004.

[40] Memsic Inc, "www.memsic.com," Last accessed July 2014.

[41] Atmel, "www.atmel.com," Last accessed July 2014.

[42] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister, "System architecture directions for networked sensors," *ACM SIGPLAN Notices*, vol. 35, pp. 93–104, Nov. 2000.

[43] IEEE Approved Std P802.15.4a/D7, Jan 2007, "Part 15.4: Wireless medium access control (mac) and physical layer (phy) specifications for low-rate wireless personal area networks (wpans)," April 2009.

[44] Sun Laboratories, "Sun(TM) Spot - Theory of Operation - Red Release 5.0." https://wikis.oracle.com/display/sunspotworld/Home, June 2009. Last accessed June 2012.

[45] U. of Karlsruhe, "Teco particle homepage." http://particle.teco.edu/devices/, 2007. Last accessed May 2015.

[46] J. Beutel, "Fast-prototyping using the btnode platform," in *DATE '06: Proceedings of the conference on Design, automation and test in Europe*, (3001 Leuven, Belgium), pp. 977–982, European Design and Automation Association, 2006.

[47] U. Maurer, A. Rowe, A. Smailagic, and D. Siewiorek, "eWatch: A Wearable Sensor and Notification Platform," in *IEEE Body Sensor Networks Workshop*, March 2006.

[48] S. Bagchi, "Nano-kernel: a dynamically reconfigurable kernel for wsn," in *MOBILWARE '08: Proceedings of the 1st international conference on MOBILe Wireless MiddleWARE, Operating Systems, and Applications*, pp. 1–6, 2007.

[49] SCIRO, "SCIRO - CSIRO ICT Centre Queensland Centre for Advanced Technologies (QCAT) Home Page." http://www.sensornets.csiro.au/. Last accessed May 2015.

[50] P. Corke, S. Sen, P. Sikka, P. Valencia, and T. Wark, "2-year progress report," tech. rep., SCIRO - CSIRO ICT Centre Queensland Centre for Advanced Technologies (QCAT), 2006.

[51] A. Schmidt, F. Siegemund, M. Beigl, S. Antifakos, F. Michahelles, and H.-W. Gellersen, "Mobile ad-hoc communication issues in ubiquitous computing – the smart-its experimentation platforms," in *Proc. 8th Intl. Conference on Personal Wireless Communication (PWC 2003), Springer-Verlag, LNCS 2775*, (Venice, Italy), pp. 213–218, Sept. 2003.

[52] M. Beigl and H. Gellersen, "Smart-its: An embedded platform for smart objects," in *In proc. Smart Objects Conference (SOC 2003)*, pp. 15–17, May 2003.

[53] Microchip, "www.microchip.com," Last accessed July 2014.

[54] TinyNode Inc, "Tinynode 584." Accessed April 2015.

[55] Texas Instruments, "Texas instruments," Last accessed July 2014.

[56] Texas Instruments, *CC2530 User Guide and Datasheet*, April 2014. Last accessed May 2015.

[57] Texas Instruments, "CC2538 Device Overview," January 2015. Last accessed May 2015.

[58] Atmel, "Atmel SAM R21E / SAM R21G Datasheet and User Guide," January 2015. Last accessed May 2015.

[59] Atmel, "ATmega128RFA1 Microcontroller with Low Power Transceiver for ZigBee," 2006. Last accessed May 2015.

[60] Freescale, "MC1321x Datasheet and User Guide," March 2009. Last accessed May 2015.

[61] Freescale, "MC1322x Datasheet and User Guide," April 2010. Last accessed may 2015.

[62] Silicon Labs Corp, "EM35x/EM358x/EM359x System-on-Chip (SoC) / Network Co-Processor (NCP)â," 2010. Last accessed May 2015.

[63] ST Microelectronics, "High-performance, IEEE 802.15.4 wireless system-on-chip with up to 256 Kbyte of embedded Flash memory," March 2015. Last accessed May 2015.

[64] Texas Instruments, *Zigbee PRO Network Processor Specification*, January 2010. Application note.

[65] Wikipedia, "List of wireless sensor nodes." http://en.wikipedia.org/wiki/ List_of_-wireless_sensor_nodes#List_of_Wireless_Sensor_Nodes. Last accessed April 2015.

[66] P. Huang, L. Xiao, S. Soltani, M. Mutka, and N. Xi, "The evolution of mac protocols in wireless sensor networks: A survey," *Communications Surveys Tutorials, IEEE*, vol. 15, pp. 101–120, January 2013.

[67] I. Demirkol, C. Ersoy, and F. Alagoz, "Mac protocols for wireless sensor networks: a survey," *Communications Magazine, IEEE*, vol. 44, no. 4, pp. 115–121, 2006.

[68] HART Communication Foudation, "Hart communication protocol specification 7.5," May 2013.

[69] International Society of Automation, "Wireless systems for industrial automation: Process control and related applications - ansi/isa-100.11a," 2011.

[70] S. Petersen and S. Carlsen, "Wirelesshart versus isa100.11a: The format war hits the factory floor," *Industrial Electronics Magazine, IEEE*, vol. 5, no. 4, pp. 23–34, 2011.

[71] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," *The Fifth Symposium on Operating Systems Design and Implementation*, vol. 36, pp. 147–163, 2002.

[72] D. Djenouri and M. Bagaa, "Synchronization protocols and implementation issues in wireless sensor networks: A review," *Systems Journal, IEEE*, vol. PP, no. 99, pp. 1–11, 2014.

[73] I.-K. Rhee, J. Lee, J. Kim, E. Serpedin, and Y.-C. Wu, "Clock synchronization in wireless sensor networks: An overview," *Sensors*, vol. 9, no. 1, pp. 56–85, 2009. http://www.mdpi.com/1424-8220/9/1/56.

[74] L. J. G. Villalba, A. L. S. Orozco, A. T. Cabrera, and C. J. B. Abbas, "Routing protocols in wireless sensor networks," *Sensors*, vol. 9, pp. 8399–8421, 2009.

[75] A. Boukerche, B. Turgut, N. Aydin, M. Z. Ahmad, L. Bölöni, and D. Turgut, "Routing protocols in ad hoc networks: A survey," *Elsevier Computer Networks*, vol. 55, no. 13, pp. 3032 – 3080, 2011.

[76] W. R. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive protocols for information dissemination in wireless sensor networks," in *MobiCom '99: Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, (New York, NY, USA), pp. 174–185, ACM, 1999.

[77] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: a scalable and robust communication paradigm for sensor networks," in *MobiCom '00: Proceedings of the 6th annual international conference on Mobile computing and networking*, pp. 56–67, 2000.

[78] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, "Next century challenges: scalable coordination in sensor networks," in *MobiCom '99: Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, pp. 263–270, 1999.

[79] R. C. Shah and J. M. Rabaey, "Energy aware routing for low energy ad hoc sensor networks," in *IEEE Wireless Communications and Networking Conference (WCNC)*, vol. 1, pp. 350–355 vol.1, March 2002.

[80] D. Braginsky and D. Estrin, "Rumor routing algorthim for sensor networks," in *WSNA '02: Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, (New York, NY, USA), pp. 22–31, ACM, September 2002.

[81] C. Schurgers and M. Srivastava, "Energy efficient routing in wireless sensor networks," in *Military Communications Conference, 2001. MILCOM 2001. Communications for Network-Centric Operations: Creating the Information Force. IEEE*, vol. 1, pp. 357–361 vol.1, October 2001.

[82] M. Chu, H. Haußecker, and F. Zhao, "Scalable information-driven sensor querying and routing for ad hoc heterogeneous sensor networks," *International Journal of High Performance Computing Applications*, vol. 16, pp. 293 – 313, August 2002.

[83] C.-M. Chen, T.-J. Chan, and T.-R. Chen, "An efficient routing protocol with no acknowledgement algorithm in ad-hoc wireless sensor network," in *ICCOM'08: Proceedings of the 12th WSEAS international conference on Communications*, ICCOM'08, (Stevens Point, Wisconsin, USA), pp. 375–379, World Scientific and Engineering Academy and Society (WSEAS), 2008.

[84] Y. Yao and J. Gehrke, "The cougar approach to in-network query processing in sensor networks," *SIGMOD Rec.*, vol. 31, pp. 9–18, September 2002.

[85] N. Sadagopan, B. Krishnamachari, and A. Helmy, "The acquire mechanism for efficient querying in sensor networks," in *In IEEE International Workshop on Sensor Network Protocols and Applications (SNPA'03)*, pp. 149–155, May 2003.

[86] K. Akkaya and M. Younis, "A survey on routing protocols for wireless sensor networks," *Elsevier Ad Hoc Network Journal*, vol. 3, pp. 325–349, May 2005.

[87] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS*, vol. 1, pp. 660–670, October 2002.

[88] S. Lindsey and C. S. Raghavendra, "Pegasis: Power-efficient gathering in sensor information systems," in *Aerospace Conference Proceedings, IEEE*, vol. 3, pp. 3–1125–3–1130 vol.3, March 2002.

[89] A. Manjeshwar and D. P. Agrawal, "Teen: a routing protocol for enhanced efficiency in wireless sensor networks," in *Parallel and Distributed Processing Symposium., Proceedings 15th International*, pp. 2009–2015, April 2001.

[90] A. Manjeshwar and D. P. Agrawal, "Apteen: a hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks," in *Parallel and Distributed Processing Symposium., Proceedings International, IPDPS 2002, Abstracts and CD-ROM*, pp. 195–202, April 2002.

[91] M. Younis, M. Youssef, and K. Arisha, "Energy-aware routing in cluster-based sensor networks," in *MASCOTS '02: Proceedings of the 10th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems*, p. 129, IEEE Computer Society, October 2002.

[92] L. Subramanian and R. H. Katz, "An architecture for building self-configurable systems," in *MobiHoc '00: Proceedings of the 1st ACM international symposium on Mobile ad hoc networking & computing*, pp. 63–73, August 2000.

[93] E. L. Li and J. Y. Halpern, "Minimum-energy mobile wireless networks revisited," in *Communications, 2001. ICC 2001. IEEE International Conference on*, vol. 1, pp. 278–283 vol.1, Jun 2002.

[94] Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed energy conservation for ad hoc routing," in *MobiCom '01: Proceedings of the 7th annual international conference on Mobile computing and networking*, MobiCom '01, (New York, NY, USA), pp. 70–84, ACM, July 2001.

[95] B. Karp and H. T. Kung, "Gpsr: greedy perimeter stateless routing for wireless networks," in *MobiCom '00: Proceedings of the 6th annual international conference on Mobile computing and networking*, pp. 243–254, ACM, 2000.

[96] D. Niculescu and B. Nath, "Trajectory based forwarding and its applications," in *MobiCom '03: Proceedings of the 9th annual international conference on Mobile computing and networking*, pp. 260–272, ACM, 2003.

[97] M. do V. Machado, O. Goussevskaia, R. A. F. Mini, C. G. Rezende, A. A. F. Loureiro, G. R. Mateus, and J. M. Nogueira, "Data dissemination using the energy map," *Wireless on Demand Network Systems and Service, International Conference on*, vol. 0, pp. 139–148, 2005.

[98] Q. Huang, C. Lu, and G.-C. Roman, "Spatiotemporal multicast in sensor networks," in *SenSys '03: Proceedings of the 1st international conference on Embedded networked sensor systems*, (New York, NY, USA), pp. 205–217, ACM, 2003.

[99] Q. Huang, S. Bhattacharya, C. Lu, and G.-C. Roman, "Far: Face-aware routing for mobicast in large-scale sensor networks," *ACM Trans. Sen. Netw.*, vol. 1, no. 2, pp. 240–271, 2005.

[100] A. M. V. Reddy, A. P. Kumar, D. Janakiram, and G. A. Kumar, "Wireless sensor network operating systems; a survey," *International Journal of Sensor Networks*, vol. 5, no. 4, pp. 236–255, 2009.

[101] A. K. Dwivedi, M. K. Tiwari, and O. P. Vyas, "Operating systems for tiny networked sensors: A survey," *International Journal of Recent Trends in Engineering*, vol. 1, May 2009.

[102] J. Hill, M. Horton, R. Kling, and L. Krishnamurthy, "The platforms enabling wireless sensor networks," *Communications ACM*, vol. 47, no. 6, pp. 41–46, 2004.

[103] S. Bhatti, J. Carlson, H. Dai, J. Deng, J. Rose, A. Sheth, B. Shucker, C. Gruenwald, A. Torgerson, and R. Han, "Mantis os: an embedded multithreaded operating system for wireless micro sensor platforms," *Mobile Networks and Applications*, vol. 10, no. 4, pp. 563–579, 2005.

[104] A. Dunkels, B. Grönvall, and T. Voigt, "Contiki - a lightweight and flexible operating system for tiny networked sensors," in *Proceedings of the First IEEE Workshop on Embedded Networked Sensors (Emnets-I)*, Nov. 2004.

[105] Texas Instruments, *Z-Stack OS Abstraction Layer Application Programming Interface*. Texas Instruments, April 2011.

[106] C.-C. Han, R. Kumar, R. Shea, E. Kohler, and M. Srivastava, "A dynamic operating system for sensor nodes," in *MobiSys '05: Proceedings of the 3rd international conference on Mobile systems, applications, and services*, (New York, NY, USA), pp. 163–176, ACM Press, 2005.

[107] R. Barr, J. C. Bicket, D. S. Dantas, B. Du, T. W. D. Kim, B. Zhou, and E. G. Sirer, "On the need for system-level support for ad hoc and sensor networks," *ACM SIGOPS Operating Systems Review*, vol. 36, no. 2, pp. 1–5, 2002.

[108] M. Korkalainen, M. Sallinen, N. Karkkainen, and P. Tukeva, "Survey of wireless sensor networks simulation tools for demanding applications," *Networking and Services, International conference on*, vol. 0, pp. 102–106, 2009.

[109] M. Greis, "Tutorial for the network simulator ns." http://isi.edu/nsnam/ns/tutorial/. Last accessed in April 2015.

[110] L. F. Perrone, B. Ward, and A. Hallagan, "An experiment automation framework for ns-3," in *SIMUTools 2010 Conference, Poster Session*, March 2010.

[111] NS3 Community, "What is NS3?." https://www.nsnam.org/overview/what-is-ns-3/, Jan 2011. Last accessed May 2015.

[112] OmNetpp, "OMNeT++ - C++ simulation library and framework." http://omnetpp.org/. Last accessed in April 2015.

[113] Opnet, ""modeler network simulator"." http://www.riverbed.com/products/performance-management-control/opnet.html?redirect=opnet, Last accessed May 2015.

[114] Institute for Software Integrated Systems, "Prowler: Probabilistic wireless network simulator." http://www.isis.vanderbilt.edu/Projects/nest/prowler. Last accessed May 2015.

[115] R. Vullers, R. van Schaijk, I. Doms, C. V. Hoof, and R. Mertens, "Micropower energy harvesting," *Elsevier Solid-State Electronics*, vol. 53, pp. 684–693, July 2009.

[116] C. Mammano, "Portable power - a designer's guide to battery manegement," technical report, Unitrode Corporation, 2000.

[117] C. Knight, J. Davidson, and S. Behrens, "Energy options for wireless sensor nodes," *MDPI Sensors*, vol. 8, pp. 8037–8066, December 2008.

[118] R. Rajagopal, D. M, and M. M, "Article: Zoological tracking system with portable power using sensors and fuel cell system," *International Journal of Applied Information Systems*, vol. 4, pp. 1–6, December 2012. Published by Foundation of Computer Science, New York, USA.

[119] F. Chraim and S. Karaki, "Fuel cell applications in wireless sensor networks," in *Instrumentation and Measurement Technology Conference (I2MTC), 2010 IEEE*, pp. 1320–1325, May 2010.

[120] S. Chalasani and J. M. Conrad, "A survey of energy harvesting sources for embedded systems," in *Southeastcon. IEEE*, pp. 442–447, April 2008.

[121] S. Sudevalayam and P. Kulkarni, "Energy harvesting sensor nodes: Survey and implications," *Communications Surveys Tutorials, IEEE*, vol. 13, pp. 443–461, March 2011.

[122] G. Tewolde, "Current trends in low-power embedded computing," in *Electro/Information Technology (EIT), 2010 IEEE International Conference on*, pp. 1–6, May 2010.

[123] S. Beeby and N. White, *Energy Harvesting for Autonomous Systems.* Artech House, 2010.

[124] A. Harb, "Energy harvesting: State-of-the-art," *Elsevier Renewable Energy Journal*, vol. Vol. 36, pp. 2641–2654, October 2011.

[125] R. Freeland, "Energy harvesting: A practical reality for wireless sensing." http://www.perpetuum.com/resources.asp, January 2012. Last accessed May 2015.

[126] A. Kansal, J. Hsu, S. Zahedi, and M. B. Srivastava, "Power management in energy harvesting sensor networks," *ACM Trans. Embed. Comput. Syst.*, vol. 6, Sept. 2007.

[127] M. Kuorilehto, M. Kohvakka, J. Suhonen, P. Hamalainen, M. Hannikainen, and T. D. Hamalainen, *Ultra-Low Energy Wireless Sensor Networks in Practice: Theory, Realization and Deployment.* Wiley Publishing, 2008.

[128] W. Seah, Z. A. Eu, and H.-P. Tan, "Wireless sensor networks powered by ambient energy harvesting (wsn-heap) - survey and challenges," in *Wireless Communication, Vehicular Technology, Information Theory and Aerospace Electronic Systems Technology, 2009. Wireless VITAE 2009. 1st International Conference on*, pp. 1–5, May 2009.

[129] C. Carvalho, J. Oliveira, and N. Paulino, "Survey and analysis of the design issues of a low cost micro power dc-dc step up converter for indoor light energy harvesting applications," in *Mixed Design of Integrated Circuits and Systems (MIXDES), 2012 Proceedings of the 19th International Conference*, pp. 455–460, May 2012.

[130] A. Weddell, G. V. Merrett, and B. Al-Hashimi, "Photovoltaic sample-and-hold circuit enabling mppt indoors for low-power systems," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 59, pp. 1196–1204, June 2012.

[131] A. Decker, "Solar energy harvesting for autonomous field devices," *Wireless Sensor Systems, IET*, vol. 4, pp. 1–8, March 2014.

[132] S. Ghosh, H.-T. Wang, and W. Leon-Salas, "A circuit for energy harvesting using on-chip solar cells," *Power Electronics, IEEE Transactions on*, vol. 29, pp. 4658–4671, Sept 2014.

[133] W. Chye, Z. Dahari, O. Sidek, and M. Miskam, "Electromagnetic micro power generator; a comprehensive survey," in *Industrial Electronics Applications (ISIEA), 2010 IEEE Symposium on*, pp. 376–382, oct. 2010.

[134] K. Tao, G. Ding, P. Wang, Z. Yang, and Y. Wang, "Fully integrated micro electromagnetic vibration energy harvesters with micro-patterning of bonded magnets," in *Micro Electro Mechanical Systems (MEMS), 2012 IEEE 25th International Conference on*, pp. 1237–1240, Jan 2012.

[135] K. Ono, N. Sato, T. Shimamura, M. Ugajin, T. Sakata, S. Mutoh, and Y. Sato, "A millimeter-sized electret-energy-harvester with microfabricated horizontal arrays and vertical protrusions for power generation enhancement," in *Solid-State Sensors, Actuators and Microsystems Conference (TRANSDUCERS), 2011 16th International*, pp. 1863–1866, June 2011.

[136] R. Caliò, U. Rongala, D. Camboni, M. Milazzo, C. Stefanini, G. de Petris, and C. Oddo, "Piezoelectric Energy Harvesting Solutions," *MDPI Sensors*, vol. 14, pp. 4755–4790, March 2014.

[137] P. D. Mitcheson and E. M. Yeatman, "Energy harvesting for pervasive computing." http://www.perada-magazine.eu/pdf/1306/1306.pdf, October 2008.

[138] X. Lu and S.-H. Yang, "Thermal energy harvesting for wsns," in *Systems Man and Cybernetics (SMC), 2010 IEEE International Conference on*, pp. 3045–3052, October. 2010.

[139] V. Leonov, "Thermoelectric energy harvesting of human body heat for wearable sensors," *Sensors Journal, IEEE*, vol. 13, pp. 2284–2291, June 2013.

[140] A. Post, C. Knight, and E. Kisi, "Thermomagnetic energy harvesting with first order phase change materials," *Journal of Applied Physics*, vol. 114, pp. 033915–033915–7, Jul 2013.

[141] M. Kiziroglou, S. Wright, T. Toh, P. Mitcheson, T. Becker, and E. Yeatman, "Design and fabrication of heat storage thermoelectric harvesting devices," *Industrial Electronics, IEEE Transactions on*, vol. 61, pp. 302–309, Jan 2014.

[142] R. Vullers, R. Schaijk, H. Visser, J. Penders, and C. Hoof, "Energy harvesting for autonomous wireless sensor networks," *Solid-State Circuits Magazine, IEEE*, vol. 2, pp. 29–38, spring 2010.

[143] Y. Ding, T. Arslan, and A. Hamilton, "Broadband antenna for rf energy scavenging system," in *Antennas and Propagation Conference (LAPC), 2012 Loughborough*, pp. 1–4, Nov 2012.

[144] S.-E. Adami, V. Marian, N. Degrenne, C. Vollaire, B. Allard, and F. Costa, "Self-powered ultra-low power dc-dc converter for rf energy harvesting," in *Faible Tension Faible Consommation (FTFC), 2012 IEEE*, pp. 1–4, June 2012.

[145] R. Shigeta, T. Sasaki, D. M. Quan, Y. Kawahara, R. Vyas, M. Tentzeris, and T. Asami, "Ambient rf energy harvesting sensor device with capacitor-leakage-aware duty cycle control," *Sensors Journal, IEEE*, vol. 13, pp. 2973–2983, Aug 2013.

[146] B. Ulmen, P. D. Desai, S. Moghaddam, G. H. Miley, and R. I. Masel, "Development of diode junction nuclear battery using ^{63}ni," *Journal of Radioanalytical and Nuclear Chemistry*, vol. 282, pp. 601–604, November 2009.

[147] D. yong Qiao, X.-J. Chen, Y. Ren, and W.-Z. Yuan, "A micro nuclear battery based on sic schottky barrier diode," *Microelectromechanical Systems, Journal of*, vol. 20, pp. 685–690, June 2011.

[148] H. Guo, Y. Shi, Y. Zhang, Y. Zhang, and J. Han, "Fabrication of sic p-i-n betavoltaic cell with 63ni irradiation source," in *Electron Devices and Solid-State Circuits (EDSSC), 2011 International Conference of*, pp. 1–2, Nov 2011.

[149] A. Lal, R. Duggirala, and H. Li, "Pervasive power: A radioisotope-powered piezoelectric generator," *IEEE Pervasive Computing*, vol. 4, no. 1, pp. 53–61, 2005.

[150] N. Roscoe and M. Judd, "Optimization of voltage doublers for energy harvesting applications," *Sensors Journal, IEEE*, vol. 13, pp. 4904–4911, Dec 2013.

[151] Linear Technology, "Energy harvesting device parametric search." www.linear.com/parametric/Energy_Harvesting. Last accessed May 2015.

[152] Maxim Integrated Circuits, *MAX17710 Energy-Harvesting Charger and Protector*. Maxim IC, http://www.maximintegrated.com/ en/products/power/battery-management/MAX17710.html, 2012.

[153] Texas Instruments, *BQ25504 Ultra Low-Power Boost Converter With Battery Management for Energy*, 2014.

[154] M. Zhu, P. Baker, N. M. Roscoe, M. Judd, and J. Fitch, "Alternative power sources for autonomous sensors in high voltage plant," in *2009 IEEE Electrical Insulation Conference*, June 2009.

[155] T. Taithongchai and E. Leelarasmee, "Adaptive electromagnetic energy harvesting circuit for wireless sensor application," in *Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, 2009. ECTI-CON 2009. 6th International Conference on*, vol. 01, pp. 278–281, May 2009.

[156] A. Kosonen and J. Ahola, "Communication concept for sensors at an inverter-fed electric motor utilizing power-line communication and energy harvesting," *Power Delivery, IEEE Transactions on*, vol. 25, pp. 2406–2413, October 2010.

[157] M. A. Rashed H. Bhuiyan, Roger A. Dougal, "A miniature energy harvesting device for wireless sensors in electric power system," *IEEE Sensors Journal*, vol. 10, pp. 1249–1258, July 2010.

[158] D. Gunduz, K. Stamatiou, N. Michelusi, and M. Zorzi, "Designing intelligent energy harvesting communication systems," *Communications Magazine, IEEE*, vol. 52, pp. 210–216, January 2014.

[159] N. Michelusi, K. Stamatiou, L. Badia, and M. Zorzi, "Operation policies for energy harvesting devices with imperfect state-of-charge knowledge," in *Communications (ICC), 2012 IEEE International Conference on*, pp. 5782–5787, June 2012.

[160] R. Shigeta, T. Sasaki, D. Quan, Y. Kawahara, R. Vyas, M. Tentzeris, and T. Asami, "Ambient RF energy harvesting sensor device with capacitor-leakage-aware duty cycle control," *Sensors Journal, IEEE*, vol. 13, pp. 2973–2983, August 2013.

[161] R. Moghe, A. Iyer, F. Lambert, and D. Divan, "A low-cost electric-field energy harvester for an mv/hv asset-monitoring smart-sensor," in *Energy Conversion Congress and Exposition (ECCE), 2013 IEEE*, pp. 2676–2683, Sept 2013.

[162] N. Roscoe and M. Judd, "Harvesting energy from magnetic fields to power condition monitoring sensors," *Sensors Journal, IEEE*, vol. 13, pp. 2263–2270, June 2013.

[163] Z. Wu, Y. Wen, and P. Li, "A power supply of self-powered online monitoring systems for power cords," *Energy Conversion, IEEE Transactions on*, vol. 28, pp. 921–928, Dec 2013.

[164] F. Guo, H. Hayat, and J. Wang, "Energy harvesting devices for high voltage transmission line monitoring," in *Power and Energy Society General Meeting, 2011 IEEE*, pp. 1–8, July 2011.

[165] J. Froehlich, E. Larson, S. Gupta, G. Cohn, M. Reynolds, and S. Patel, "Disaggregated end-use energy sensing for the smart grid," *Pervasive Computing, IEEE*, vol. 10, pp. 28–39, Jan 2011.

[166] J. Gutierrez, J. Villa-Medina, A. Nieto-Garibay, and M. Porta-Gandara, "Automated irrigation system using a wireless sensor network and gprs module," *Instrumentation and Measurement, IEEE Transactions on*, vol. 63, pp. 166–176, Jan 2014.

[167] R. Yan, H. Sun, and Y. Qian, "Energy-aware sensor node design with its application in wireless sensor networks," *Instrumentation and Measurement, IEEE Transactions on*, vol. 62, pp. 1183–1191, May 2013.

[168] Y. Kim, R. Evans, and W. Iversen, "Remote sensing and control of an irrigation system using a distributed wireless sensor network," *Instrumentation and Measurement, IEEE Transactions on*, vol. 57, pp. 1379–1387, July 2008.

[169] S. Baghaee, S. Chamanian, H. Ulusan, O. Zorlu, E. Uysal-Biyikoglu, H. Kulah, and H. Kulah, "Demonstration of energy-neutral operation on a wsn testbed using vibration energy harvesting," in *European Wireless 2014; 20th European Wireless Conference; Proceedings of*, pp. 1–6, May 2014.

[170] J. Song and Y. K. Tan, "Energy consumption analysis of zigbee-based energy harvesting wireless sensor networks," in *Communication Systems (ICCS), 2012 IEEE International Conference on*, pp. 468–472, Nov 2012.

[171] H. Kim, D. Choi, S. Gong, and K. Park, "Stray electric field energy harvesting technology using mems switch from insulated ac power line," *Electronics Letters*, vol. 50, pp. 1236–1238, Aug 2014.

[172] IEEE Standard Association, "IEEE std 802.15.4-2009, part 15.4: Wireless medium access (mac) and physical layer (phy) specifications for low-rate wireless personal area networks (wpans)," 2009.

[173] Texas Instruments, "Z-stack - zigbee protocol stack." http://www.ti.com/tool/z-stack, Last accessed July 2013.

[174] Microchip Technology Inc., "Zigbee pro stack." http://www.microchip.com, Last accessed June 2012.

[175] Atmel, "Bitclowd - zigbee pro." http://www.atmel.com/tools/bitcloud-zigbeepro.aspx, March 2014.

[176] P. Baronti, P. Pillai, V. Chook, S. Chessa, A. Gotta, and Y. Hu, "Wireless sensor networks: A survey on the state of the art and the 802.15.4 and zigbee standards," *Computer Communications*, vol. 30, pp. 1655–1695, May 2007.

[177] R. Severino, M. Alves, and A. Koubaa, "On the use of ieee802.15.4/zigbee for time-sensitive wireless sensor network applications," Master's thesis, ISEP, October 2008.

[178] Bluetooth Special Interest Group, "Specification of the bluetooth system, covered core package version." http://bluetooth.org, June 2010. Last accessed May 2015.

[179] Zensys, A/S, *Z-Wave Node Type Overview and Network Installation Guide*. Z-Wave Aliance, March 2007.

[180] Enocean(TM), "Enocean website." http://www.enocean.com/. Last accessed May 2015.

[181] F. P. Miller, A. F. Vandome, and J. McBrewster, *KNX (standard)*. Alpha Press, 2010.

[182] C. Reinisch, W. Granzer, G. Neugschwandtner, F. Praus, and W. Kastner, "Wireless Communication in KNX / EIB," *KNX Scientific Conference*, pp. 1–15, 2006.

[183] Digi International Inc, "The digimesh (tm) networking protocol." http://www.digi.com/technology/digimesh/, Last accessed July 2012.

[184] Jennic, "Jennet protocol stack." http://www.jennic.com/ products/protocol_stacks /jennet, Last Accessed July 2013.

[185] Olsson, Jonas, "6LoWPAN demystified." http://www.ti.com/lit/wp/swry013/swry013.pdf, March 2014. Last accessed May 2015.

[186] J. P. Amaro, F. J. T. E. Ferreira, R. Cortesão, and J. Landeck, "Implementing an advanced meter reading infrastructure using a z-wave compliant wireless sensor network," in *Energetics IYCE 2011. 3rd IEEE International Youth Conference on*, July 2011.

[187] A. De Gloria, ed., *Applications in Electronics Pervading Industry, Environment and Society*, vol. 351. Springer International Publishing, 2014.

[188] Z. Hanzalek and P. JurcÌiÌk, "Energy efficient scheduling for cluster-tree wireless sensor networks with time-bounded data flows: Application to ieee 802.15.4/zigbee," *Industrial Informatics, IEEE Transactions on*, vol. 6, pp. 438–450, August 2010.

[189] Q. Tian and E. Coyle, "A mac-layer retransmission algorithm designed for the physical-layer characteristics of clustered sensor networks," *Wireless Communications, IEEE Transactions on*, vol. 5, pp. 3153–3164, November 2006.

[190] J. Mao, Z. Wu, and X. Wu, "A TDMA scheduling scheme for many-to-one communications in wireless sensor networks," *Computer Communications*, vol. 30, pp. 863 – 872, February 2007.

[191] O. Valle, A. Milack, C. Montez, P. Portugal, and F. Vasques, "Experimental evaluation of multiple retransmission schemes in ieee 802.15.4 wireless sensor networks," in *Factory Communication Systems (WFCS), 2012 9th IEEE International Workshop on*, pp. 201–210, May 2012.

[192] Z. Chen, C. Lin, H. Wen, and H. Yin, "An analytical model for evaluating IEEE 802.15.4 CSMA/CA protocol in low-rate wireless application," in *Advanced Information Networking and Applications Workshops, 2007, AINAW '07. 21st International Conference on*, vol. 2, pp. 899–904, May 2007.

[193] T. R. Park, T. Kim, J. Choi, S. Choi, and W. Kwon, "Throughput and energy consumption analysis of IEEE 802.15.4 slotted CSMA/CA," *Electronics Letters*, vol. 41, pp. 1017–1019, September 2005.

[194] E. Casilari, J. M. Cano-GarcÃÂa, and G. Campos-Garrido, "Modeling of current consumption in 802.15.4/zigbee sensor motes," *MDPI Sensors*, vol. 10, pp. 5443–5468, June 2010.

[195] A. Koubaa, M. Alves, and E. Tovar, "A comprehensive simulation study of slotted CSMA/CA for IEEE802.15.4wireless sensor networks," in *Factory Communication Systems, 2006 IEEE International Workshop on*, pp. 183–192, IEEE, June 2006.

[196] M.-S. Pan, C.-H. Tsai, and Y.-C. Tseng, "The orphan problem in zigbee wireless networks," *Mobile Computing, IEEE Transactions on*, vol. 8, pp. 1573–1584, November 2009.

[197] J.-H. Lee, E.-S. Lee, and D.-S. Kim, "Network joining algorithm for mobile nodes in ubiquitous sensor networks," in *Computer Sciences and Convergence Information Technology (ICCIT), 2010 5th International Conference on*, pp. 836–839, November 2010.

[198] S. Farahani, *ZigBee Wireless Networks and Transceivers*. Newton, MA, USA: Newnes, 2008.

[199] C. Hsiang, A.-W. Chen, C.-J. Chang, B.-Y. Shih, and C.-Y. Chen, "Development of mechanisms for MAC channel selection to improve the performance of IEEE 802.15.4 beacon-enabled network," in *Broadband Network and Multimedia Technology (IC-BNMT), 2010 3rd IEEE International Conference on*, pp. 598–602, October 2010.

[200] J.-W. Kim, J. Kim, and D.-S. Eom, "Multi-dimensional channel management scheme to avoid beacon collision in LR-WPAN," *Consumer Electronics, IEEE Transactions on*, vol. 54, pp. 396–404, May 2008.

[201] N. Langhammer and R. Kays, "Performance evaluation of wireless home automation networks in indoor scenarios," *Smart Grid, IEEE Transactions on*, vol. 3, pp. 2252–2261, August 2012.

[202] M. Galetzka, J. Haufe, M. Lindig, U. Eichler, and P. Schneider, "Challenges of simulating robust wireless sensor network applications in building automation environments," in *Emerging Technologies and Factory Automation (ETFA), 2010 IEEE Conference on*, pp. 1–8, September 2010.

[203] Linear Technology, "LTSPICE IV - SPICE simulator." http://www.linear.com/ designtools/software/LTspice, Last accessed July 2013.

[204] Linear Technology, *LTC3108 Datasheet*, Last Accessed July 2014.

[205] Texas Instruments, *MSP430FR5739 User Guide and Datasheet.* http://www.ti.com/product/msp430fr5739, May 2011. Last accessed July 2014.

[206] Texas Instruments, *Zigbee Pro Zigbee Processor - Accelerate your Zigbee Development.* TI, 2010.

[207] Texas Instruments, *Z-Stack Monitor and Test API.* San Diego, May 2010.

[208] V. Sedlakova, J. Sikula, J. Majzner, H. Navarova, M. Chvatal, and T. Zednicek, "Tantalum and Niobium oxide capacitors: Field crystallization, leakage current kinetics and reliability," in *Microelectronics Proceedings (MIEL), 2010 27th International Conference on*, pp. 439–442, May 2010.

[209] Vishay BCcomponents, *Aluminium Capacitors 150 CRZ Series*, April 2013. Last accessed May 2015.

[210] Cellergy, *Cellergy capacitors.* http://www.cellergycap.com/, April 2013. Last accessed May 2015.

[211] Texas Instruments, *Z-Stack Sample Applications.* Texas Instruments, San Diego, California, June 2011.

[212] Zigbee Alliance, "Zigbee cluster library specificaton." 075123r02ZB, July 2008.

[213] I. Dietrich and F. Dressler, "On the lifetime of wireless sensor networks," *ACM Trans. Sen. Netw.*, vol. 5, pp. 5:1–5:39, Feb. 2009.

[214] J. Kim and C. Park, "A calibration technique for multibit stage pipelined a/d converters via least-squares method," *Instrumentation and Measurement, IEEE Transactions on*, vol. 62, pp. 3390–3392, Dec 2013.

[215] M. Kusljevic, "On ls-based power frequency estimation algorithms," *Instrumentation and Measurement, IEEE Transactions on*, vol. 62, pp. 2020–2028, July 2013.

[216] S. J. Miller, *The Method of Least Squares*. Mathematics Department Brown University Providence, October 2010.

[217] M. Mienkina, *Filter-Based Algorithm for Metering Applications*. Freescale Semiconductor, December 2013. Last accessed May 2015.

[218] G. Hudson and T. Lean, "Open energy monitor." http://openenergy monitor.org/emon/, September 2014. Last accessed in May 2015.