PhD Dissertation

International Doctorate School in Information and Communication Technologies

DISI - University of Trento

Power Management and Power Consumption Optimization Techniques in Wireless Sensor Networks

Andrey Somov

Advisor:

Prof. Roberto Passerone

Università degli Studi di Trento

December 2009

Abstract

A Wireless Sensor Network (WSN) is a distributed collection of resource constrained tiny nodes capable of operating with minimal user attendance. Due to their flexibility and low cost, WSNs have recently become widely applied in traffic regulation, fire alarm in buildings, wild fire monitoring, agriculture, health monitoring, building energy management, and ecological monitoring. However, deployment of the WSNs in difficult-to-access areas makes it difficult to replace the batteries - the main power supply of a sensor node. It means that the power limitation of the sensor nodes appreciably constraints their functionality and potential applications. The use of harvesting components such as solar cells alone and energy storage elements such as super capacitors and rechargeable batteries is insufficient for the long-term sensor node operation. With this thesis we are going to show that long-term operation could be achieved by adopting a combination of hardware and software techniques along with energy efficient WSN design. To demonstrate the hardware power management, an energy scavenging module was designed, implemented and tested. This module is able to handle both alternating current (AC) based and direct current (DC) based ambient sources. The harvested energy is stored in two energy buffers of different kind, and is delivered to the sensor node in accordance with an efficient energy supply switching algorithm. The software part of the thesis presents an analytical criterion to establish the value of the synchronization period minimizing the average power dissipated by a WSN node. Since the radio chip is usually the most power hungry component on a board, this approach can help one to decrease the amount of power consumption and prolong the lifetime of the entire WSN. The following part of the thesis demonstrates a methodology for power consumption evaluation of WSN. The methodology supports the Platform Based Design (PBD) paradigm, providing power analysis for various sensor platforms by defining separate abstraction layers for application, services, hardware and power supply modules. Finally, we present three applications where we use the designed hardware module and apply various power management strategies. In the first application we apply the WSN paradigm to the entertainment area, and in particular to the domain of Paintball. The second one refers to a wireless sensor platform for monitoring of dangerous gases and early fire detection. The platform operation is based on the pyrolysis product detection which makes it possible to prevent fire before inflammation. The third application is connected with medical research. This work describes the powering of wireless brain-machine interfaces.

Keywords
[Wireless Sensor Network, Power Management, Platform Based Design, Energy Scavenging, WSN Synchronization, Gas Sensor, Wireless Brain-Machine]

Viam supervadet vadens

Acknowledgments

This is a great opportunity to express my gratitude and respect to all the people who supported me while I was doing my PhD. They all contributed to this work each in his/her own way.

It is difficult to overstate my gratitude to my PhD supervisor Prof. Roberto Passerone. First, I would like to thank him for guiding and inspiring me during my research period at the University of Trento and for showing me what research is about. Secondly, I highly appreciate his support not only in my research work but also in different life situations where he was assisting me in any possible way. Finally, I should thank him for provided encouragement throughout my thesis-writing period and the final prove ready of this work. I would have been lost without him.

I would like to express my gratitude to Prof. Alexander Baranov (Moscow Aviation Technological University) who encouraged me to enter the University of Trento and invigorated me throughout the entire period of research.

I would like to thank Prof. Jan Rabaey, for giving me a great opportunity to work during four months in his research group at UC Berkeley. A special thank I need to give to one of the members of his group Michael Mark, who, since my arrival, became my research companion. We performed a lot of research work together and continue our collaboration. I am also pleased to thank other researchers I met at the UC Berkeley and in particular Christine Ho, Louis Alarcon, and David Chen.

I would like to thank all the administrative and technical staff of the University of Trento, and in particular Galina Kamburova, Manuel Zucchellini, Alessandro Tomasi, and Sebastiano Perisi.

I am indebted to my many student colleagues from University of Trento for providing a stimulating and fun environment in which all of us learn and grow. I am especially grateful to Anton Ageev, Andrei Papliatseyeu, Ivan Minakov, Olga Zlydareva, Nadzeya Kiyavitskaya, Yury Audzevich, and Mikalai Krapivin. Being my close friends, they were

always ready to help and support me. I would also acknowledge my friends from an enormous Russian-spoken community at our university, and in particular I am very thankful to Marina Repich, Ruslan Asaula, Alena Simalatsar, Ivan Tankoyeu, and Denis Babenko.

Lastly, and most significantly, I wish to thank my parents Nina Somova and Sergey Somov and my girlfriend Julia. They helped me, believed in me, supported me, and loved me. To them I dedicate this thesis.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This section introduces the problem and provides its positioning in the framework of the main research areas of ICT (telecommunications, computer science, electronics).

1.1 The Context

From the very beginning, Wireless Sensor Networks (WSN) have been applied to various difficult-to-access areas: environment and industrial monitoring, emergency services and tracking of movements. WSNs provide the opportunity to build autonomous systems without the need to wire up an entire network, thus progressively replacing wired embedded systems.

A WSN consists of wireless nodes which measure physical conditions using sensors (i.e., temperature, humidity, vibration, pressure), digitize it and keep or distribute the measured data over the network. The typical WSN architecture and data distribution is depicted in Figure 1.1.

According to the application requirements, the wireless sensor nodes can be deployed over a large area or can be amassed in a small section for the specific data measurements. Applying a multi hop paradigm [27] the sensor nodes communicate the data to one or more gates. The gate is a device which serves as a data sink and manages the "usual" nodes. In fact, either one or more sensor nodes are able to operate as the gates, or another wireless device can be assigned as a gate. The gate communicates the data to the user via a data

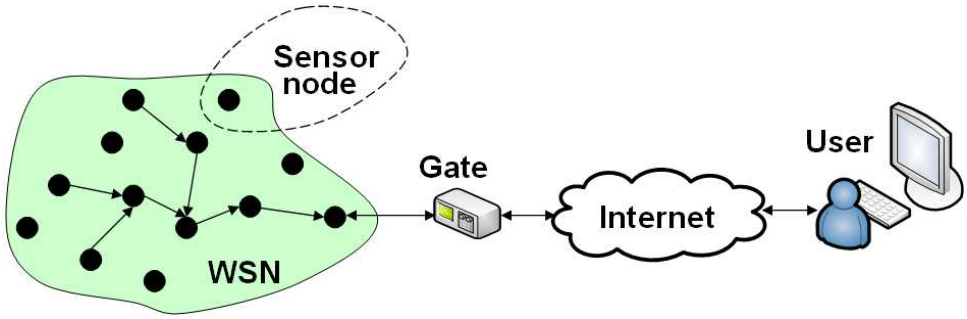

Figure 1.1: Typical WSN architecture and networking.

communications network, e.g., Internet. However, the feed back "user-sensor node" allows the user to monitor the network, and do appropriate configuration and management, if necessary.

A WSN contains a number of sensor nodes connected by a radio channel. A typical sensor node block diagram is presented in Figure 1.2. It has four main blocks on a board: a Central Processing Unit (CPU), memory (e.g., program memory, flash memory, volatile memory), input-output (I/O) ports and a radio. Depending on the application, the sensor node may have various extra hardware components on board.

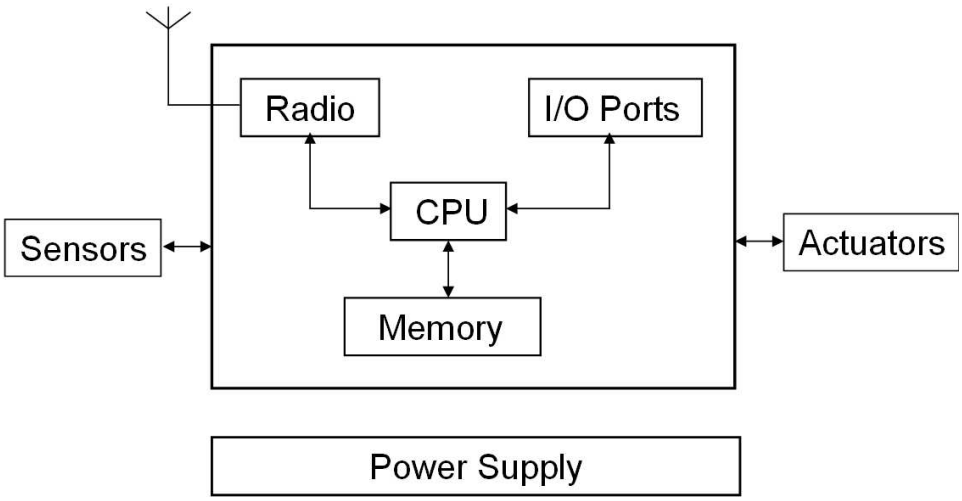

Figure 1.2: Sensor node anatomy.

The sensor nodes have to meet several important requirements:

Low power consumption: due to the fact that the WSN is energy constrained, it must

operate using power efficient hardware and software technologies to increase its lifetime. Energy scavenging technology can improve the WSN lifetime and decrease the cost of the network maintenance: the nodes are usually powered by AA cells, but the batteries must be replaced when exhausted. According to OnWorld Research the battery replacement cost will be $1 Billion in 2013 [44]. Energy scavenging techniques are aimed at decreasing or even canceling the network maintenance cost;

Small form factor: to make it possible to embed the WSN into the daily environment, a high level of component integration of sensor node is required;

Low cost implementation: to increase the network reliability and accuracy of measured data the density of the network must be high, therefore the cost per node must be minimal.

Along with the listed advantages there is a serious limitation which will be described below.

1.2 The Problem

Nowadays, WSNs are adopted in various applications: environmental and industrial monitoring, vehicle detection, agriculture, entertainment and in military application. However, the challenges associated with the efficient power management and lifetime of WSN nodes significantly constrain their functionality and potential applications. Moreover, the energy constraints lead to the cost increase of WSN maintenance, especially for the networks deployed in remote areas. Thus, the problem of WSN long-term operation is of vital importance.

The lifetime problem is a complex problem of WSN and can not be resolved one-sidedly. It has to be addressed at three main levels:

Design - application, modeling, simulation;

Hardware - hardware components, technology, energy efficient conversion and storage, maximum power point tracking, energy scavenging technology;

Software - communication protocol, middleware, energy saving techniques.

At the design level the entire network operation and configuration depends on the application. This application may determine the hardware architecture and power management strategy of the sensor node and network. Besides, prior to hardware implementation of the node it should be modeled and the entire network has to be simulated.

The lifetime problem at the hardware level includes the correct HW components choice along with energy efficient conversion and storage. Energy scavenging technology may considerably increase the WSN lifetime, but for the maximum efficiency it is desirable to apply it with the maximum power point tracking (MPPT) technique [89]. MPPT refers to drawing power from an energy harvesting source at a level that maximizes the power output. In fact, the MPPT can be either hardware or software.

Software plays an important role in WSN lifetime as well. For example, the adaptive duty-cycling algorithm [66] allows the utilization of up to 58% more environmental resources in comparison to the systems without this technique. Due to the fact that the radio is the most power "hungry" component on board, the energy efficient middleware and energy efficient techniques are required.

1.3 The Solution

To overcome the lifetime problem of WSN we address it at three main levels: the design level, the hardware and the software, as defined previously in Section 1.2.

At the design level we propose a methodology for power consumption evaluation. This methodology is a flexible and extensible simulation framework for power consumption estimation of sensor network applications for arbitrary HW platforms. This framework allows designers of sensor networks to estimate power consumption of the explored HW platform which permits the selection of an optimal HW solution and SW implementation for the specific projects.

The designed EnTick platform is aimed to resolve the WSN lifetime problem from the HW point of view. This platform supports an energy scavenging technology which enables "*perpetual*" operation of sensor nodes. Besides, the EnTick contains a two-level

energy buffer based on off-the-shelf components such as rechargeable batteries and super capacitors, so that the sensor node can be supplied either from super capacitors or the battery or directly from the ambient source. The energy is delivered to the sensor node in accordance to an efficient energy supply switching algorithm. In addition, we have developed a modification of the EnTick platform called "TreBer". The TreBer platform is identical to the EnTick architecture and uses the same HW components, but instead of off-the-shelf super capacitors we use an electrochemical supercapacitor (with improved electrical characteristics) printed directly on the bottom of the module using the "direct write" technology [29]. The smaller size of this platforms increases the potential applications of WSN, and support the harvesting components of different kinds.

Since the radio module is usually the most energy-consuming component in tiny, low-cost sensor devices, node long-term operation can be extended by duty cycle reduction of the radio chip. The second way of energy saving on the radio is simply to turn it on to run the task of assigned application, whereas it could be in sleep mode at the remaining time. In fact, this approach is effective in case when the tasks are scheduled periodically and if the nodes are synchronized well, i.e., have a common clock. In reality, if the network is unsynchronized, some of the nodes may wake up while others are sleeping, so that some connections can not be established. Due to the fact that time synchronization among the nodes can be maintained within the bounds of known uncertainty solely by recurrently adjusting the local clocks, synchronization tasks can also be scheduled periodically. According to this, we devised an analytical criterion which determines the value of the synchronization period to minimize the average power consumed by a sensor node. The proposed approach is validated by both simulations and experiments.

1.4 Innovative Aspects

At the design part of the thesis we present a methodology for power consumption estimation of a sensor node. Except for the traditional three layers of abstraction, i.e., *application, service* and *platform* layers, we introduced a fourth *energy source* layer. The methodology is implemented as a SystemC-based framework that combines the event-

driven simulation engine and HW model composer, and allows a user to describe the application using a set of service calls and user functions described in the C/C++ language. Another contribution of the framework is that the evaluation procedure is automatic. Besides that, we have strengthened current tools for power consumption analysis by adding the *energy source* layer which enables a power management of the entire system on the basis of battery model. Therefore, the developed methodology provides a user with a flexible and detailed analysis of power consumption of the sensor node.

The core idea of the HW part is to apply a modular uniform technology for WSN (see Figure 1.3) with a power management technique. The entire system contains three main parts: a harvesting component - the energy scavenging module (the EnTick) - the wireless sensor node. Of course, we have in mind that ambient power source is the main power source for the all the blocks in the system. A harvesting component, for instance a solar panel or a piezoelectric converter, converts ambient energy into electric energy. As a matter of fact, the EnTick platform is a mediator between a harvesting component and the sensor node: any sensor node can be connected to any harvesting component via the mediator which enables the harvesting of both (AC) and (DC) based ambient energy. Besides, the mediator enables to keep the collected energy in two-level storage - rechargeable battery and two super capacitors wired in series.

The TreBer platform has the printed capacitors on board manufactured using the "direct write" technology.

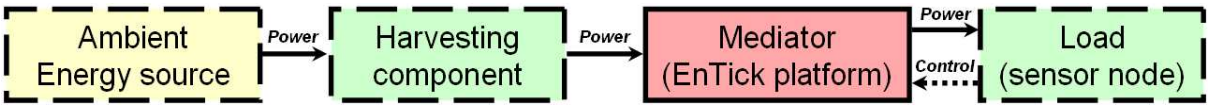

Figure 1.3: Modular uniform technology.

Afterwards, we investigated how synchronization of WSN affects the power consumption. An optimal synchronization technique conduces to lifetime increase of the network. This approach proves that the average power consumption of a generic WSN node can be minimized without undermining communication reliability if the interval between subsequent synchronizations is set equal to a specified optimal value. The proposed analysis

has been validated by means of both simulations and experimental results and opens new perspectives in the design of long-term monitoring applications for WSNs. The devised formula for optimal period of synchronization will support the long-term operation of the WSN node at the software/middleware level.

Chapter 2

State of the Art

This chapter presents the state of the art pertaining to the research work presented in this doctorate thesis. It includes the most relevant materials that form the background of the presented research activity as well as a selection of the significant works performed by other research groups in the world. The first four sections are dedicated to power management at the design level. They introduce the system-level design methodology, system-level design language, simulation frameworks, and overview of battery models. After that we talk about energy harvesting technologies, energy storage devices, and discuss how to scavenge, convert, and store ambient enery efficiently. Next we provide a description and critical discussion of academic as well as industrial wireless sensor platforms with energy harvesting technology application. Finaly, we overview recent software power management techniques and describe the synchronization uncertainty together with synchronization protocols used in WSN.

2.1 System-Level Design Methodologies

In this section we are going to present several System-Level Design (SLD) methodologies used for the design space exploration of Wireless Sensor Networks (WSN).

2.1.1 Platform-Based Design

Platform-Based Design (PBD) is a methodology that combines the specification, validation and synthesis steps of the design flow, while maintaining a clear separation between the corresponding models [35, 100]. By doing so, the designer can operate separately on the distinctive steps and maintain a global view of the impact of his/her design decisions on the final implementation. The methodology includes hardware and embedded software design, where the design of the system starts at a high level of abstraction (initial design description) and proceeds to a detailed implementation by mapping the executable functional model onto progressively more detailed architectures under a set of constraints.

2.1.2 Rialto Framework

There exist several SLD methodologies applying PBD principles developed in the area of WSN design space exploration. Bonivento et al. have presented a new methodology for the design of WSN [30], which is well correlated with the nature of our research work. They introduced a framework called Rialto that initially included two basic platforms, the application interface called Sensor Network Service Platform (SNSP) and the hardware platform layer called Sensor Network Implementation Platform (SNIP). Lately this framework was extended by a third intermediate layer called Sensor Network Adhoc Protocol Platform (SNAPP) that defines a library of communication protocols and the interfaces that these protocols offer to the SNSP. This framework allows the application description independently from the network architecture. The sensor network service platform is used to describe an application in a Rialto Model in terms of logical component queries and commands. This model is then translated into RialtoNet format that allows exploration of the possible sequence of queries and commands from the application. Later the functional description is mapped into an architecture platform instance. This work is focused mainly on high level operating system services, while we will focus on single node platform power consumption estimation.

2.1.3 COSI Framework

The COmmunication Synthesis Infrastructure (COSY) [91] developed by Alessandro Pinto et al. is a framework for the design exploration and synthesis of interconnection networks, where "interconnection networks" refers to networks on-chip (NoC) as well as distributed embedded systems. They introduce a general methodology for the design space exploration of networks that allows studying of the optimization algorithms, communication protocols, partial designs, and models for interconnection design in terms of network performance and cost. We can apply this methodology in the future while extending our methodology for being used for a whole WSN power consumption estimation.

2.2 SystemC: System-level Design Language

SystemC is a language created by the Language Open Group (LOG) of the Open SystemC Initiative (OSCI), and is targeted to a wide range of designers. SystemC supports different models of computation and allows the design of heterogeneous systems [108], [59]. Basically, SystemC is a C++ class library, where C++ plays the role of language foundation while the library provides both a notion of process and interface, and a simulation kernel based on the Discrete-Event model. The SystemC library includes its own structural elements like modules, ports, interfaces and channels (signals, FIFO, mutex, semaphores etc.). It also introduces new data types, such as 4-valued logic, bits and bit vectors, arbitrary precision integers and fixed-point types, which are useful in the specification of hardware components. Additional libraries, like the Verification Library and the Heterogeneous System Specification Methodology Library (HetSC), have also been developed to extend the original functionality of the language. SystemC suits very well for building executable hardware and software models, but it does not support mapping of functional models into hardware platforms.

2.3 Network Simulators

This section reviews the most popular network simulators in the WSN research community.

2.3.1 NS-2 simulator

NS-2 [11], perhaps, is the most popular general purpose network simulator. NS-2 supports simulation for widely used IP network protocols. These include TCP, routing and multicasting protocols for conventional wired and wireless networks. NS-2 has a highly extensible object-oriented architecture with discrete-event engine. Its object-oriented model allows extension of simulation functionality by adding customers components and libraries. The simulation in NS-2 environment is based on a combination of C++ and OTcl [14] languages where protocols are implemented in C++. OTcl is used as a scripting language to describe and control the simulation process. The complexity of NS-2 object-oriented model creates substantial dependencies and execution overheads. It makes impossible to scale simulation for a large number of network units, which is inherent to WSNs. While object-oriented model is advantageous in terms of extensibility, it is a restriction for scalability and performance. Besides, NS-2 does not provide representation for the HW network components.

2.3.2 OMNet++ simulator

Like NS-2, OMNet++ [13] provides deep analysis of network activities at the packet layer. Besides, OMNet++ provides a GUI front-end for simulation and debugging processes. It has a component-based architecture with a discrete-event simulation kernel. It exploits modules and channels to implement and connect simulation components, where components are connected in a hierarchical fashion via generic interfaces (gates). OMNet++ has extension for sensor network simulation, called SenSim [67]. It represents sensor node as modular hierarchical structure of simple OMNet++ components. This simulatorprovides more scalability and runs faster than NS-2. However, despite the apparent benefits of OMNet++ and SenSim, there is no precise and accurate HW model of sensor node. It,

in turn, does not allow to study sensor networks from an energy perspective.

2.3.3 TOSSIM

TOSSIM simulation environment is included in the TinyOS [21] framework. TinyOS has gained general acceptance as a standard operating system for WSN applications. It has a component-oriented programming model, based on the nesC language [9]. A TinyOS program is presented as set of components, where each componnet is an independent computational entity. The TinyOS framework includes a simple FIFO task scheduler and hardware independent drivers for abstract HW components. The inter-component communications occur through command-event mechanism. By changing a small number of TinyOS components, TOSSIM simulates the behavior of the low-level hardware. It includes models for CPUs, analog-to-digital converters (ADCs), clocks, timers, flash memories and radio components. The network communication over the wireless channel is abstracted as a directed graph, where vertexes and edges represent nodes and links between them, respectively. TOSSIM simulation architecture provides high level of scalability and execution speed for the networks with large number of sensor nodes. However, the abstract HW model of TOSSIM does not capture low-level details of timings and interrupts, which can be important for precise power analysis. In addition, simulation is supported only for the single HW platform (Micaz [43]). Obviously, it largely restricts the applying scope.

2.3.4 VIPTOS and VisualSense

VIPTOS (Visual Ptolemy and TinyOS) is a graphical development and simulation environment for TinyOSbased WSN applications [40]. VIPTOS bridges together the VisualSense [22] simulator and the TinyOS framework. VisualSense is a Ptolemy II [17] based graphical simulation environment designed for WSNs. It exploits the actor-oriented computational model of Ptolemy II, a general modeling framework for heterogeneous embedded systems. VisualSense defines actor-oriented models for sensor node subsystems and communication channels.

However, VIPTOS does not provide accurate HW representation of sensor node. Substantially, it focuses more on algorithmic and application domains. Additionally, VIPTOS has been integrated only with the first version of TinyOS, which is not currently supported.

2.3.5 AVRORA

AVRORA [109], like TOSSIM, is one of the widely used WSN simulation tool. It exploits cycle accurate instruction-by-instruction manner to run code. AVRORA runs actual applications without the need to specially adapt it for simulation. AVRORA represents each HW component as corresponding object classes thus as classes of CPUs, Timers, flash memories, ADCs and off-chip components such as sensors. The HW model of a single sensor node is the combination of such objects in a hierarchical manner. The CPU object contains the simulation engine with the event queue for the entire node. This architecture allows node replication for network simulation, where each node is run as independent computational entity. However, AVRORA supports solely AVR MCU [1] cores and does not provide any extensions for others CPU architectures.

2.4 Battery Models

The battery model is an essential part of the energy consumption evaluation framework to analyse power dissipated during a sensor node operation.

Several battery models have been designed recently. However, in most cases each type of the model makes an emphasis on specific battery purpose and has various levels of accuracy. Mathematical battery models [94, 97] apply empirical equations to address the battery charge/discharge behavior, efficiency, or simulate the electrochemical or thermal behavior of the battery. In fact, the mathematical models can not be adopted for circuit simulation since its accuracy lies in the range of 5-20% and do not provide one with the current-voltage characteristic.

Electrochemical models [105, 47] are complex and require much time for simulation since they typically present the battery at low level. However, the electrical models [60, 31, 58] are the most appropriate for circuit analysis, simulation and optimization. They are

easy to handle, provide the electrical engineers with the current-voltage characteristics, and, finally, their accuracy lies within 5%.

The electrical battery model presented in [39] takes into consideration all dynamic characteristics of the battery. It consists of two parts: battery lifetime evaluation and voltage-current characteristics emulation. Battery lifetime evaluation part contains a resistor, which determines energy loss during long idle period, a capacity to characterize battery charge, and a current, which is applied to charge or discharge the battery. The voltage-current characteristics emulation part of the model contains open-circuit voltage, which is able to change according to the state of charge (SOC) of the battery, and an RC network. This model is both intuitive and capable to predict an accurate runtime and voltage-current parameters.

2.5 Energy Harvesting

This section reviews conventional outdoor energy sources. The goal of this section is not to suggest the best way of ambient energy conversion to power wireless sensor node, but to understand their potential as an alternative power source.

Renewable environment sources have inherent advantages compared to nonrenewable sources like oil, coal, gas, and nuclear. The merits are as follows: they are non-polluting, and do not require maintenance.

As it is presented in Table 2.1 the most attractive way for energy harvesting is solar light. For all technologies, the energy is available only under particular environmental conditions: sunlight, vibration, high temperature, surrounding noise. Each given form of energy can be harvested by a different class of generator that performs conversion to electricity. That is why wireless sensor nodes have to keep energy in storage elements. In this case, a reasonable solution seems to be the combination of an energy storage system and scavenging technology. Various power sources were explored, but we will consider only the most adequate ones for application in WSN domain.

The first three sources provide the higher power density and are therefore more applicable for WSN. Photovoltaic power is the most popular ambient energy source for

Harvesting technology	Power density
Solar cells (outdoor at noon)	15 mW/cm^2
Piezoelectric (shoe inserts)	330 $\mu W/cm^3$
Vibration (small microwave oven)	116 $\mu W/cm^3$
Thermoelectric 10 °C gradient	40 $\mu W/cm^3$
Acoustic noise (100 dB)	960 nW/cm^3

Table 2.1: Ambient power sources [98].

wireless nodes with energy scavenging technology amid designers. Photovoltaic power can be produced in many ways with widely varying efficiency and cost. Solar technology can be divided into two main groups: discrete cell technology and integrated thin film technology.

Discrete cell technology:

- Single-crystal silicon (15% efficiency of commercial modules);
- Multicrystalline silicon (14% efficiency of commercial modules);
- Edge-defined film-fed growth ribbons;
- Dendritic web;
- Gallium Arsenide (28% efficiency of research cell under concentrated sunlight).

Integrated thin film tehnology:

- Copper Indium Diselenide (17.7% efficiency of research cell);
- Amorphous Silicon;
- Cadmium Telluride (8.34% efficiency of commercial modules [19]).

The technologies above are well described in the literature. However, photovoltaic engineers developed advanced design for the discrete cell technology as concentrated photovoltaic technology [2] which allows the device to focus incident solar light to the cell

by means of three optical elements and increase conversion efficiency. But in this case, conversion efficiency is reduced a bit because of overheating a cell during the radiation concentrating process.

The core idea of innovative holographic technology, patented by Prism Solar Technologies [16], lies in using transparent holographic optical elements to collect, spectrally select and focus useful wavelength of solar light onto the cell. The advantages of this technology are: up to 85% economy of the silicon, transparent properties, low cost.

Vibration harvesting modality has less power density than solar. Three conversion mechanisms are distinguished by which vibrations can be converted to electrical energy. They are piezoelectric, electro-magnetic, and electrostatic. In piezoelectric conversion, mechanical strain in a piezoelectric material produces a charge separation across the material, generating a voltage. Regarding the electro-magnetic case, the relative motion between a coil and a magnetic field produces a current to flow in the coil. Electrostatic generation, in turn, contains two conductors separated by a dielectric, which move relative to one another. As the conductors move, the energy stored in the capacitor changes, thus providing the mechanism for mechanical to electrical energy conversion [98].

Wireless sensors platforms with energy harvesting devices are considered in Section 2.8.

2.6 Energy Storage Devices

Wireless sensor node with the energy scavenging technology application requires a buffer to store harvested energy. This energy buffer allows the system to be supplied even if an ambient source is unavailable at the moment. There are two popular energy storage technologies exploiting in WSN: rechargeable battery and supercapacitor. Both technologies have advantages as well as disadantages. Thus, there are three typically used combinations of these storage devices application: supercapacitor only, rechargeable battery only or combination of them to supplement each other.

Of course, we mentioned the most popular storage devices. However, these devices can be divided into sub groups. Besides, there are other power supply devices for sensor nodes, but they are not widely applicable. Table 2.2 summarizes the energy storage technologies

applied in WSN.

Rechargeable battery (or a secondary energy buffer) is usually considered as a backup energy source in WSN. It is being used when the supercapacitor (or a primary energy buffer is exhausted). It happens because of the limited number of the battery charge/discharge cycles (300-500 cycles). However, rechargeable batteries have significant energy density and low leakage current that provides the long-term energy storage (see Table 2.2).

There are four main kinds of rechargeable batteries: NiCd, NiMH, Lithium-ion, and Lithium-polymer. Lithium cells have the highest energy density, the lowest leakage current per month, but require additional circuit for the charging. A single lithium cell provides a system with high voltage. NiCd and NiMH rechargeable batteries have simple charging circuit (or in individual case they do not require it at all).

2.6.1 Basic Battery Theory

A battery is made of two electrodes, a positive cathode and a negative anode, with a porous separator between them. The time of battery charging is normally determined by the milliamp hour (mAh) capacity of the battery. For a 400 mA source, for example, charging a 400 mAh battery will take 400/400 = 1 hour (C), while charging a 200 mAh battery will require 200/400 = 0.5 hours (0.5C). However, in reality, charging a battery requires minimum 1.5*C to secure the complete replenishment [17]. In order to avoid confusion, capacitance is designated as C (in italics) as distinct from battery capacity indicated with C.

Though progress in battery technology is not as rapid as in electronics and computing, with new battery chemistries and enhanced manufacturing techniques it has become possible to create the smaller and more reliable batteries for handheld electronics. By the present almost every small, portable electronic device is powered by rechargeable batteries of the nickel metal hydride or lithium ion/polymer types. In the situations of a wall outlet or a car adapter being easily accessible, the efficiency of battery charging is rarely taken into consideration. It becomes crucial with the increase in demand for wireless sensors that can operate indefinably long. In some cases, the applicability of other energy storage

Type	**Model**	**Op.voltage/ Capacity, mAh**	**Charge cycles**	**Leakage, %/month**	**Energy density, Wh/Kg**	**Memory effect/ charging**
NiCD	Sanyo KR-1100AAU	1.2/1100	>500	25-30	55	Yes/trickle
NiMH	Sanyo HR-AAU	1.2/1500	>500	30-35	64	No/trickle
Li+	Sanyo UR18500H	3.7/1450	300-500	<10	160	No/pulse
Li-polymer	Sanyo UPF386369	3.7/1500	300-500	<10	163	No/pulse
Supercap	Aerogel B-series	2.5/22F	>100000	25/day	1.73	No/trickle
Printable capacitor	UC Berkeley	1.5V/40mF/ cm2	>120000	N/A	10uW/ cm2	No/trickle
Solid state thin film battery	Cymbet CBC-050	3.8/50uAh	>5000	N/A	N/A	N/A /trickle

Table 2.2: Energy Storage Devices

devices (such as a capacitor) should be considered - these cases will be discussed later.

2.6.2 Rechargeable Nickel Metal Hydride Batteries

Nickel metal hydride (NiMH) batteries use hydrogen, absorbed in a metal alloy, as the active negative material as opposed to cadmium used in nickel cadmium (NiCd) batteries [79]. In rechargeable applications, NiMH batteries have, for the most part, replaced nickel cadmium (NiCd) batteries. As compared to their NiCd relative, NiMH batteries are not potentially harmful to environment, Furthermore, NiMH have no signs of the so-called "memory effect" associated with the NiCd variety. The most widespread charging method for NiMH batteries is a constant-current charge using limited current to avoid too great an uprise of the battery temperature or not to exceed the rate of the oxygen-recombination reaction [79].

Being easily accessible, nickel metal hydride batteries are known for their extremely high discharge rate which makes their application for storing such relatively small amounts of harvested energy rather questionable. In general, a nickel-based battery will discharge 10% to 15% of its capacity within first 24 hours after charging following which the discharge rate makes additional 10% to 15% per month (cf. the li-ion self-discharge of approximately 5% for the first 24 hours and further 1% to 2%) [46].

2.6.3 Rechargeable Lithium-based Batteries

The recent trend has been towards replacing nickel metal batteries with lithium ion batteries characterized by even greater specific energy and energy density [79]. In addition, lithium ion cells have considerably higher discharge voltage (3.6 V), much lower self-discharge rate as compared to NiCd and NiMH, and do not exhibit any memory effects [87].

Rechargeable lithium ion batteries utilize a reversible insertion and extraction of lithium ions into and from a lithium insertion compound during the discharging and charging cycle [87].

The nominal open-circuit voltage of most widely used rechargeable lithium ion cells is 3.7 V.

The capacity loss associated with the lithium's side reactions inside the cell increases with the depth of discharge per cycle. Some of the secondary lithium cells can reach many cycles, though granted very shallow discharge conditions [71]. In other words, while the majority of lithium based secondary batteries claim between 500 and 1000 charge cycles of theoretical life, the actual life of the battery is typically much shorter than the upper limit.

For safety reasons, the maximum charge and discharge current for a lithium ion battery is limited to 1 C, though some up-to-date chemistries are able to ensure the charge/discharge rate of up to 20 C. One of the core issues of lithium technology is the fact that lithium metal starts melting at 180 °C. The liquid metal is highly reactive as compared to solid state. Upon reaching its melting point, lithium tends to react with the cathode material and the electrolyte's components providing a considerable amount of thermal energy [71]. For commercial and military aerospace applications this could mean disaster or destruction during high altitude or outer space missions.

Unlike nickel metal hydride batteries, lithium ion charging requires additional circuitry to prevent overcharging or overdischarging of the cell. Overcharging may be dangerous, as described above, and both conditions can reduce the overall life of the battery. However, despite the dangers associated with lithium ion batteries, they remain the portable option of choice due to relatively high specific energy and energy density.

2.6.4 Solid-state Thin Film Batteries

Cymbet^{TM} Corporation manufactures the solid-state, rechargeable thin film batteries. The batteries are packed into a tiny case that enables to use them directly on board of a sensor node. In fact, the thin film battery technology used by Cymbet was originally developed at Oak Ridge National Laboratory [12].

For many applications demanding high energy and power densities, good capacity retention for thousands of discharge/charge cycles, and an extremely low self-discharge rate, the ideal micropower sources are rechargeable thin-film solid-state lithium batteries. They are able to secure high power levels, long cycle life, insignificant self-discharge and attractive charge rates [49]. The battery technology is based upon a sequence of physical

vapor deposition processes [50]. Metallic lithium anode films are deposited by thermal evaporation of the metal under vacuum. The lithium batteries' capacity is limited by the thickness of the cathode films, as the lithium film is several times thicker than that required to cycle the battery.

The batteries fabricated using this technology can be adopted for a number of applications including backup power, implantable medical devices, redio-frequency transmitters.

2.6.5 Supercapacitors

Supercapacitors, also referred to as ultracapacitors or electrochemical double layer capacitors, differ from the conventional electrostatic and electrolytic capacitors as they contain an electrolyte which enables the electrostatic charge to also be stored in the form of ions [106]. Being governed by the same fundamental equations as conventional capacitors, they utilize the larger surface area electrodes and thinner dielectrics to achieve greater capacitances [61]. Since these devices are designed to store energy using both ionic capacitance and surface redox reactions, their classification is closer to the standard battery's one than to the classification of their conventional capacitor relatives.

It should be noted that, in literature, the two designations (super- and ultra-) are often used interchangeably and are provided with vague definitions which only contribute to the confusion rather than clarify it. Since both words are, in fact, applicable to the same device, it has been proposed to designate such devices under the generic name 'electrochemical capacitor' [106]. However, the term supercapacitor will still be used hereinafter for convenience.

A model of a capacitor should comprise an Equivalent Series Resistance (ESR) to account for internal losses. Depending on applications, the ESR may significantly affect the voltage fluctuation across the capacitor during charging and the rate of current leakage out of the capacitor over extended periods of inactivity.

2.6.6 Solid-state Electrochemical Capacitors

The utilization of organic solvents [42], and more recently novel ionic liquids [77] as supercapacitor electrolytes has enabled greater electrochemical stability over larger potential

ranges than aqueous electrolytes that are only stable up to $\Delta 1$ V. As a result, energy density, being a function of the square of the potential difference achievable in a capacitor, increases significantly. Ionic liquids especially are interesting because they effectively remain liquid at room temperature with negligible vapor pressure, and therefore are not volatile [56]. Besides being safe and requiring less hermetic packaging, these novel liquids provide an interesting opportunity when incorporated with polymer gels to form solid-like films that maintain liquid-like properties. Developed a gel electrolyte with high mechanical strength and good ionic conductivity by incorporating 1-butyl-3-methylimidazolium tetrafluoroborate (BMIM+BF4-) ionic liquid with polyvinilydene fluoride (PVDF) polymer [64]. Using this gel electrolyte, solid-state electrochemical capacitors can be deposited utilizing direct write solutions processing.

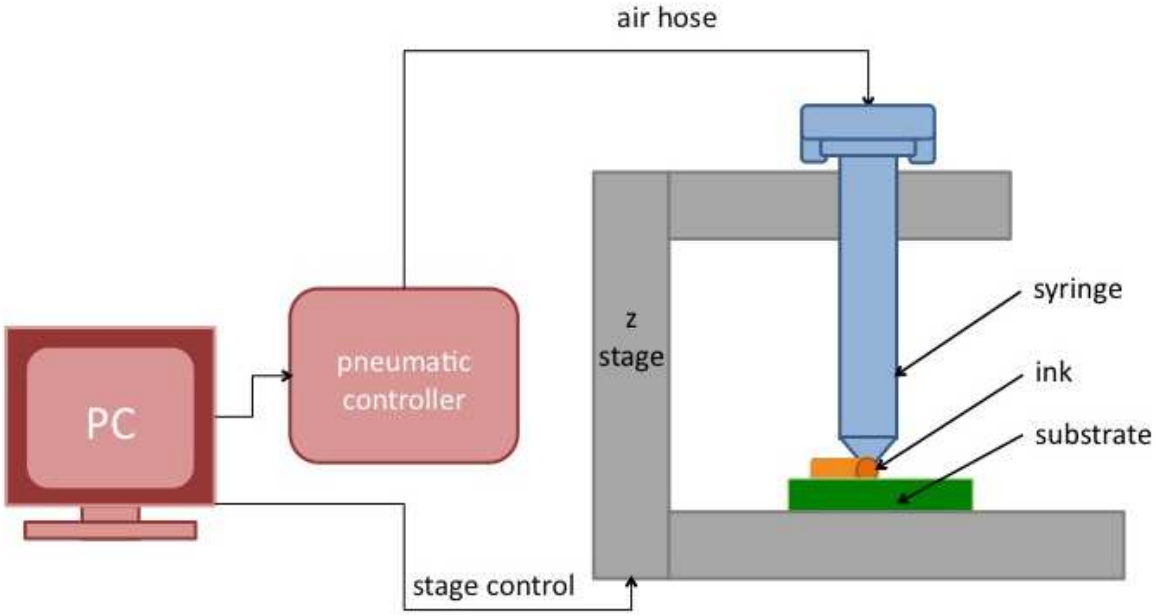

Figure 2.1: Schematic of direct write, pneumatic dispenser printer [courtesy of C. Ho, UC Berkeley].

Direct write solutions processes are typically simple methods for depositing materials additively onto a substrate at ambient and room temperature conditions. They provide a suitable alternative to standard thin film microfabrication techniques for devices that benefit from thick films or non-planar geometries such as electrochemical capacitors. We

have developed a pneumatic dispenser printing tool which extrudes slurries and solutions onto a substrate mounted on a 3-axis stage with micron accuracy (Figure 2.1). Because the deposition of solutions is gentle, films of different compositions can be deposited in a subsequent manner, and devices can be built. Several direct write processes have been used to deposit similar structures [63, 29], but the printer provides the added benefits of reducing energy inputs and waste generated, as well as being able to deposit material around existing components, effectively utilizing any open space on a substrate. Earlier results have shown this method to be both cost effective and scalable for the mass production [107].

2.7 Energy Efficiency and Power Management

It should be noted that efficiency may decrease due to the direct connection between harvesting components and energy storage devices. Efficiency break-down can be presented as follows: efficiency of converting one form of energy to another, efficiency of transferring from the source to the supply, efficiency of buffering after the energy has been harvested and, finally, consumption efficiency.

The first thing to be considered with regard to energy harvesting is voltage. Insufficiently high voltage makes it difficult (if not impossible) to either directly power the system or charge an energy storage device. Such systems as wireless sensor nodes are built using analog and RF components that are sensitive to noise in the power supply. What is needed, therefore, is voltage conversion to some known, controllable level to secure consumption of regulated power.

The function of filling up the gap between the supply and the consumer is performed by voltage regulators. Linear regulators have the advantage of providing the clean, stable power required for analog/RF components. However, apart from the lower conversion efficiency, they incur a voltage drop and dissipate more heat. Switching regulators characterized by considerably higher efficiency and are widely applied for digital subsystems which are more tolerant to noise.

The maximum power point tracking (MPPT) refers to power drawing from the energy

harvesting source at the level maximizing the power output. For DC sources like, for example, solar panels, the maximum power point (MPP) is a voltage-current combination which ensures the power output maximization under certain sunlight and temperature conditions. To achieve the power maximization, the supply and the load should be matched in terms of impedance [101].

Finally, we would like to note that following the energy neutral mode [92], i.e. consume only as much energy as harvested, will raise the opportunity of indefinitely long lifetime, limited only by the hardware longevity.

2.8 Energy Scavenging Platforms for WSN

In this section we review recent wireless sensor platforms with the energy scavenging technology application. Table 2.3 presents the overview of these platforms.

2.8.1 Everlast and ZebraNet Platforms

In this subsction we consider the sensor node platforms which are designed as a single module.

Everlast [101] has only a primary energy buffer, because its designers consider rechargeable batteries as lifetime limiting factor. Nevertheless, the system novelty concerns just to super capacitors which could be efficiently charged using a Pulse Frequency Modulated (PFM) converter and open-circuit solar voltage approach to track the maximum power point. Lifetime of Everlast is predicted as 20 years operation requiring no maintenance.

ZebraNet [116] is a GPS based WSN for animal migration tracking with sensors built in collars on neck of zebras. The wireless sensor node has the following HW features: single energy buffer as lithium-ion battery, dual-clock microcontroller, separate power supplies for each device to lower energy consumption, and off-chip memory since the system has to generate detailed logs. The middleware of the system has GPS sensing and radio communication as priority events, and the remaining events could be regarded as a combination of scheduled and unplanned events. To reduce power consumption developers carried out three low-level techniques: timely use of components, on-the-fly processing,

Platform	**Goal or Application**	**System Composition**	**Harvesting Technology**
Everlast [101]	Operating on supecaps, lifetime, low cost, high performance, durable WSN	Everlast node	Solar radiation
Heliomote [93]	Simplicity, ecosystem sensing	Mica2 [43] + heliomote board	Solar radiation
Prometheus [68]	Perpetual operation and prove of concepts	TelosB [43] + prometheus board	Solar radiation
Trio [52]	Sustainable, flexible, scalable WSN, experementation purpose	TelosB [43] + XSM [51] + prometheus platform	Solar radiation
ZebraNet [116]	Efficient power management, miniaturization, zebra‘s movement tracking	ZebraNet node	Solar radiation
VIBES [110]	Lifetime, acceleration data measuring	VIBES node	Vibration
PMG Perpetuum [15]	Battery-free sensor system, condition monitoring, industrial, aerospace	PMG sensor node	Vibration
PicoCube [38]	Small volume, tire presure monitoring	Five stacked boards	Vibration
AmbiMax [89]	Multiple harvesting components adoption	Eco node	Wind, solar radiation, thermal, vibration
Electro acoustic liner system [90]	The suppression of aircraft engine noise	Self-powered node	Acoustic noise
Cymbet CBC-EVAL-08 [44]	Quick development of energy harvesting applications, commercial sensor node	Cymbet CBC-EVAL-08 module	Solar radiation, AC-based energy

Table 2.3: Energy Scavenging Platforms

and dual clock scheme which assists to consume twice less energy using a slow clock. Besides, the ZebraNet node has evolved through different versions, but none of versions apply thought-out SW power management.

2.8.2 Heliomote, Prometheus, and Trio Platforms

In this subsection we rereview the sensor node platforms which contain several bords in the module, e.g. are composite constructions.

Prometheus [68] is a wireless sensor platform which includes the Prometheus power board and Telos. The entire system is being powered by the Prometheus power board. This power board is implemented with a two-stage storage system containing two supercapacitors as a primary buffer and a lithium rechargeable battery as a secondary buffer. The two supercapacitors are wired in series to decrease leakage of current. The lithium battery is charged only from the primary buffer where energy is stable and pulsing is possible. The SW of the Prometheus platform is pushed to the Telos MicroController Unit (MCU). Developers reduced the number of components on a board and thus decreased power consumption and board space having performed control logic by means of MCU. Developers of this system claim 43 years operation under 1% load, 4 years under 10% load, and 1 year under 100% load [68].

Prometheus, Telos [43] and XSM [51], in turn, are combined into the Trio [52] node. Telos is necessary for low power operation, Prometheus is responsible for solar power supply and whole charging system, and finally XSM is a set of indispensable sensors. Prometheus in this case is modified in order to enhance its performance and guarantee fail-safe operation. The system has a four-tier HW architecture: (1-st) mote tier performs sensing, local processing and communication, (2-nd) gateway tier forward traffic between server and mote tiers, (3-rd) server tier monitors and collects network statistics, multiplexes traffic from several gateways, and (4-th) client tier runs client applications. Each tier has special SW: the mote tier is controlled by system SW, besides application SW and middleware services manage mote, server, and client tiers.

Heliomote platform has solely NiMH rechargeable batteries as energy buffer. HW support for batteries long term operation have power routing switches that provide overcharge

and undercharge protection, and an on-board power monitor integrated circuit. Because of the inherent leakage and due to the increased overhead of energy storage management, developers renounced exploiting the super capacitors.

2.8.3 VIBES, PMG Perpetuum, and PicoCube Platforms

VIBES [110], PMG Perpetuum [15] and PicoCube [38] are microsystems powered from ambient vibrations. The intended application for the VIBES sensor node is an air compressor unit, though vibrations and frequencies measured pointed out an opportunity of different industrial applications. The microsystem, in general, contains three main units: the microgenerator, converting ambient vibration energy into electrical energy, the voltage multiplier, which converts and stores the energy in a super capacitor, and the microcontroller based subsystem consisting of an accelerometer and RF transmitter. VIBES is an energy aware system and has a possibility to adjust the duty cycle according to the available energy.

The second platform based on vibration energy-harvesting microgenerator is PMG by Perpetuum. PMG platforms by Perpetuum have three types of sensor nodes each developed for a particular application, namely PMG-17 is intended for control of plant machinery and continuous monitoring, PMG-27 enables aerospace industry, PMG-37 was developed for transportation applications. PMG platforms have a primary energy buffer for improved flexibility and do not require any maintenance. The PMG-17 energy harvester, for instance, can generate around 50 mW at 1g that enables to operate continuously. The technical parameters of PMG platforms open up new potentialities for the most data intensive applications. Depending on the application, the PGM node can be configured to transmit large data bursts rarely or transmit smaller data burst more frequently, for example, for data in a dynamic environment.

The PicoCube is a 1 cm^3 sensor node which consists of five stacked printed boards: radio, switch, sensor, MCU, and storage. However, the node does not contain a harvesting device. The harvested energy is being stored in a NiMH battery. In the future, the authors plan to exchange the packed battery with a battery manufactured by using direct write printing method [64].

2.8.4 Electroacoustic Liner System

Self-powered electroacoustic liner system [90] was designed specifically for the suppression of aircraft engine noise. The module consists of three subsystems: energy harvesting, communication, and impedance tuning. One of two electromechanical Helmholtz resonators (EMHR) is assigned for energy harvesting whereas the second one for noise supression. However, more EMHR can be used if more power is required to supply the system. The designed module has a secondary energy buffer as a rechargeable battery.

2.8.5 AmbiMax Platform and Platform by Cymbet

Two wireless sensor node platforms reviewed in this section support more than one energy harvester. However, their organization of energy buffers is various.

AmbiMax platform [89] enables the energy harvesting of solar, thermal, vibration, and wind. AmbiMax is comprised of energy harvesting system, reservoir capacity array, and control/charger. Each energy harvesting device collects ambient energy and replenishes its own reservoir capacitors. These different reservoir capacitors make up the whole reservoir capacity array. Furthermore, the platform has the secondary or in other words backup energy buffer as a rechargeable battery which is being used when the supercapacitors are discharged and there is a lack of ambient energy. To charge the supercapacitors at maximum efficiency the maximum power point tracking mechanism is applied. This method reveals the optimal power output at the given level of available ambient energy.

CBC-EVAL-08 node [44] is a demonstration kit designed by Cymbet Corporation. This node has a small solar panel on board as well as it enables the AC-based ambient signal harvesting. An external AC-based harvester can be connected to the node through the connector on the board. To keep the harvested energy, the module has two 50 uAh EnerChip batteries manufactured by application of the thin film technology [24]. In order to save as much energy as possible, the module supports SimpliciTI [20] the energy aware wireless network protocol. The power management block allows the preventing of deep battery discharge and the monitoring of high current operation. In spite of low battery capacity and in case of proper device use the CBC-EVAL-08 developers claim 10 years

lasting of the module operation.

2.9 Power Saving Techniques

The hardware solutions considered in the previous section can significantly increase a wireless sensor node lifetime. But non-intelligent hardware can not expand energy resources efficiently without appropriate power management techniques. In this section we consider the power saving techniques for WSN aimed to economical usage of energy resources and efficient management of sensor node hardware.

2.9.1 Adaptive Power Management in Energy Harvesting Systems

The authors of [85] propose an adaptive power management approach with the goal of decreasing the duty cycle when the scavenged energy is not enough for operating and vice versa to increase the duty cycle at that time when the scavenged energy is sufficient. The estimation task of the model proposed, based on previous measurements, predicts the scavenged energy available in the future, and a controller adjusts properties of the application. The feature of given approach is that energy is stored in a buffer before being utilized.

2.9.2 Adaptive Duty Cycling for Energy Harvesting Systems

In contrast to [85], [66] allows both direct and stored usage of the energy from harvesting source, but in both cases energy has to be measured. The proposed algorithm has three main goals, namely (a) load should not consume more energy than it is provided by the environment, (b) adjust to the dynamics of the energy source in the field (while [85] proposes parameterized specification and the calculation by online controller), (c) improve the system performance according to application utility model.

2.9.3 Energy Management Architecture for Wireless Sensor Networks

[85] and [66] are based on preliminary measurements and historical data to adapt the duty cycle according to efficient power management. The energy management architecture

(EMA) [69] for WSN, in general, is grounded upon prioritized enforcement of policy theory at the time of operation. It means that EMA at first performs a prioritized task, and so on in decreasing order.

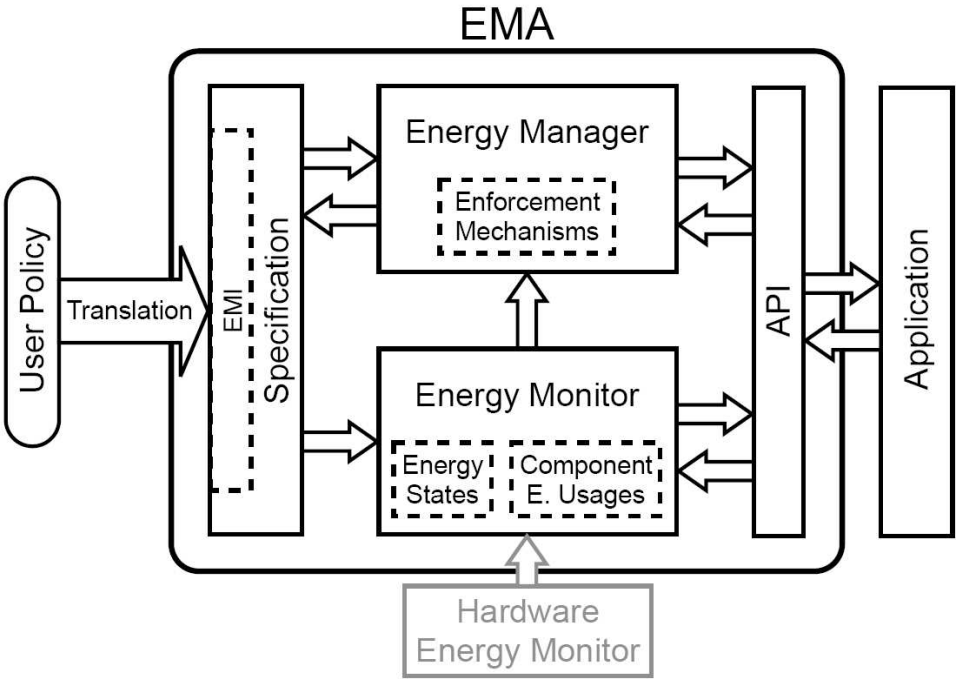

Figure 2.2: The Energy Management Architecture [69].

EMA presented in [69] considers energy as a crucial component and contains three basic components: a specification component, an energy monitor, an energy manager. The goal of the specification component lies in providing a set of interfaces and connecting the energy policy to the monitor and the manager. The monitoring component, in turn, supervises granular component energy usages according to a set of interfaces to the application. Besides, it observes the overall system energy such as remainder battery energy and incoming collected energy. The management component gets energy information from the monitoring component in order to oblige user policies to be transferred by the specification component and executes admission control on activities by means of interfaces to the application [69].

2.9.4 High-level Model for Estimating Power Consumption of Bluetooth Devices

The authors of [82] offer a novel technique to estimate the power consumption of Bluetooth modules. The estimation technique is based on a measurementbased approach devised to assure a suitable trade-off between estimation time, accuracy and number of prior measurements. The trends in this area of research will lead to the opportunity of power consumption estimation for each module of the embedded system. However, we can predict power consumption from the battery point of view. The battery model in [39] takes into consideration all dynamics characteristics of the battery with an opportunity to predict lifetime. The decision of the power consumption estimation of Bluetooth modules is valuable for circuit designers since it helps to extend battery runtime and minimize power dissipation. We will use this technique to improve power management optimization.

2.10 Synchronization in WSN

Time synchronization is essential in WSNs to coordinate activities performed by multiple nodes: multisensor data fusion, time-scheduling operations, and power-efficient duty cycles. However, time synchronization in WSNs is a specific field differing from the synchronization in wired networks [72, 96], since it requires the consumption of limited resource - energy. Moreover, radio chip is the most power-hungry component of sensor node [43]. Hence, it is a challenging task to effectively maintain WSN nodes synchronized while minimizing power consumption.

In this section we describe several synchronization techniques which aim at synchronizing the network while focusing to a variable extent on such competing characteristics as accuracy, power efficiency, reliability, and simplicity.

2.10.1 Approaches to WSN synchronization

There are several features in which synchronization protocols differ from one to another to provide best-fit solution for particular cases [96]. The synchronization is said to be *external* when all network nodes synchronize themselves to one external source of time

(e.g. UTC); or it is referred to as *internal* when only relative time differences between network nodes are compensated within given uncertainty boundaries (any node might be used as time reference). For this purpose, a group of network nodes can use one particular time value received from some sender (*sender-receiver* approach); conversely, those devices can register the arrival time of one radio message and then exchange these timestamps between them to know the time of each other (*receiver-receiver* approach). *Clock correction* is used when each node adjusts its clock; *clock transformation* instead is used when every node stores differences between its own clock reading and the clock readings of other network nodes to estimate their time. The synchronization can be *periodic* (i.e. performed repeatedly after some period) or it can be *on-demand* when network nodes are synchronized peradically, e.g. after some particular event.

2.10.2 Synchronization Uncertainty Sources

In order to be synchronized, network nodes must exchange some information related to time. The way in which that information is collected and it is transferred to a node is subject to several time uncertainty contributions which cause synchronization inaccuracy. Uncertainty sources are usually divided into two main groups: clock-related contributions and communication delay contributions [96, 74]. Clock-related uncertainty is due to the crystal frequency offset, changing environmental conditions and aging of the crystal. Communication-related uncertainty is caused by non-deterministic latencies appearing when synchronization messages travel from node to node in a network. Synchronization protocols can use some uncertainty models to minimize the synchronization inaccuracy. Synchronization activities defined by a certain protocol depend on the uncertainty notion accepted in this protocol. For example, a synchronization algorithm may perform the estimation and compensation of communication delays without addressing the clock skew [57, 95, 45]. The uncertainty model of such a protocol comprises only communication-related contributions.

2.10.3 The Time-Stamp Synchronization protocol

The Time-Stamp Synchronization (TSS) protocol, proposed by Romer [95], supports internal synchronization on demand. According to this protocol, some network nodes can generate a timestamp and transfer it over a multi-hop path. Any receiver of the timestamp-message can transform the received value to its own time-scale and retransmit it. The transformation is done by computing the time elapsed between the timestamp generation and its arrival at the receiver. Then, this time is subtracted from the arrival time of the message registered by the receiver's clock. The protocol enables the synchronization of multi-hop networks of highly dynamic topology at the expense of low overhead. However, the synchronization uncertainty tends to increase with the number of hops.

2.10.4 The Reference-Broadcast Synchronization protocol

The Reference-Broadcast Synchronization (RBS) protocol, invented by Elson et al. [53], is based on the receiver-receiver approach: a node sends several reference messages to its one-hop neighbors which then exchange arrival times of messages to compare their clocks. In order to be synchronized, receivers can compute relative clock offsets and skews by means of least-square linear regression. The obtained values are used to perform time transformation. Multi-hop networks can be divided into one-hop clusters with mutual gateway nodes that can transform time values of one cluster to those of another. The protocol provides low energy consumption (for small WSNs), high accuracy synchronization, and it eliminates the influence of non-deterministic communication latencies. However, RBS is affected by high communication overhead.

2.10.5 The Timing-sync Protocol for Sensor Networks

The Timing-sync Protocol for Sensor Networks (TPSN), devised by Ganeriwal et al. [57], defines the construction of a spanning tree, whose root is elected as a synchronization master. The master broadcasts a request message to initiate synchronization. Each node synchronizes to its "parent" in the tree topology by means of two message exchange (round-trip estimation of clock offsets and communication delays). If it happens that the

network topology changes or some node fails, the tree construction is repeated. The protocol succeeds in high synchronization accuracy under relatively low overhead. However, the hierarchical tree construction may introduce a significant overhead in networks with rapidly changing topology. The less a network the less energy-efficient the protocol is.

2.10.6 The Flooding Time Synchronization Protocol

The Flooding Time Synchronization Protocol (FTSP), designed by Maroti et al. [84], induces the flooding of the whole network with synchronization messages containing timestamps periodically generated by a single dynamically re-elected leader-node, and retransmitted by other nodes. Every node collects several leader-timestamps and arrival-timestamps of the corresponding synchronization messages. Then nodes compute their clock offsets and skews relative to the leader. Time-stamping is performed at the MAC-layer when a radio-message is sent or received. The FTSP implementation achieves a significant accuracy and it is robust, i.e. resistant to heavy changes of the network topology. However, this protocol causes a long convergence time to synchronize the whole network.

Chapter 3

A Methodology for Power Consumption Evaluation in Wireless Sensor Networks

The design of WSN systems faces challenges which are specific to its application domain. For instance, sensor nodes are commonly resource-constrained, battery-powered embedded devices. Replacing the batteries is, however, at best inconvenient, and sometimes downright impossible after the network deployment. Therefore, the efficiency of using the battery power determines the lifetime of the nodes and of the entire network. Due to these limitations, energy consumption is one of the most constraining requirements for design and implementation of wireless sensor networks. Thus, power analysis is a critical step in the sensor network development process. The choice of the application algorithms, operating systems, scheduling policies and program style may have a considerable impact on energy consumption. Power analysis provides a study of critical operations on limited power sources. It gives the designers a clear idea of how to adapt their applications in order to prolong the system lifetime. Reliable and accurate energy evaluation can be achieved by capturing all relevant low-level details and operating states of the studied HW platform. Rough approximation and high abstraction HW modeling fail due to the lack of such information. However, HW modeling should be lightweight enough and should

act on the proper abstraction level in order to provide scalable simulation for numerous sensor nodes.

In this chapter we present a methodology for power consumption evaluation of the individual nodes of a wireless sensor network. Our methodology supports the Platform Based Design (PBD) paradigm [35, 100], providing power analysis for various sensor platforms by defining separate abstraction layers for application, services, hardware and power supply modules. It is implemented as a SystemC-based framework that combines the event-driven simulation engine and HW model composer, and allows a user to describe the application using a set of service calls and user functions described in C/C++ language. The presented methodology allows one to estimate the lifetime of a node composed of three main parts: transmission module, computational module and a battery.

This research was carried out in collaboration with Ivan Minakov, Alena Simalatsar, Giorgio Fontana, and Prof. Roberto Passerone (University of Trento, Italy). My contribution to this work was focused on battery model implementation, its validation, and the entire system simulation.

3.1 Methodology

Our methodology is based on the platform based design (PBD) paradigm [35, 100]. In our methodology we distinguish four separate layers (platforms) as depicted on Figure 3.1. We start out description from the *platform layer*, where we represent the node HW platform as the composition of several HW components where the CPU plays the role of a central component that manages the access to the peripheral components such as: transceiver, ADC, several timers, a flash memory and an IRQ controller. Each of the components is defined by means of a finite state machine (FSM), where the number of states of each particular machine is limited by the number of possible states (e.g., running, sleeping, hibernate for the CPU) with numerical parameters (e.g., current and time) determining the power consumed by this hardware component in a particular state. The FSMs of the peripheral components are synchronized only with the FSM of the CPU by means of a bidirectional channel. Our framework allows the definition of several standard WSN

platforms composed of a number of hardware components that build the *platform layer*. The choice of a HW platform is left to the user to be defined by choosing a specific platform library that include the energy and timing parameters for operations performed by HW components.

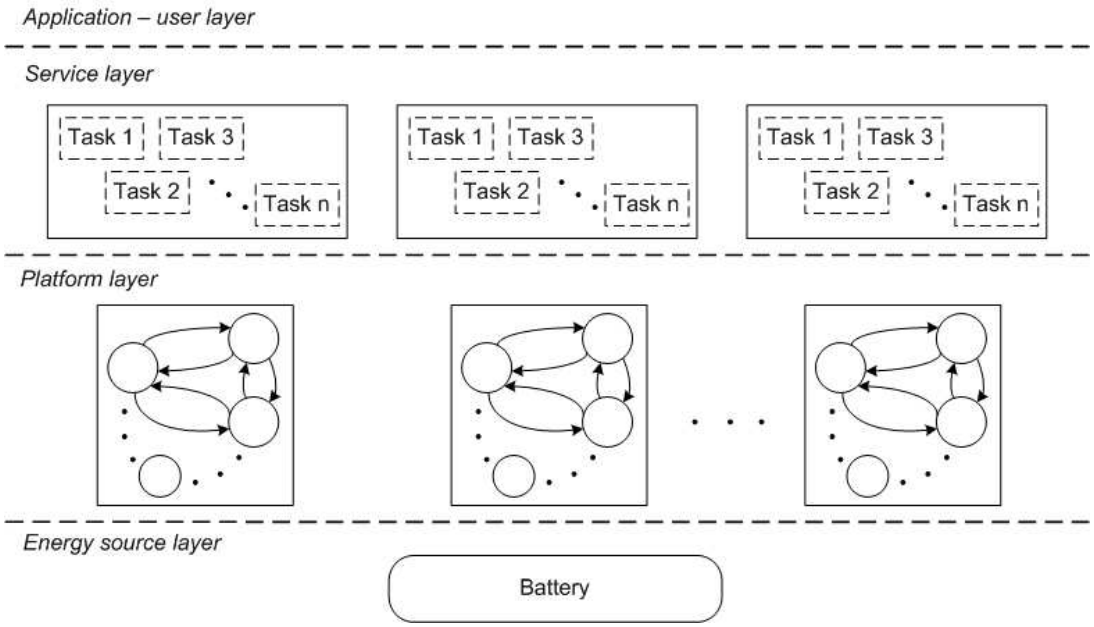

Figure 3.1: Methodology.

The upper layer, called *application layer*, is used to present an application through a set of semantic primitives. This is an open layer, where the application description is left to the user. Each application is presented as a set of services calls and user functions described in C/C++ language. At this layer user interacts with hardware by means of services provided by the *services layer* in a purely functional way.

The *services layer* acts as an application interface that defines a set of services available to the end user to specify an application. This layer abstracts details of the presented HW and provides HW independent API for the application needs. The services themselves are commands to the HW modules to perform particular action. Conceptually they are similar to HW independent drivers. By separating the platform services from the actual HWplatform we introduce the separation between the system behavior and its cost/performance thus introducing the flexibility to the design space exploration.

The total set of services provided by the *services layer* is composed of a number of subsets, where each subset is dedicated to a particular hardware component, and represents the maximum possible set of services that may be provided by all possible platform vendors. The choice of a particular platform defines the subset of the total set of services by making them available at the *application layer*. However, the number of provided services by a hardware component may vary from one supplier to another, which introduces constraints to the application interpretation. Thus, at the application layer validation of the tasks composition according to the chosen platform is taking place. For example, if application requires a service that is not specified for a particular platform, the user will see a compilation error. Thus the user needs to be aware of the set of available services.

As an example here we present a set of basic services provided by a Timer component:

```
void RunTimer( void );
void RunTimerPeriodic( void );
void StopTimer( void );
void SetTimerUPeriod( u_time );
void SetTimerMPeriod( u_time );
void SetTimerParam( uint );
void EnableIrq( bool );
```

The execution of a service triggers the transitions of the FSMs representing the dedicated hardware component, thus changing the power state of this component.

Besides the components services, there exit some global services to specify user defined functions as interrupt service routines (ISRs) for every component that may generate an interrupt (IRQ) event. Inside the ISR the user can define an arbitrary functionality (e.g., data processing, MAC layer etc.). Currently, we abstract this functionality thus it has to be annotated with equivalent time needed to execute it on the target HW component.

We assume that all the IRQ handlers are atomic and can not be preempted by other ISRs and that the power consumed to perform the context switch operation is negligibly small in comparison to the total power consumed by the interrupt handlers execution. In addition the shift of the execution time frame of an IRQ handler withing one operation

cycle has no influence to the total power consumed during this cycle.

In addition to IRQ handlers the CPU may execute some background computations represented as a main CPU thread that have no real-time requirements (e.g., light up a LED). The execution of this main thread will be suspended by any IRQ coming from peripheral components until the IRQ event handler will complete its execution. In our case study the main thread includes only one command that sets the CPU to the *idle* mode.

As long as we talk about the power consumption, we introduce an *energy source layer*, which is currently represented with a battery. Each hardware component, being in a particular state will consume certain amount of power, which is accounted by the battery model. This layer may also include harvesting components or other sources of energy, which opens possibility to analyze power management techniques.

3.2 Case Study

In this section we describe the case study simulation of an abstract WSN application mapped to a model of TelosB platform. In particular we have studied the influence of parameters of the real device and application behavior on the power consumption.

3.2.1 The Application

A typical application for WSN can be divided into several standard functional stages in which it firstly gather some controlled quantity while sensing the environment, process this data and form the packets that are then sent to a base station. The controlling algorithm case study presented in this paper follows this standard scenario. It performs the periodical sensing of the environmental temperature, with the period equal to one second, where an on-board ADC is used to convert the analog value into its digital interpretation. When the converted data are ready the application turns on the transceiver (*radio*), composes the network packet, waits till the *radio* is in the transmission mode and sends the data to an abstract base station. After the data transmission is over, the *radio* is switched to the Power Down state where it stays until it is reawaken in the following

period triggered by means of a timer. The example of the interrupt handler of the timer used to trigger the ADC conversion is presented here:

```
void _interrupt Timer_ICR(void* ptr){
      SysTop* pSys = (SysTop*)ptr;
      pSys->Adc.RunConvertion();
}
```

The pSys represents a particular HW platform that includes the ADC component providing the RunConvertion() service.

The communication algorithm (a network protocol) is abstracted from the real MAC and routing protocol implementation by annotating the functionality of a network stack with time needed to performabstract operations with data packets. Basically we focus on the HW state transitions during the execution process. The current consumption waveforms we obtained are presented on Figure 3.2 and Figure 3.3, where Figure 3.2 depicts the sequence of input pulse while Figure 3.3 presents a zoomed in single pulse of this sequence.

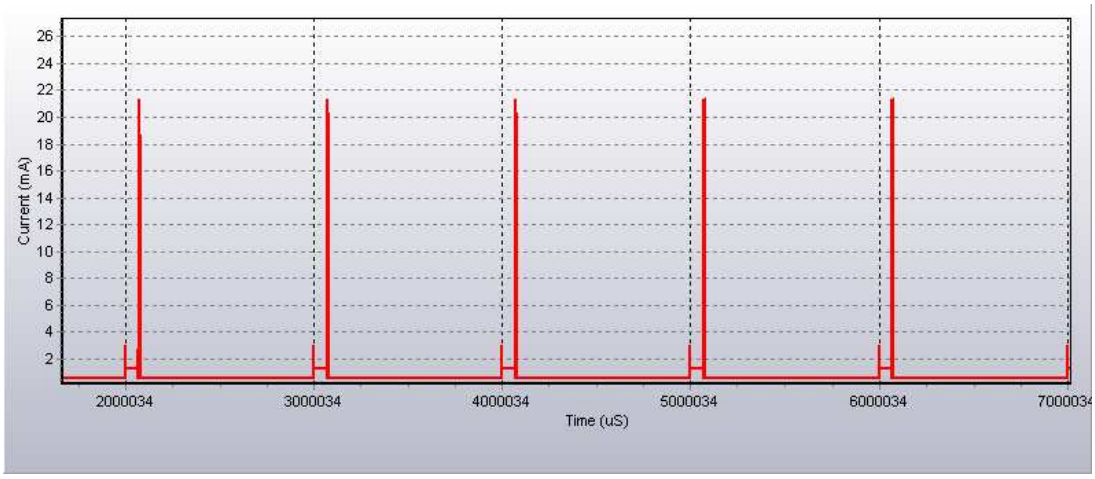

Figure 3.2: Input pulses used to discharge the battery.

3.2.2 The Platform

As a target HW architecture case study that represents an entity of our platform layer we have chosen TelosB platform [43] due to its popularity in WSN systems design. The TelosB

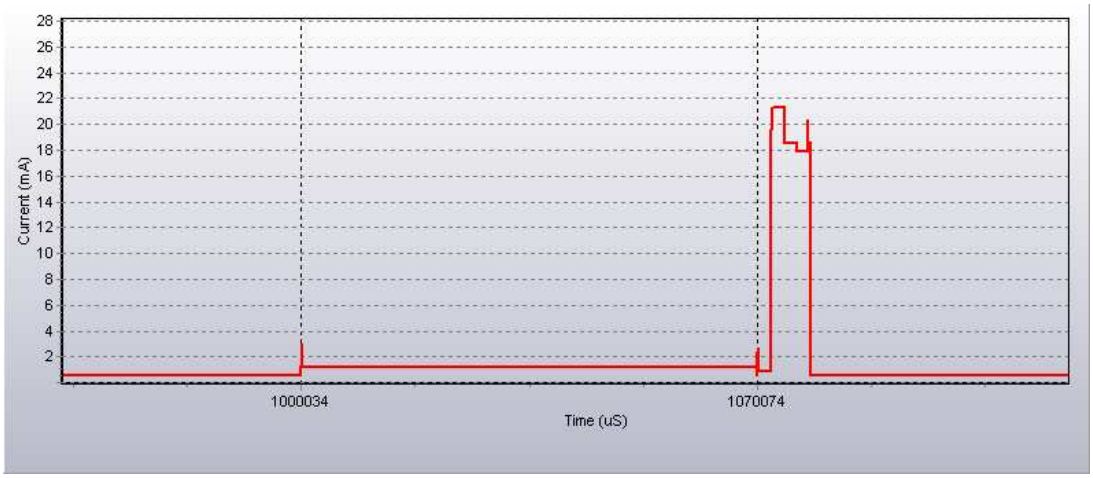

Figure 3.3: Example of a zoomed pulse.

mote is composed of the MSP430Fx1611 microcontroller, external flash memory chip, CC2420 transceiver, set of the sensors and Leds. It is powered by two AA batteries, which should provide voltage in range from 2.1 to 3.6V.

The HW platforms in our framework are defined using the preprocessing functions:

```
#define _TELOSB_MOTE_
```

The composition of the HW components, energy and timing parameters for HW operations are included in a platform library (telosb.lib). On the platform layer this library defines a set of the simulated HW components and provides platform specific information to the generic HW modules. The energy and timing parameters that we used in the simulation were partially received out of direct measurements and the rest was taken from the components specifications of the TelosB mote. The table 3.1 presents possible energy conditions of the TelosB mote.

The SystemC module of the TelosB mote is introduced as a SystemC CPU module that coordinates the work of the peripheral component, such as an ADC, a transceiver, timers and an interrupt controller included and initialized inside the platform SystemC module:

```
                ...
SysTop(sc_module_name _name ): GCpu( _name ),
                               Adc("Adc"),
                               Radio("CC2420"),
```

States	**Current consumed**	**CPU**	
Radio		*Active (8MHz)*	2 mA
Voltage regulator off (OFF)	0.02 uA	*Active (1MHz)*	0.6 mA
Power Down mode (PD)	20 uA	*ADC-12 conversion*	500 uA
Idle mode (IDLE)	400 uA	*SPI transmission*	700 uA
RX	18.8 mA	*LPM0*	300 uA
TX(-25 dBm)	8.5 mA	*LPM1*	80 uA
TX(-15 dBm)	9.9 mA	*LPM2*	20 uA
TX(-10 dBm)	11 mA	*LPM3*	10 uA
TX(-5 dBm)	14 mA	**LED (*each*)**	
TX(0 dBm)	17.4 mA	*On*	2.1 mA

Table 3.1: Energy conditions of the Telosb mote

```
                Timer_1("Timer_A"),
                Timer_2("Timer_B"),
                IRQ_manager("IRQ")
        ...
```

When performing the coordination the CPU is operating in the *active* operating state thus consuming more power comparing to the *idle* mode, when no interrupts need to be handled. Most of its time CPU spends in the power safe idle (LPM0) mode. Thus only about 1% of the CPU computational/time resources is utilized by the case study application. The rest of the time application operates with 2 timers (Timer A and Timer B), ADC (ADC 12), USART (SPI mode) controller and a radio transceiver (CC2420).

3.2.3 The Energy Source

The energy layer is represented by a battery model described in Section 3.3. This model is adjusted in accordance with 1.2 V type NiMH battery specification [10], which supports up to 1.6 V while recharging. Therefore, the battery model provides the system with 3.1 voltage level that is equal to power supply of two batteries connected in series of 100 mAh of total capacity.

Low capacity is chosen to decrease the simulation time of the battery chargedischarge process. The minimal voltage level required to supply TelosB platform is defined at 2.1 voltage level. Thus, in the power consumption estimation framework we apply the useful voltage range from 2.1 to 3.1V. The pulses generated by simulator are presented in Figures 3.2 and 3.3 are the input discharge pulses for the battery model

3.3 Spice Compatible NiMH Battery Model

In our work we have implemented the electrical battery model proposed by Chen [39] using PSpice [88]. Due to the absence of controllable components in the standard PSpice library, we started the model development from controllable resistor and capacitor modeling, so that their characteristics could be changed in accordance with seven behavior equations derived by Chen et al. [39]

The Spice model of the controllable resistor(see Figure 3.4) is designed in accordance with the derivative of *Ohm's law* ($V = IR$). This component has three pins, i.e., *controllable*, *input* and *output*. The *multiplier* component performs the multiplication of resistance provided by the *controllable* pin, and current from current-controlled voltage source of the *input* pin. The desirable voltage appears at the *output* pin of the controllable resistor.

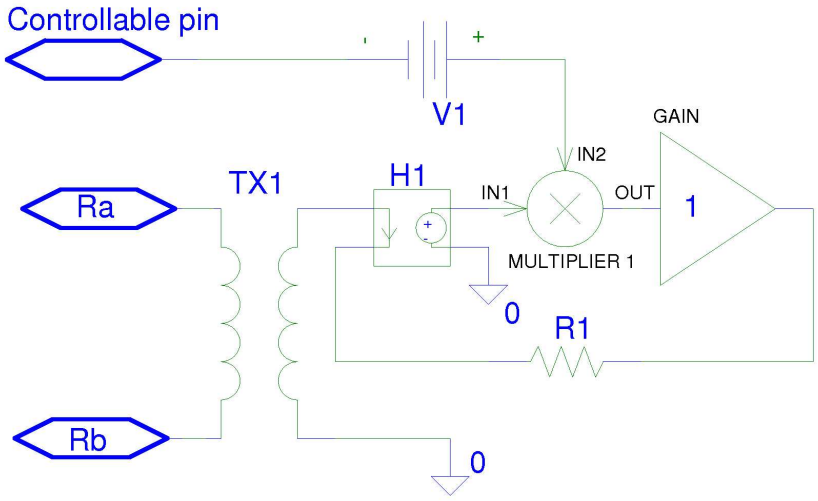

Figure 3.4: Controllable resistor.

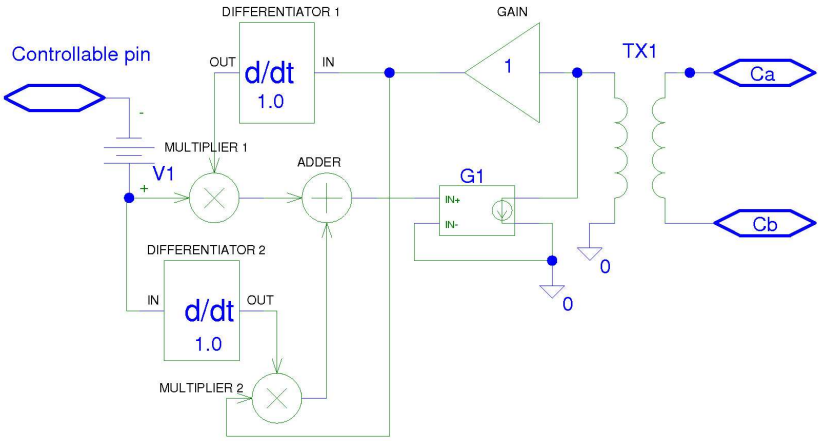

Figure 3.5: Controllable capacitor.

The controllable capacitor (see Figure 3.5) component contains the same three pins as the controllable resistor and is designed on the basis of the following equation:

$$i = C \cdot dV/dT + dC/dT \cdot V \tag{3.1}$$

It contains two terms: the product of capacitance with the voltage derivative, and the product of voltage with the capacitance derivative. To get the first term the model multiplies the capacitance from the *controllable* pin and differentiates the voltage from the voltage-controlled current source which is connected directly to the *input* pin. The second term is derived by multiplying the voltage from the *input* pin with the differentiated capacitance from the *controllable* pin. Finally, the *adder* component accumulates both terms. In order to save space and avoid the undesirable interconnections during the model debugging, we created the sub circuits for controllable components.

In addition, in order to connect both components to nodes with a voltage other than ground, we applied a linear transformer with high inductance in parallel to the *input* and *output* pins of the controllable components.

To ensure that the controllable models of the resistor and the capacitor function properly, we simulated both using Spice. Next, we compared the results with a conventional resistor and capacitor respectively. The simulation curves of the controllable and conventional components are equal.

The PSpice battery model is depicted in Figure 3.6. The designations of the components

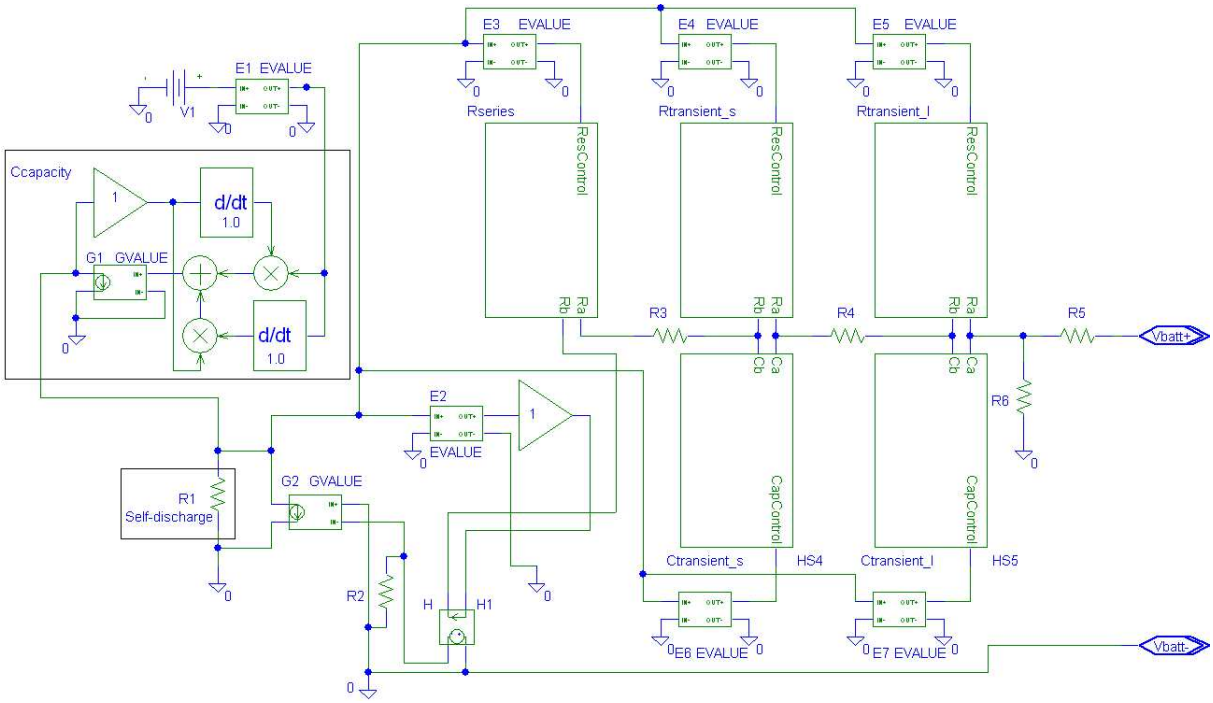

Figure 3.6: Spice model of a NiMH battery.

and their interconnections are performed in accordance with the electrical battery model [39].

Pins V_{batt+} and V_{batt-} are positive and negative poles of the implemented PSpice model. Pins a-b in the created sub circuits are standard input-output pins for passive components. The *Control* pin establishes the controllable components behavior specified by behavior equations [39] for each component. *Value form* (EVALUE) components E1, E3-E7 determine the value changing of controllable components. into $C_{capacity}$ (battery charge), R_{series} (instantaneous voltage drop of the step response), $R_{transient\_s}$, $R_{transient\_l}$, $C_{transient\_s}$ (short-time constants of the step response), $C_{transient\_l}$ (long-time constants of the step response) models correspondingly. The values of these components are changing according to seven behavior equations derived by Chen et al. [39]. Self-discharge resistor R1 represents self-discharge loss of power when battery is kept for a long time.

3.4 Battery Model Verification

To validate the NiMH battery model developed in PSpice we have carried out applying constant current charge and discharge, and two various pulse discharge experiments both for the model

and a real battery. In our experiment, we used a 200-mAh, 8.4 V NiMH Coop rechargeable battery (9 V prismatic type). This battery type has the cut-off discharge voltage at 7.0 V level (the gap is 1.4 V) that is more convenient for battery model verification in comparison to 1.0 V cut-off discharge level of 1.2 V battery type (the gap is 0.2 V). The first experiment aimed to discharge the battery with a constant 23 mA current which is the typical current consumption of TelosB sensor nodes with the radio turned on. The end-of-charge battery voltage is 7 V. To record the battery voltage we used (in all cases) an Agilent 34411 digital multimeter. The discharge profile of the real battery and of battery model is depicted in Figure 3.7. The maximum voltage and runtime error for each charge/discharge profile are shown in Table 3.2.

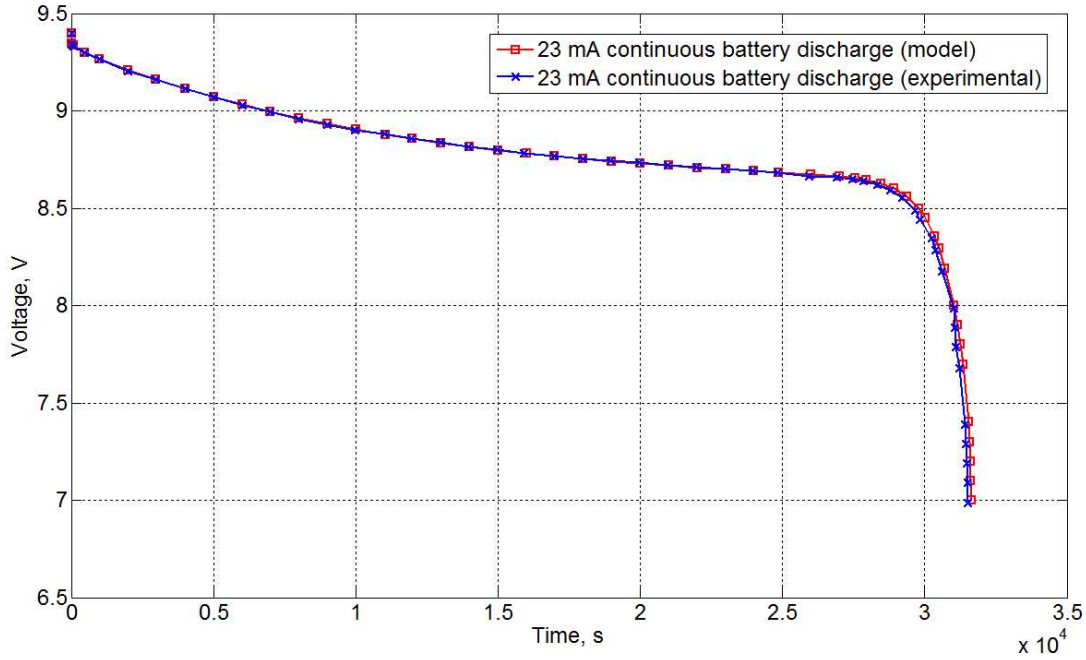

Figure 3.7: Battery discharging with continuous 23 mA current.

The second case is a continuous battery charging with 0.1C current, where C is a nominal battery capacity (see Figure 3.8). Agilent E3631A DC power supply was applied to generate the charging current.

The last two cases refer to battery discharging with 0.1C and 0.2C current pulses (see Figure 3.9 and 3.10 respectively). 0.1C current pulses have a 1000 s pulse width for a 1200 s pulse period. 0.2C current pulses, in turn, have a 2000 s pulse width for a 2500 s pulse period.

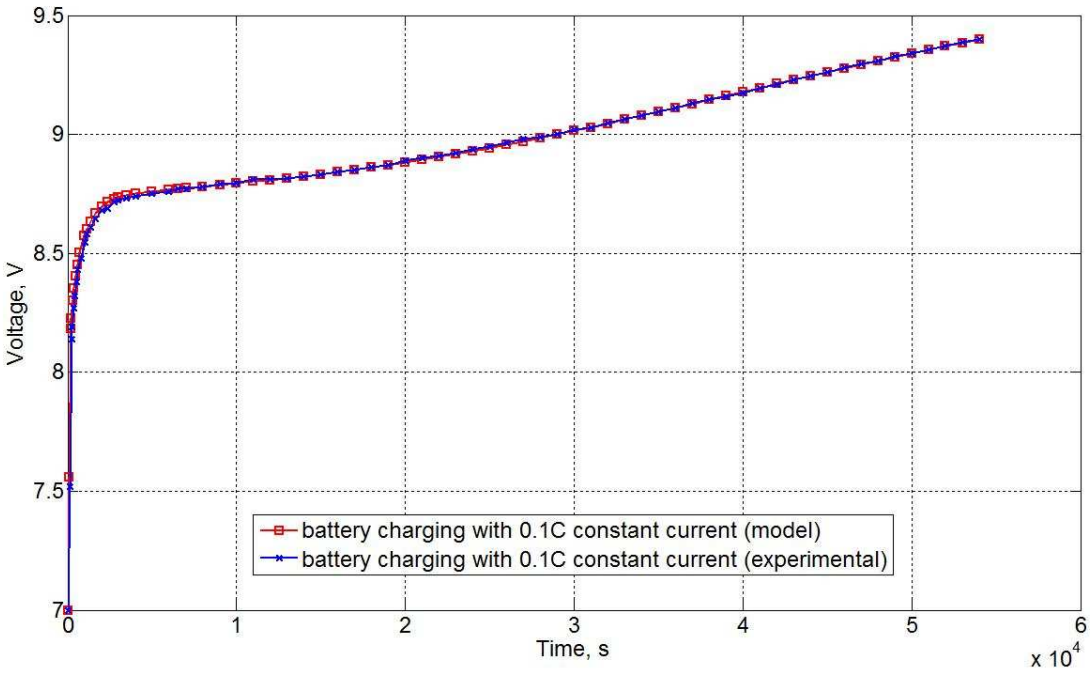

Figure 3.8: Continuous battery charging (0.1C).

The pulses were generated by an HP 33220A programmable pulse generator which handled the Schrack RA 200006 relay to close and break the loop.

The implemented PSpice battery model on the basis of electrical model proposed in [39] demonstrates accurate voltage response and runtime prediction.

Comparison type	**Maximum error in voltage, mV**	**Runtime error, %**
0.1C continuous charging	43	0.278
23 mA continuous discharging	34	0.099
0.1C pulse discharging	24	0.352
0.2C pulse discharging	26	0.112

Table 3.2: Battery model validation results

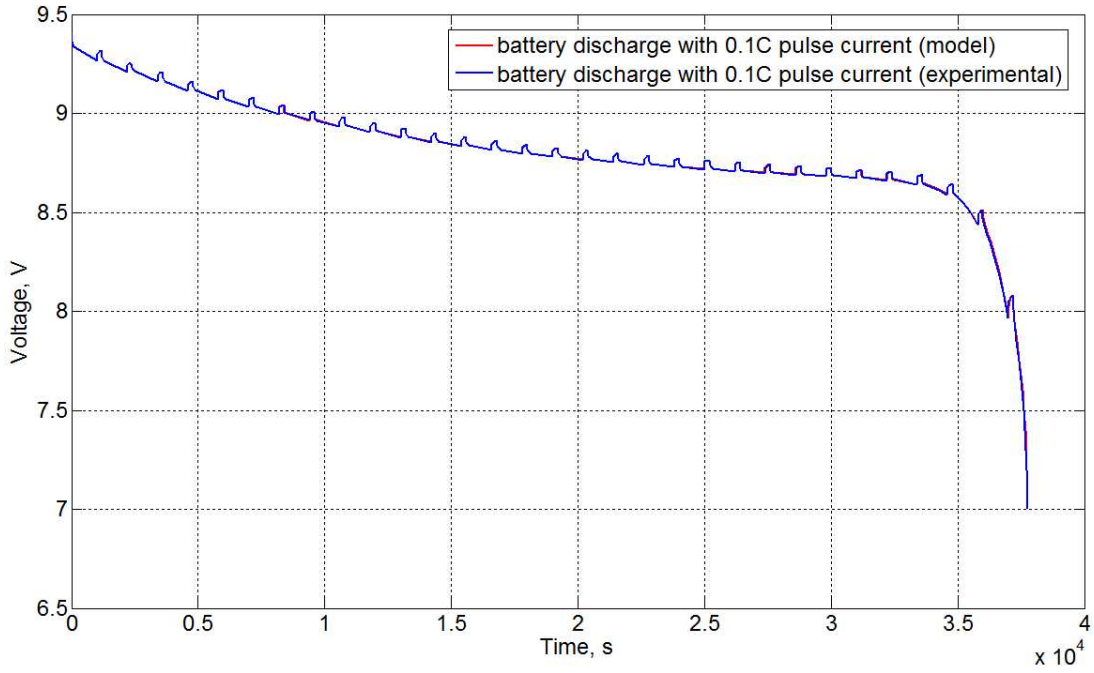

Figure 3.9: Pulse battery discharging (0.1C).

3.5 Experimental Resullts

Figure 3.11 presents the graph that is composed of two curves, one dedicated to the battery charge period (during 80000s up to 3.1V) and the other one to its discharge (during 93962s down to 2.1 voltage level) using pulses generated by the SystemC framework simulating the work of a single node with the 100% duty cycle. The battery discharging is quite smooth at 3.1 to 2.4 voltage range. However, starting from 2.4 V the discharging curve slumps quickly.

To carry out the energy consumption experiments we were charging the battery model during 80000 seconds up to 3.1 V (we imply two cylindrical type NiMH batteries connected in series). The battery model capacity was defined as 330mAh. The input discharge pulses for the battery were generated by sensor node state simulator presented in this paper. The five discharge profiles were applied in order to estimate the system life time: 100%, 50%, 50% with doubled discharge and idle pulses, 10% and 1% duty cycles. Table 3.3 presents the battery run time under specified duty cycle.

It is obvious that 1% duty cycle is the most economical mode for system use. Moreover, in

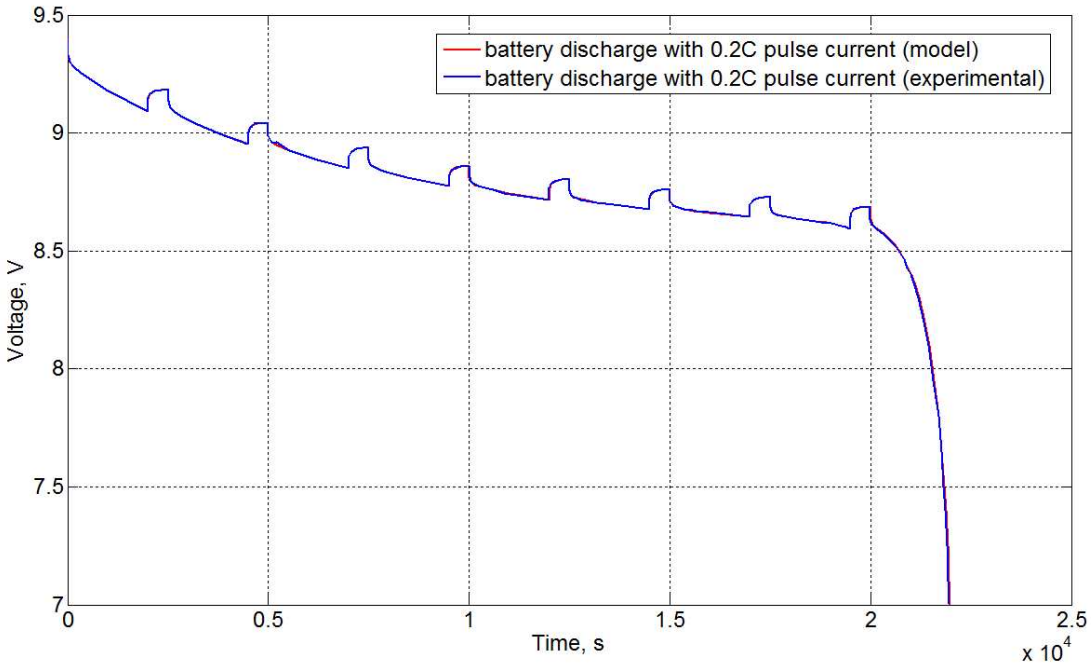

Figure 3.10: Pulse battery discharging (0.2C).

comparison to both of 50% and 10% duty cycle experiments this mode is more energy efficient in terms of utilization of battery charge unit per second. Besides, two various experiments with 50% battery discharge reveal that short time pulses lead to the battery life time increase. This fact can be explained by the relaxation effect of the battery. The experimental results presented in Table 3.3 give a general idea how the discharge pulses rate has an influence on the system long-term operation.

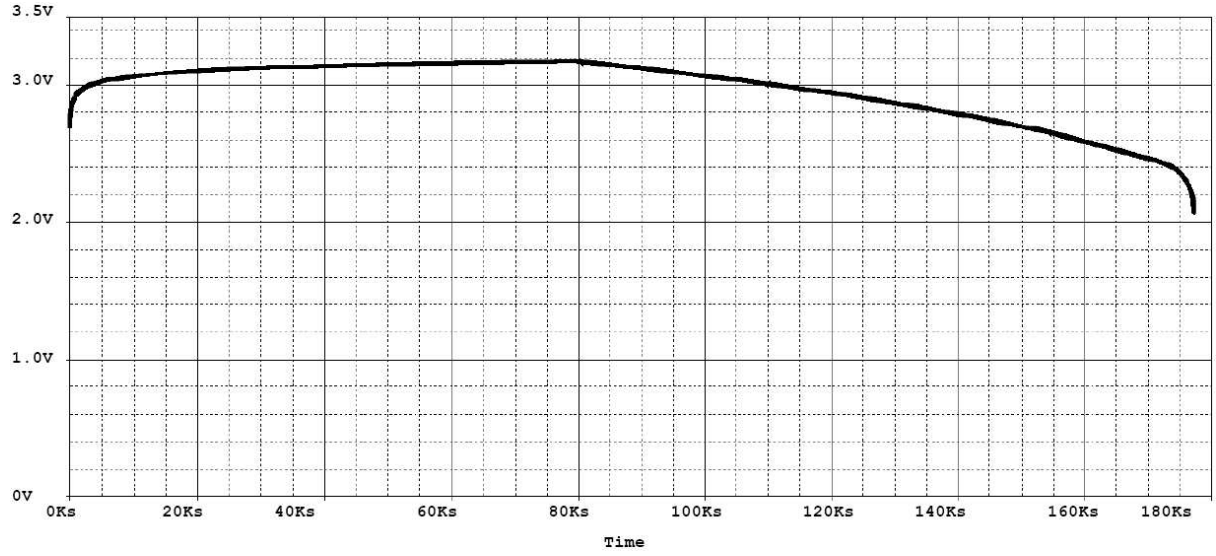

Figure 3.11: 100% duty cycle.

Duty cycle	**Run time (at 2.1 V)**
100%	177962 s
50%	347945 s
50% doubled*	346602 s
10%	1744743 s
1%	17449958 s
*time interval and pulse time are doubled relatively *50% experiment*	

Table 3.3: Battery run time (simulation results)

Chapter 4

An Energy Scavenging Module

In this chapter we present two Energy Scavenging Modules (ESM). The EnTick is the ESM for WSN, designed to interact uniformly with several scavenging components. This ESM can harvest both AC and DC environmental energy. The scavenged energy is stored in two energy buffers. The primary energy buffer is composed of two super capacitors wired in series; the secondary buffer is a rechargeable Li-ion battery. For the design of the module, we take into account the specifications of the hardware components, which have their own lifetime or number of possible charge-discharge cycles. To increase the lifetime of the system, we have developed an algorithm for efficient use of a harvesting component and the two energy buffers, by selecting the appropriate source according to the system and environment conditions.

The second module called "TreBer" has a different primary buffer: instead of packed off-the-shelf electrochemical capacitors it applies a supercapacitor fabricated using "direct write" method described is Section 2.6.6.

4.1 System Concept

The goal of the ESM is to mediate between a harvesting component and a sensor node, providing a flexible modular system, and to increase a sensor node lifetime. Both AC-based and DC-based harvesting components are supported, although the two cannot be present at the same time. The system uses two energy buffers, a super capacitor and a Li-ion rechargeable battery. The ESM applies an algorithm to choose the most appropriate energy source at any moment. The

algorithm is developed in accordance with the hardware components lifetime cycles and the availability of the necessary voltage level to supply the sensor node.

4.1.1 The Energy Scavenging Module Architecture

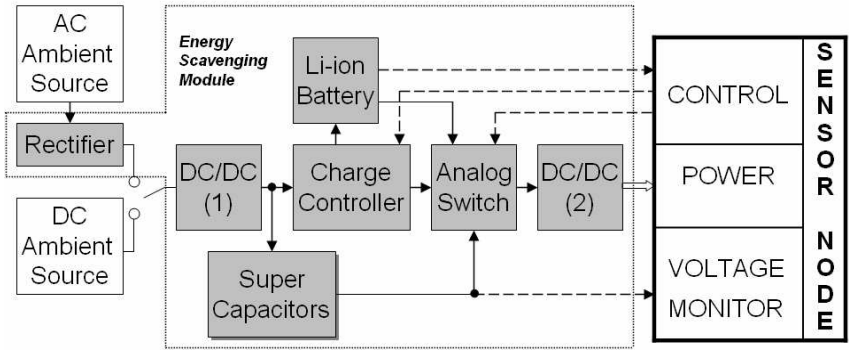

Figure 4.1: Architecture of energy scavenging module.

Figure 4.1 shows the hardware architecture of the Energy Scavenging Module (ESM). The ESM architecture contains six main blocks shadowed in grey. The first block DC/DC(1) operates as a transformer. It is supplied either by the AC-based ambient source (i.e., noise, vibrations) or the DC-based ambient source (i.e., solar radiation, thermal energy). The purpose of the DC/DC(1) block is to provide the ESM with stable voltage equal to 4.5V. Super capacitors, wired in series with each other and in parallel with the DC/DC(1) block, operate as a buffer. In other words, if ambient source is available the system exploits environmental energy transformed to the necessary voltage level by the DC/DC(1) block. Otherwise, the system can be supplied by super capacitors in case they are charged. The sensor node monitors the voltage level available in the super capacitors to ensure if they can be a power supply for the system or not. The third power supply depicted in the system architecture is a Li-ion battery. The Li-ion battery is a backup power supply. The charging/discharging states of the battery are controlled by the Charge Controller which, in turn, is managed by the sensor node. Thus, the ESM is designed in such a way that a sensor node can be supplied by one of three energy sources: super capacitors, Li-ion battery or directly by environmental energy source. However, a user must specify the type of ambient source applicable using "jumpers" before the sensor network deployment. Before the DC/DC(2) block generates the required 3V output to supply the sensor node, the Analog switch

chooses the power supply on the basis of the voltage level available and supply source priority (see Section 4.1.2). The Analog switch and the Charge Controller are managed by the onboard sensor node microcontroller.

4.1.2 The Algorithm for Efficient Power Supply Switching

In order to increase a Wireless Sensor Network (WSN) lifetime, the system adopts the following energy strategy:

- take what is available;

- use economically;

- use efficient conversion and optimal power harvesting for high-quality performance.

The Finite State Machine (FSM) depicted in Figure 4.2 explains how the ESM module handles the power sources to increase the long term operation of the entire network. As discussed,

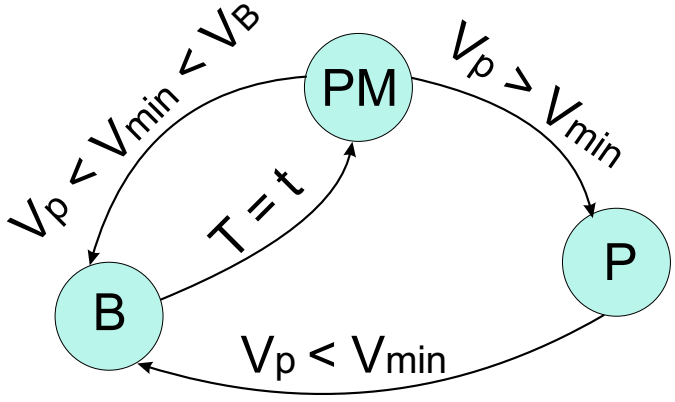

Figure 4.2: FSM explaining the choice of power supply for a sensor node.

there are three available sources of power: an environmental power source, the super capacitors and a Li-ion battery. Since the Li-ion battery has the shortest lifetime, which mostly depends on the number of charge-discharge cycles, its energy is the most valuable. The environmental power supply is inexhaustible, but unstable. Thus, a Power Management (PM) algorithm at first considers the ambient power source and super capacitors as a primary power supply P. The battery is defined as a backup or secondary power supply B. Initially, PM compares the

available voltage level of the primary power source Vp with the minimal voltage threshold $Vmin$ needed to supply the load. If $Vp \geq Vmin$, PM selects P as a power supply for the load. In case the P source becomes unavailable, the load is switched to the backup B. The algorithm then monitors the level Vp every T time units, and switches back to P when environmental energy is available again. Otherwise, B remains as a power supply for the load.

4.2 Implementation

The implemented four layers 50x70 mm prototyping board of the ESM is shown in Figure 4.3. The ESM supports a 5.5W ambient power source (input current = max. 500mA; voltage = max. 11V), and a "jumper" is used to switch between the AC and DC source. The AC-to-DC voltage conversion is performed by a standard DB102S diode rectifier and an electrolytic capacitor. Further stabilization of the DC voltage to 4.5V is implemented by MAX1672, a step-up/down DC/DC converter [6].

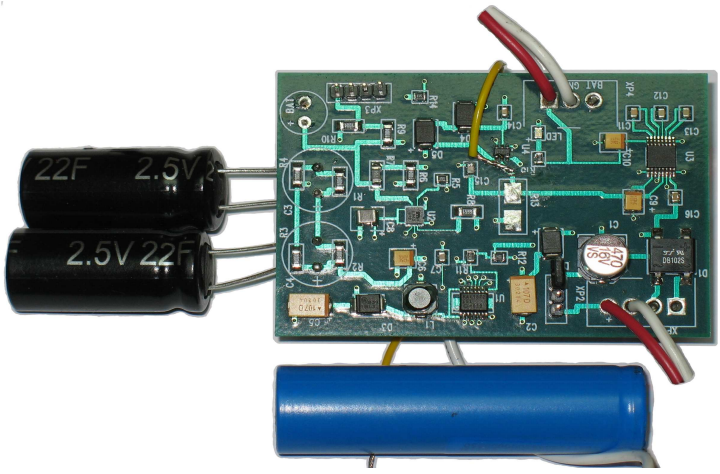

Figure 4.3: The energy scavenging module - EnTick platform.

Two super capacitors wired in series to reduce leakage current are used as the primary energy buffer. In line with [68], we have chosen Cooper Bussmanns [3] 2.5V, 22F capacitors as a trade-off in terms of capacitance and leakage current. The series wiring of super capacitors requires balancing to guarantee equal voltage sharing. We applied passive balancing to maintain similar voltages between the super capacitors where the leakage currents may be slightly different. Passive balancing uses equal value resistors in parallel with the super capacitors to let a small

current flow (of the order of 2-3 micro amperes that does not essentially affect the sensor node lifetime) between the super capacitors, thus keeping the voltages at the same level.

The secondary energy buffers functions are performed by a EEMB LIR14500 Li-ion rechargeable battery [4] having 3.7 nominal voltage and 750mAh typical capacity. Having no memory effect, lithium rechargeable batteries provide a high number of charge-discharge cycles (in comparison to NiMH batteries), highest density and lowest leakage [68]. However, they require a more complex charging circuit, e.g., a special charge controller. The MIC79110 battery charge controller [7] is to provide sufficient charging of the secondary energy buffer. This integrated circuit performs a number of functions needed for thorough battery maintenance, i.e. thermal shutdown, current limit and reverse current protection, precise voltage control. The secondary buffer is charged by the primary buffer. However, with the primary buffer being charged and the ambient power source providing stable voltage, the secondary buffer can be charged directly. The Single Pole Double Throw (SPDT) switch MAX4624 [6] having high current carrying capability and low power consumption is used to choose between the primary and secondary power buffer to supply the wireless sensor node. The self-powering modules output voltage of 3.0 volts is provided using the LM3352 [8] step-up/down DC/DC converter which provides voltage stabilization. In order to ensure interconnection between the ESM and the wireless sensor node, we have used a 4-pin connector for charge/monitor battery control, super capacitor voltage monitoring and analog switch control.

4.3 EnTick Evaluation

In this section, we describe the experimental setup and evaluate the designed system.

4.3.1 Experimental Setup

To evaluate the ESM, we used the experimental setup shown in Figure 4.4. This experimental setup allows us to conduct three experiments: (a) to charge the super capacitors from DCsource, i.e., solar panel BP SX305M, and then charge the battery using the same source (the super capacitors are a filter in this case), (b) to charge the super capacitors from ACsource, and (c) to discharge both the battery and super capacitors by connecting a sensor node.

Since we did not have a harvester for noise or vibration capture, we carried out the experiment

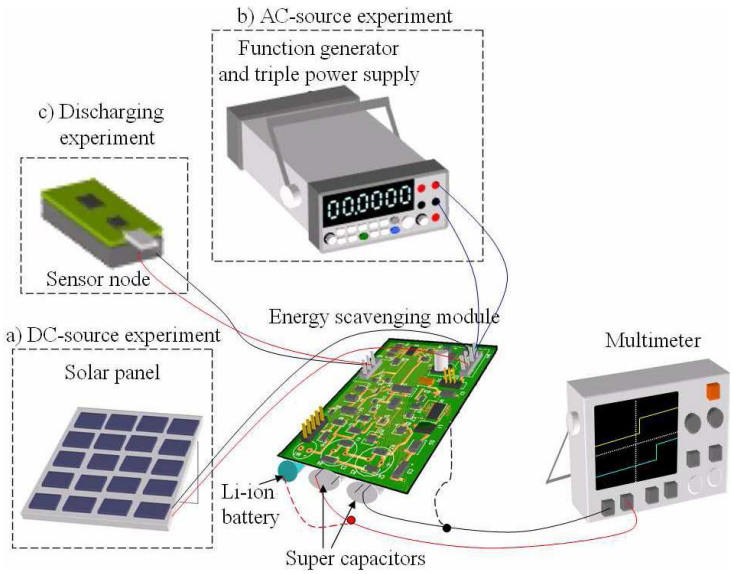

Figure 4.4: The experimental setup for the charging of supercapacitors from (a) DC-source, (b) AC-source.

having simulated noise with the triple power supply HAMEG HM8040 and function generator HM8030. The digital multimeter Agilent 34411A measured with a 2s sampling period the voltage level of the two super capacitors wired in series and recorded the data in a log-file.

4.3.2 Results and Analysis

Since it was cold but sunny weather, we installed the solar panel transversely to sunlight. Then we connected the solar panel wires to the ESM power connector. The multimeter was connected directly to the plus and minus pins of the super capacitors taking into consideration that they are wired in series. The "jumper" was installed in DC ambient source position. These experiments were carried out in the electronics lab at the University of Trento in February.

Figure 4.5 shows the voltage generated by the solar panel during the experiment. The purposes of this experiment are to demonstrate that the super capacitors could be promptly charged from the ambient source, and that they are able to support an uninterrupted operation of the sensor node in real conditions at least during the night. The voltage was stable at approximately 1.1 V. Two super capacitors were fully charged in 1464 seconds (25 minutes). Next, we connected a TelosB sensor node [43] to the ESM. The node (1.8 mA current draw in active mode) was programmed to measure humidity (3mW power consumption) and communicate (23

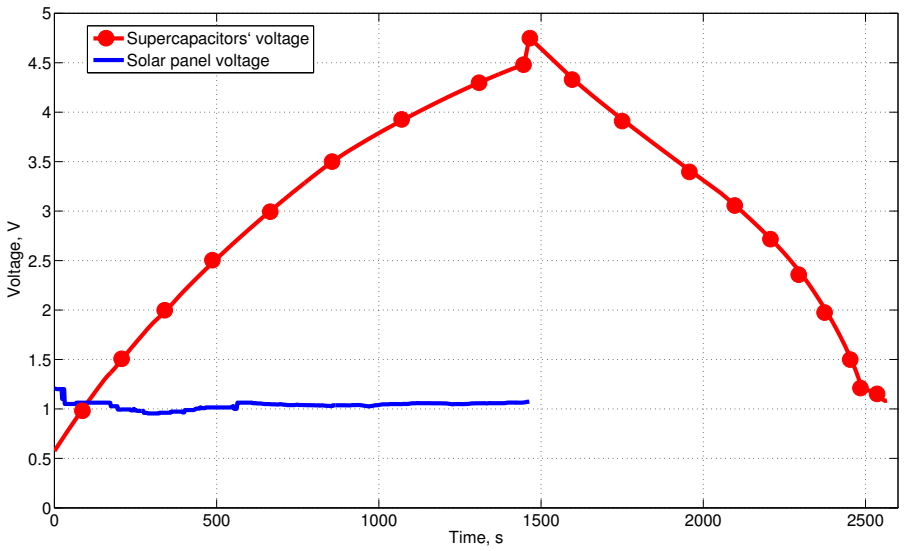

Figure 4.5: The charging and discharging of super capacitors.

mA current draw) the collected data over the network each 125ms. Because of the load, the super capacitors started discharging and reached the 1.203 voltage level in 1102 seconds (18 minutes). Starting from this voltage level the ESM can not provide 3 V for the proper operation of the TelosB node. However, already after 1024 seconds (17 minutes) at 1.206 voltage level the output DC/DC of ESM does not ensure stability of operation: the output ESM voltage fluctuates in the range from 0 V to 3 V level. If we were to run the node at a duty cycle of 1% in real WSN deployment, e.g., executing one reading every 12.4 seconds, the ESM could operate on super capacitors for around 28h 24min. It ensures that the sensor node operation on super capacitors can last more than the length of a night (12 hours), and provides the sensor node with an uninterrupted operation for this period.

The graph shown in 4.6 shows the charging behavior of the Li-ion battery when supplied by the solar panel. It took 334 minutes to fully charge the battery from 2.8V to its maximum voltage.

Next, we discharged the super capacitors and began to charge them with AC voltage generator. Obviously, the AC generator can not simulate an AC-based harvesting component exactly. However, the main purpose of this experiment was to verify the AC voltage rectifier made of a diode (bridge) rectifier and an electrolytic capacitor. Figure 4.7 demonstrates the charging of

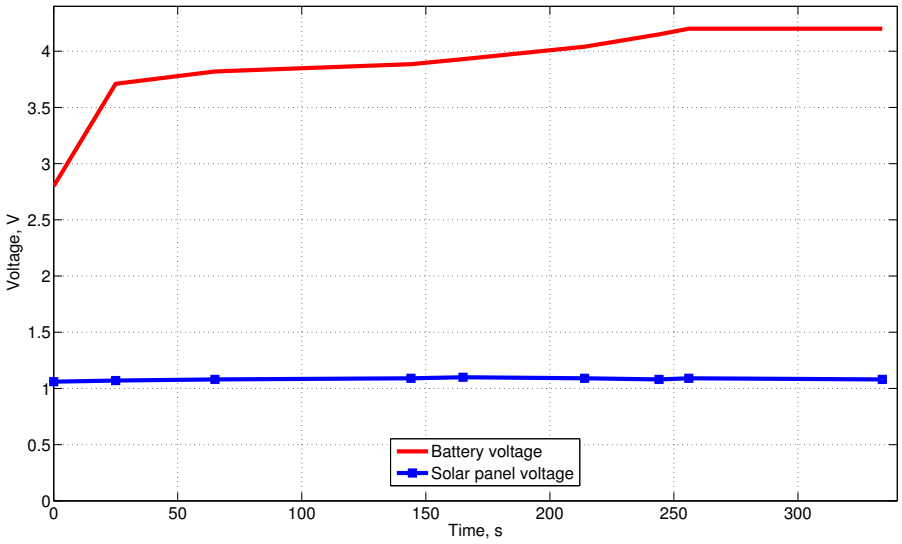

Figure 4.6: The charging of Li-ion battery with solar panel.

super capacitors with the AC voltage generator. Following [98], we simulated the piezoelectric converter which generates 3.5 V at 120 Hz. The goal of this experiment is to ensure that the ESM voltage rectifier enables to convert AC "ambient" voltage into DC voltage properly. We measured the rectified AC voltage and the voltage level of the super capacitors. The super capacitors were fully charged in 416 s (7 minutes).

To discharge the Li-ion battery we connected the TelosB sensor node to the ESM and run the same application (plus shining green LED (20 mA current draw)) as for super capacitors discharge. Over a period of time equal to 3 hours 55 minutes the battery have been discharged from 3.81 V to 3.77 V. According to its specification, the battery is expected to be fully discharged in approximately 94 hours with the showed rate. However, the duty cycle decreased up to 10% will prolong the sensor node lifetime to 39 days. In addition, the battery with the improved capacity (e.g. 2200 mAh) application will significantly increase the sensor node lifetime.

4.3.3 Potential Application

In spite of the popularity of wireless sensor networks (WSN), their application scenarios are still scanty [104]. In this section it is presented how the WSN paradigm can be applied to the entertainment area, and in particular to the domain of Paintball. This niche scenario poses

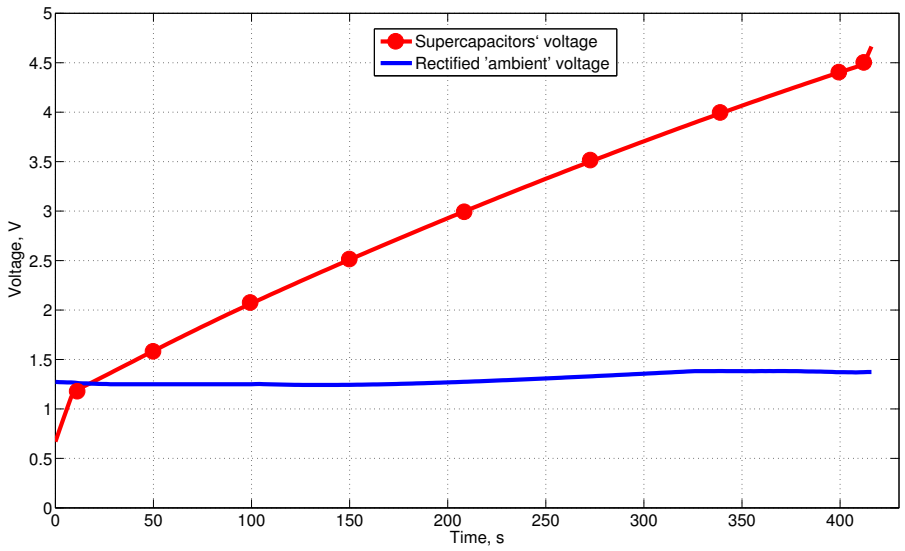

Figure 4.7: The charging of super capacitors with AC source.

challenges in terms of wireless sensor node lifetime and player localization. The main goal of localization in this context is to locate and track the player in order to facilitate his/her orientation, and to increase the level of safety. Long term operation could be achieved by adopting EnTick module.

Paintball research was carried out in collaboration with Vinay Sachidananda (Darmstadt University of Technology, Germany). My contribution to this work was focused on harware investigation.

Paintball is a sport in which players eliminate opponents from play by hitting them with paint. At present, players which are involved in Paintball use the Global Positioning System (GPS) [65] for orienting, walkie-talkie to communicate with each other or do not use these devices at all in case of short-term and bounded space scenario. However, providing each of e.g. hundred players with an individual GPS receiver is not always the preferred solution for cost reasons. Moreover, GPS itself is a power hungry component and needs at least four known satellites to be visible in order to find the coordinates of the receiver.

Key Principles of the System

The most well-known and popular Paintball types are speedball, woodsball and scenarioball. The scenarioball type appears to be most interesting for our research.

The playing field in scenario paintball (the storyline can be anything like civil war events, gangster wars of 1930s, the Oklahoma D-Day, storming a building or rescuing hostages) is normally large and in most cases unknown for the players. Moreover, the game may last several hours, therefore it would be reasonable to provide the players with a wireless sensor node to track their position by beacon nodes deployed throughout the playing field. It could be done for safety reasons in the interests of a paintball player, and with the aim of assisting him/her in orientation by showing his/her location on a Personal Digital Assistant (PDA).

Figure 4.8: Sketched paintball principles of our system.

However, we would like to pay your attention to some important requirements and constraints of the localization system. First of all, the beacon nodes should be distributed over the playing

field in such a way that the players are within the radio range to be tracked. The player location computation should be performed quickly to guarantee his relevant position. Since the playing field is in most cases a natural setting, the energy proves to be a limiting factor. The system accuracy depends on the signal power and frequency band, which, again, leads to energy restrictions. Thus, EnTick module perfectly fits for Paintball application to support a sensor node with a long-term operation.

Localization and Tracking System is described in details in [104].

4.4 TreBer module

The TreBer platform has the same platform as the EnTick module. However, instead of off-the-shelf super capacitors we use printed capacitors fabricated using direct write method [64]. This method enables printing of the capacitors directly on PCB. This section describes printed capacitors‘ fabrication process as well as introduces the experimental results of the prototypes.

4.4.1 Printed Capacitors

Electrode slurries are composed of 50 wt.% activated carbon with 24 wt.% PVDF polymer binder, 2 wt.% conductive carbon black additives, and 24 wt.% BMIM+BF4- ionic liquid. Electrolyte gels are a 1:1 composite of PVDF and BMIM+BF4-. Current collector slurries of conductive particles (such as nickel or carbon) in PVDF are also being developed. The viscosities of the slurries are tailored by adding a volatile solvent such as n-methyl-2-pyrollidone (NMP). The pneumatic dispenser printer extrudes a rapid succession of drops through a syringe tip onto the substrate. The drops eventually coalesce to form a film as NMP is removed via drying. To fabricate a capacitor, a five-layer planar and symmetric sandwich of current collector, carbon electrode, and electrolyte ink are printed as shown in the Figure 4.9. Figure 4.10 displays a micrograph showing the material morphology of the electrode and electrolyte.

To deposit supercapacitors on a substrate already populated with other components, automated alignment software and image processing are used to determine any open spaces available for printing. An image is taken of the substrate and any open spaces are translated into a bitmap image. The printer then deposits material in the pattern of the bitmap, and the alignment software maintains accurate orientation in between the printing of films even with the adjustment

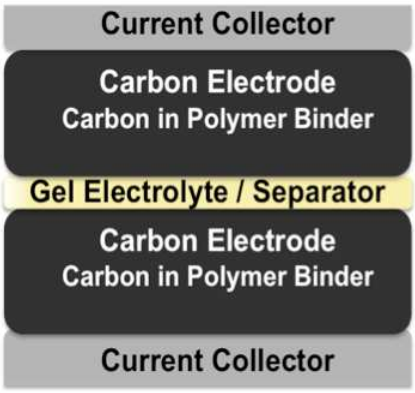

Figure 4.9: Schematic of a carbon electrochemical capacitor [courtesy of C. Ho, UC Berkeley].

of syringe tips, substrate rotation, or other printing parameters.

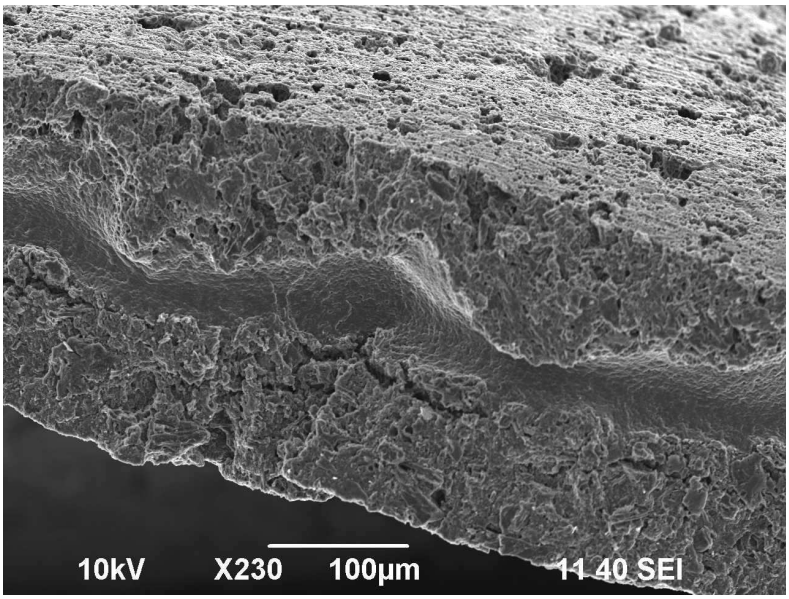

Figure 4.10: Micrograph showing the morphology of the carbon electrode and electrolyte active layers [courtesy of C. Ho, UC Berkeley].

The fabricated printed capacitors have good cycle life (over 120,000 cycles without performance degradation) and capacitances above 40 mF/cm^2 with an achievable power density of 575 uW/cm^2 and corresponding energy density of 10 uW-hr/cm^2.

4.4.2 Printed Capacitor Prototype

Figure 4.11 presents the prototype of printed capacitor.

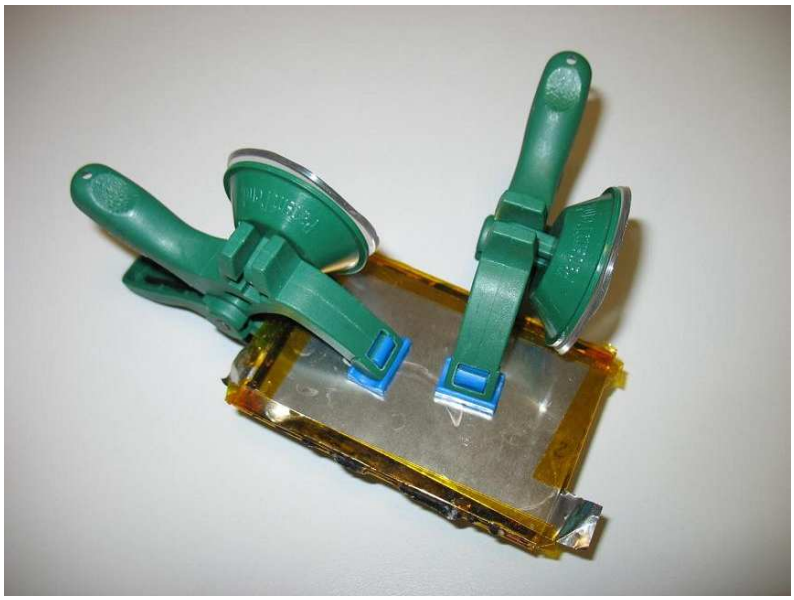

Figure 4.11: The prototype of printed capacitor.

The prototype is of 30 μF capacity. However, after 10 full charge-discharge cycles capacity may be significantly increased.

Figure 4.12 shows charge and self-discharge of this prototype. The charging was conducted under 1.4 V and 3 mA conditions.

The research on TreBer module was conducted in collaboration with Department of Materials Science and Engineering and Department of Mechanical Engineering of the UC Berkeley, USA.

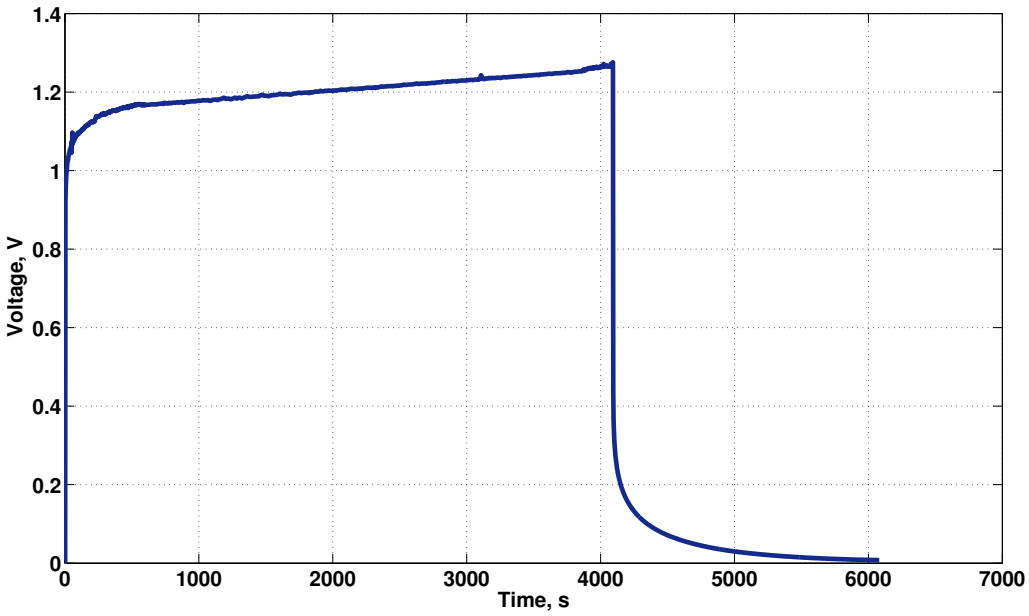

Figure 4.12: Printed capacitor charge and self-discharge.

Chapter 5

An Optimal Synchronization in Wireless Sensor Networks

In applications requiring just periodic monitoring services, major power savings could be obtained if the WSN nodes were perfectly synchronized. In this case, all nodes could remain mostly in sleep mode and they could wake up only for the time that is strictly necessary to run the scheduled monitoring tasks. Unfortunately, the time values measured by different nodes tend to drift away from one another due to the limited frequency stability of the local oscillators. As a consequence, the time offsets of various nodes can be kept within given tolerance boundaries only by running repeatedly some synchronization procedure, such as those described in [54], [37] or [25]. Obviously, frequently repeated synchronizations lead to better timing accuracy. Accordingly, the probability that some nodes are still sleeping while others are already running their own instance of the monitoring task is small. However, the energy dissipation due to frequent synchronization activities may be considerably large. On the other hand, if synchronization events are rare, the related activities are not particularly expensive in terms of energy. Nevertheless, the time intervals in which nodes are active must be longer than expected in order to allocate the worstcase time differences between node timers. Of course, this extension of the radio module activity in turn increases the amount of power consumption.

This research was carried out in collaboration with Prof. David Macii and Ageev Anton (University of Trento, Italy). My contribution to this work was focused on the modeling of the scenario and simulation.

5.1 Problem Formulation and Model Description

Let us consider a WSN consisting of N identical nodes that are mostly kept in sleep mode and wake up occasionally just to run two independent tasks, i.e.:

- One *monitoring task* of duration d_m and period $T_m >> d_m$;

- One *synchronization task* of duration d_s and period $T_s >> d_s$, which is able to keep node time offsets in the range [-ϵ_{max}, ϵ_{max}].

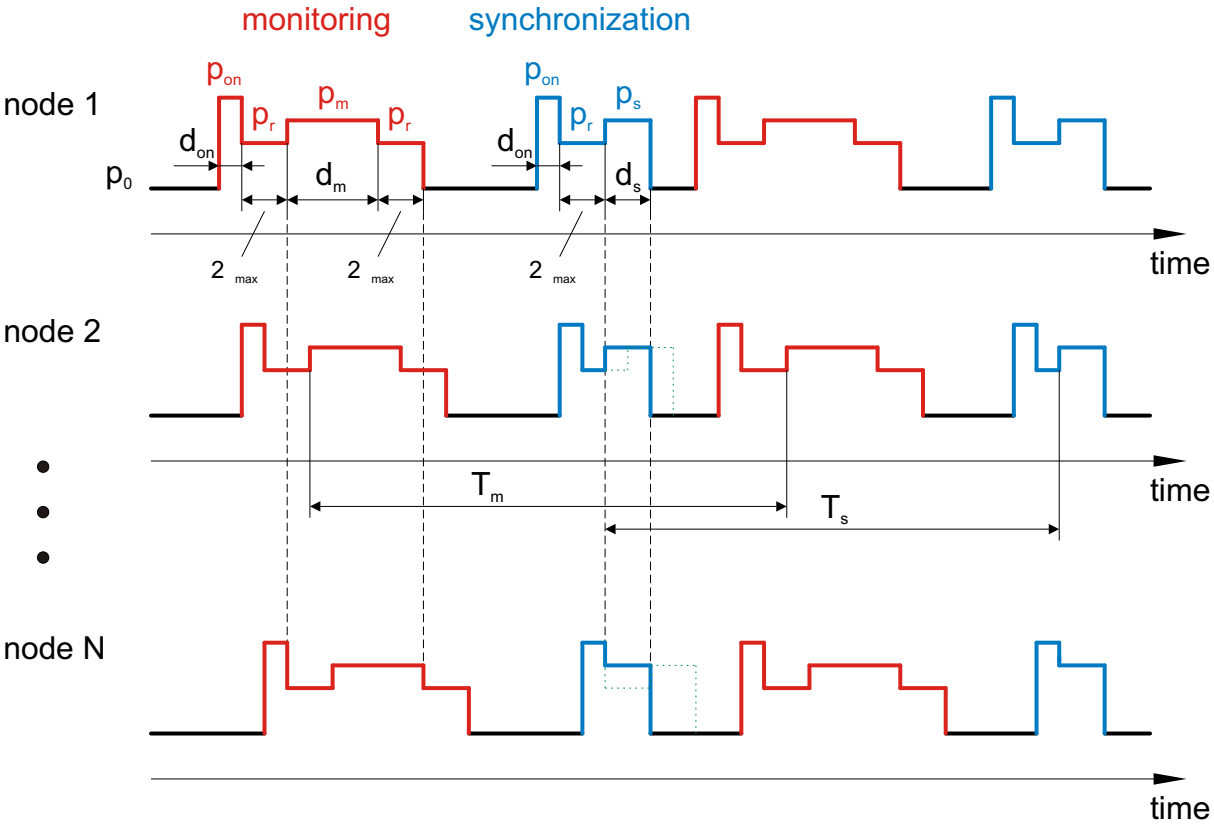

Figure 5.1: Qualitative power consumption patterns during the execution of monitoring and synchronization tasks on various nodes of a WSN.

Of course $\frac{d_m}{T_m} + \frac{d_s}{T_s} < 1$, because the overall utilization of a single node must be strictly lower than 100%. Also, switching from sleep mode to active state takes some time d_{on}, during which the amount of current drawn by the device may be quite large [115], as shown in Figure 5.1. Consider that in the proposed model both d_m and d_s include not only the time required to run the activities of a single node of the network, but also the time spent in waiting possible

messages and packets coming from other nodes. Moreover, even if the actual duration of each task may change randomly as a function of time (e.g., due to variable traffic conditions), we will assume that both dm and ds correspond to the respective maximum (i.e., worstcase) values. In fact, this choice does not alter the basic reasoning underlying the proposed model and it is quite realistic because all nodes usually may be programmed to wait data for a preset timeout value.

In general, a major difference exists between the timing behavior of monitoring and synchronization tasks. The monitoring task can be usually regarded as composed by unrelated activities. This means that a node collects, processes and sends its data to other WSN devices as soon as data are ready, independently of what other devices are doing. In this respect, some lack of synchronization is beneficial to avoid channel contention and packet collisions whenever a Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) protocol is used. However, given that the uncertainty associated with the chosen synchronization procedure lies in the interval [-ϵ_{max}, ϵ_{max}], in the worstcase the execution of an instance of the monitoring task on the node with the slowest clock rate may begin $2\epsilon_{max}$ seconds after the same instance running on the device with the fastest clock rate. Since in lowpower mode the radio module of a generic node is disabled, it could happen that the data collected by the "fastest" node are sent to the "slowest" one when it is still sleeping. Similarly, the "slowest" device might attempt to transfer data to the "fastest" node after this one has already entered into sleep mode. This situation is clearly shown in the leftmost part of Figure 5.1. Therefore, in order to avoid connection problems, the time interval during which the radio chip must be active should not just coincide with d_m, but it should be rather expanded by $2\epsilon_{max}$ both before and after the execution of each instance of the monitoring task. This means that the overall duration of the time interval in which a generic node is ready both to run its own monitoring task and to receive data is $d_m+4\epsilon_{max}$.

The scenario is quite different in the case of the synchronization task. In fact, any synchronization protocol relies on a set of coordinated activities which are usually triggered by the node starting the whole procedure. In the following, for the sake of simplicity, this node will be referred to as *synchronization master* (SM). When the SM broadcasts a synchronization packet, all nodes receiving this packet start executing the synchronization algorithm approximately at the same time. Although the time of execution of the synchronization procedure may differ slightly from node to node (e.g., due to possible differences among the activities performed by the various nodes) we can assume that the instances of the same synchronization task are si-

multaneous. Of course, a necessary condition to synchronizeWSN nodes properly is that when the SM starts the procedure, the other devices must be already active. Therefore, for the same reasons explained above, all nodes should switch on $2\epsilon_{max}$ seconds before the scheduled synchronization time. However, in this case prolonging the node activity by $2\epsilon_{max}$ after the end of the synchronization procedure is unnecessary. Indeed, all activities roughly finish simultaneously and the residual time offsets immediately after synchronization can be assumed to be negligible. As a consequence, the duration of the interval during which the SM is active is exactly $d_s+2\epsilon_{max}$. This situation is shown in the central part of Figure 5.1. Observe that the duration of the "active" interval of any WSN node different from SM is $d_s+\varepsilon$, where ε represents the delay between the time when the node $i \neq SM$ switches on and the moment when the synchronization procedure actually begins. In general, ε is a random variable because it depends on the time offset between the node i and the SM. If the synchronization procedure is started by the "fastest" device, then $\varepsilon \in [0, 2\varepsilon_{max}]$. Conversely, if the synchronization procedure is triggered by the "slowest" node, then $\varepsilon \in [2\varepsilon_{max}, 4\varepsilon_{max}]$. In all other cases, (i.e. when the elected SM is any node with intermediate clock speed) the values of ε lie in $[\Delta, \Delta+2\varepsilon_{max}]$, where Δ represents the offset between the time of the SM and the time measured by the node clocked by the "slowest" oscillator. Of course, the distribution of ε depends on the probability density function of the timing errors as well as on the policy chosen to elect the SM. If we assume that the time offsets of the various nodes are uniformly distributed in the range $[-\varepsilon_{max}, \varepsilon_{max}]$ and that all nodes have the same probability of becoming the SM, it can be shown that $E[\varepsilon] = 2\varepsilon_{max}$, where $E[\cdot]$ is the operator returning the expected value of its argument. Given that we are mostly interested in analyzing the average power dissipated by a generic node in different conditions regardless of its role in the network, in the following the duration of the "active" interval associated with the synchronization task will be set equal to the corresponding mean value, i.e. $d_s + 2\varepsilon_{max}$.

Starting from all the considerations explained above, if we refer to:

- p_0 as the average power consumption in sleep mode;

- p_{on} as the average power spent to switch on the radio module;

- p_m as the average power dissipated for running the monitoring task and for exchanging data with other devices;

- p_r as the average power consumed when a node is ready to receive messages from other devices;

- p_s as the average power associated with the execution of an instance of the synchronization task;

the overall average power dissipated by a WSN node over a time interval T equal to the least common multiple (LCM) of periods T_m and T_s results by definition from:

$$\overline{P} = \overline{P}_m + \overline{P}_s + \overline{P}_0 \tag{5.1}$$

where:

$$\overline{P}_m = \frac{d_m p_m + 4\epsilon_{max} p_r + d_{on} p_{on}}{T_m} \tag{5.2}$$

is the average power contribution due to the monitoring task,

$$\overline{P}_s = \frac{d_s p_s + 2\epsilon_{max} p_r + d_{on} p_{on}}{T_s} \tag{5.3}$$

is the average power contribution due to the periodic synchronization task,

$$\overline{P}_0 = p_0 \left(1 - \frac{d_{on} + d_m + 4\epsilon_{max}}{T_m} - \frac{d_{on} + d_s + 2\epsilon_{max}}{T_s}\right) \tag{5.4}$$

is the average power consumption when the node is in sleep mode. Notice that in 5.1 the effect of overlapping tasks is neglected. This is due to two reasons. First, the overlapping probability is generally quite small. Indeed, if the monitoring and synchronization tasks are assumed to be independent, the overlapping probability is simply equal to $\frac{d_m}{T_m}\frac{d_s}{T_s}$, and this quantity is usually much smaller than 1 if $d_m << T_m$ and $d_s << T_s$. Second, even when two instances of the monitoring and synchronization tasks overlap, we have just a small decrease in energy dissipation. Such a difference is due to the fact that the radio module switches on only once, rather than twice. Nonetheless, the time to run both task is still roughly equal to $d_m + d_s$, regardless of the adopted scheduling policy. Thus, the energy saving is just $d_{on} p_{on} + 2\varepsilon_{max} p_r$, which is negligible compared to the energy dissipated to run both tasks.

Expression (5.1) is significant because it can be used to estimate the synchronization period minimizing the average power dissipated by a WSN node. In fact, it is known that the limited frequency stability of local oscillators is the main uncertainty contribution affecting synchronization uncertainty in WSNs [26]. Moreover, the law describing how the time values measured by different nodes drift away from one another is approximately a linear function of time. Thus, if $|\nu_{max}|$ is the absolute value of the relative frequency offset of the least accurate local oscillator, the maximum time error just before performing a new synchronization is approximately

$\epsilon_{max} \approx |\nu_{max}| \cdot T_s$. Usually, the value of ν_{max} for low–cost crystal oscillators lies within ± 100 ppm. Thus, by replacing the expression of ϵ_{max} into (5.1), it follows that the derivative of (5.1) with respect to T_s is equal to zero for

$$T_s^* = \sqrt{\frac{T_m[d_s(p_s - p_0) + d_{on}(p_{on} - p_0)]}{4|\nu_{max}|(p_r - p_0)}} \tag{5.5}$$

which returns the optimal synchronization period. In conclusion, if the basic parameters of (5.1) are known, (5.5) provides an interesting criterion for energy–aware synchronization. Note that if $|\nu_{max}|$ tends to 0 (e.g., because the clock frequency skews are partially compensated by the adopted synchronization algorithm), T_s^* grows towards infinity. This is reasonable, because if $|\nu_{max}|$ is small, running frequently repeated synchronizations is unnecessary.

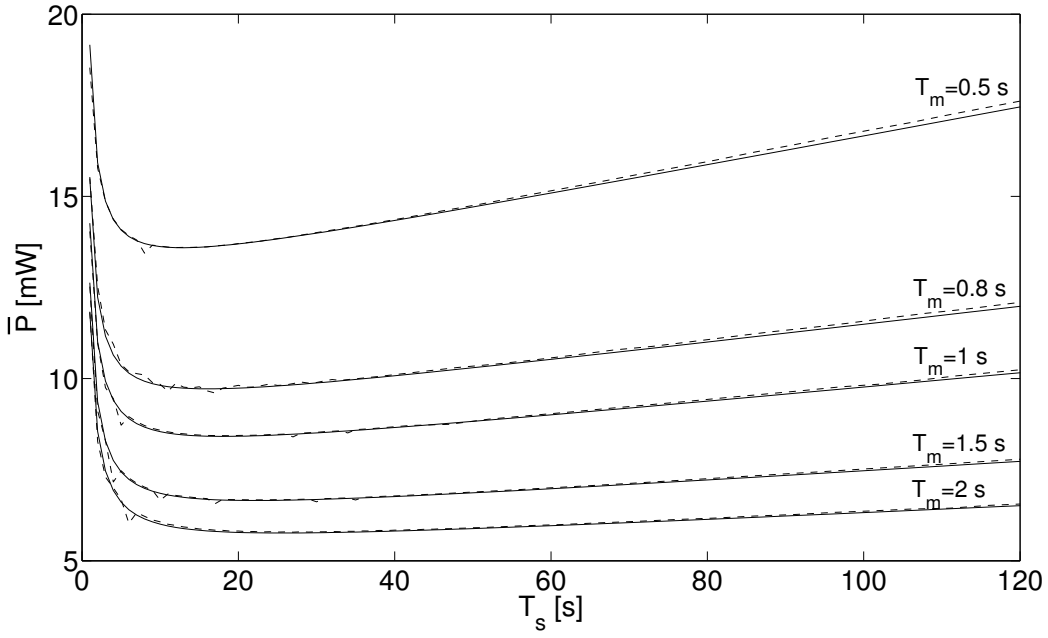

Figure 5.2: Average dissipated power as a function of the synchronization interval T_s for different periods of the monitoring task (i.e., $T_m = 0.5, 0.8, 1, 1.5, 2$ s). The solid lines result from (5.1), whereas dashed lines are obtained through simulations. In all cases the value of the worst–case frequency offset is $\nu_{max} = 100$ ppm.

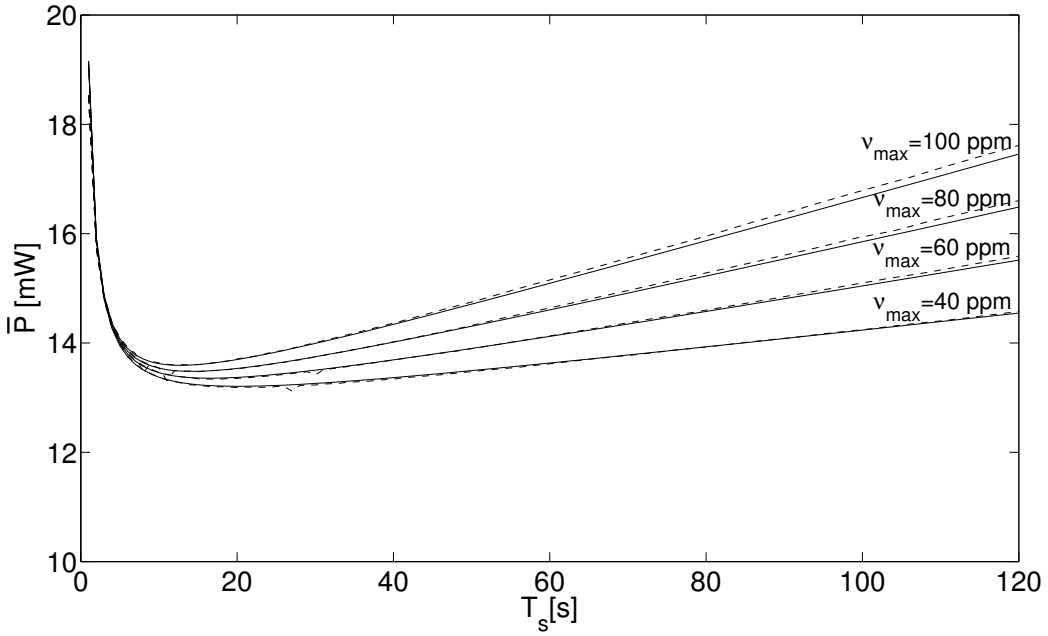

Figure 5.3: Average dissipated power as a function of the synchronization interval T_s for different worst–case relative frequency offsets (i.e., $\nu_{max} = 40, 80, 60, 100$ ppm). The solid lines result from (5.1), whereas the dashed lines are obtained through simulations. In all cases the period of the monitoring task is $T_m = 0.5$ s.

5.2 Simulation Results

In order to better understand the operative implications of (5.5), in this Section the results of some Matlab™ simulations are reported. The application developed for simulation purposes is able to generate a power waveform in accordance with the model described in Section 5.1. All task scheduling operations (including early wake–up and late switch–off times) rely on a simulated digital timer running nominally at 32768 Hz. As a consequence, the output waveforms can be generated with 1–tick resolution. In order to make simulations closer to reality, the model of the local clock offers the possibility to adjust both its systematic relative frequency offset and the amount of random jitter. Accordingly, several simulations were repeated with different values of T_s, T_m and ν_{max} during variable observation intervals comprising multiple instances of both monitoring and synchronization tasks. The common values of the parameters used in all simulations are the following: $d_{on} = 100$ μs, $d_m = 92$ ms, $d_s = 130$ ms, $p_{on} = 57.4$ mW,

$p_r = p_m = 56$ mW, $p_0 = 2.8$ mW, $p_s = 53$ mW. Such values are in reasonable accordance with the execution times and the power consumption contributions of WSN nodes working in different conditions. In Figures 5.2 and 5.3 the power consumption values resulting from simulations (dashed lines) are compared with the corresponding theoretical curves (solid lines) obtained from (5.1). In particular, in Figure 5.2 the average power curves are plotted as a function of T_s after setting $\nu_{max} = 100$ ppm and for five different values of T_m (i.e., $T_m = 0.5, 0.8, 1, 1.5, 2$ seconds). Dually, in Figure 5.3 similar curves are plotted for a fixed T_m value (i.e., $T_m = 0.5$ s), while changing ν_{max} (i.e., $\nu_{max} = 40, 80, 60, 100$ ppm). Observe that, even if the simulations include the possibility of having overlapping tasks, the difference between each pair of simulated and theoretical curves is negligible. In the first set of simulations, the optimal synchronization period T_s^* ranges from 12.4 s when $T_m = 0.5$ s to 24.8 s when $T_m = 2$ s. Indeed, it doubles when T_m increases by 4 times, as expected. In the second set of simulations T_s^* grows from 12.4 s for $\nu_{max} = 100$ ppm up to 19.6 s for $\nu_{max} = 40$ ppm. The results shown in Figures 5.2 and 5.3 confirm that in general the optimal power consumption is quite sensitive to ν_{max}. If ν_{max} becomes very small, synchronizations are rare and the total power consumption is mostly due the monitoring task, as it should be in the ideal case. Notice also that if the parameters of (5.5) are affected by some significant uncertainty it is preferable to overestimate T_s^*, because the amount of dissipated power tends to increase much more smoothly for $T_s > T_s^*$.

5.3 Experimental Results

The theoretical analysis described in Section 5.1 was validated experimentally using a small WSN consisting of five Crossbow TelosB nodes. Each device was programmed to run two concurrent applications, i.e. a simple humidity monitoring task and the synchronization procedure described in [25], suitably modified to switch off the radio chip when the node is supposed to be idle. A further device connected to a PC was used to collect some significant data from all WSN nodes (e.g., the synchronization intervals). The synchronization algorithm reported in [25] is adaptive. This means that the time interval between two subsequent synchronizations changes at run–time depending on the wanted target accuracy. In particular, for a given value of ε_{max}, the synchronization interval is expanded [i.e., multiplied by $(1 + a)$, with $a \geq 0$] if the estimated probability of having in–tolerance nodes is larger than a preset threshold Q. Conversely, the

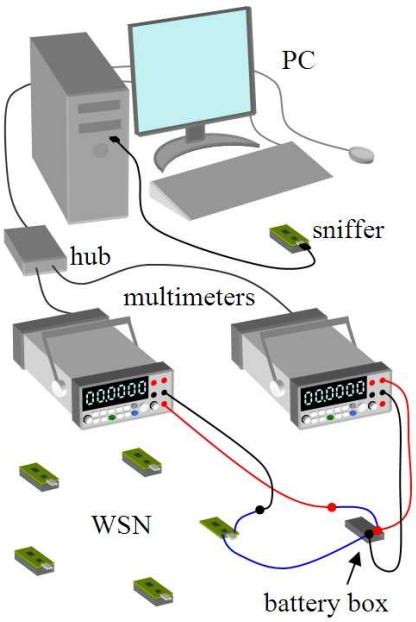

Figure 5.4: Experimental setup. Five Xbow TelosB nodes are programmed to run both a periodic humidity monitoring task and the synchronization protocol described in [25]. The applied voltage and current drain of a generic node are measured simultaneously by means of two DMMs Agilent 34411A which are connected to a PC collecting both measurement results and sensor information by means of a further node (i.e. a sniffer).

interval between two synchronizations is shortened [i.e., multiplied by $(1-b)$, with $0 \leq b < 1$] when the percentage of nodes with in–tolerance clocks is smaller than Q. As a consequence, the synchronization interval values tend to exhibit a lognormal distribution [25]. The mean of this distribution can be regarded as an estimator of the synchronization period T_{s_Q} for which the probability of having nodes with time offsets in the range $[-\varepsilon_{max}, \varepsilon_{max}]$ oscillates around Q.

Starting from these considerations, the experimental activities were carried out in two phases. In the first one, after setting $\varepsilon_{max}=1$ ms, $Q=100$ %, $a=0.02$ and $b=0.33$, the synchronization procedure was repeatedly executed for several minutes in order to collect a significant amount of interval values. Accordingly, the mean of the lognormal distribution was computed and the maximum relative frequency offset ν_{max} of the considered set of nodes was estimated from $\varepsilon_{max}/T_{s_Q}$. In the case considered ν_{max} is smaller than expected, i.e. 22 ppm.

In the second phase, at first the adaptive behavior of the synchronization algorithm was disabled by setting $a = b = 0$. In this way, the synchronization intervals remain constantly

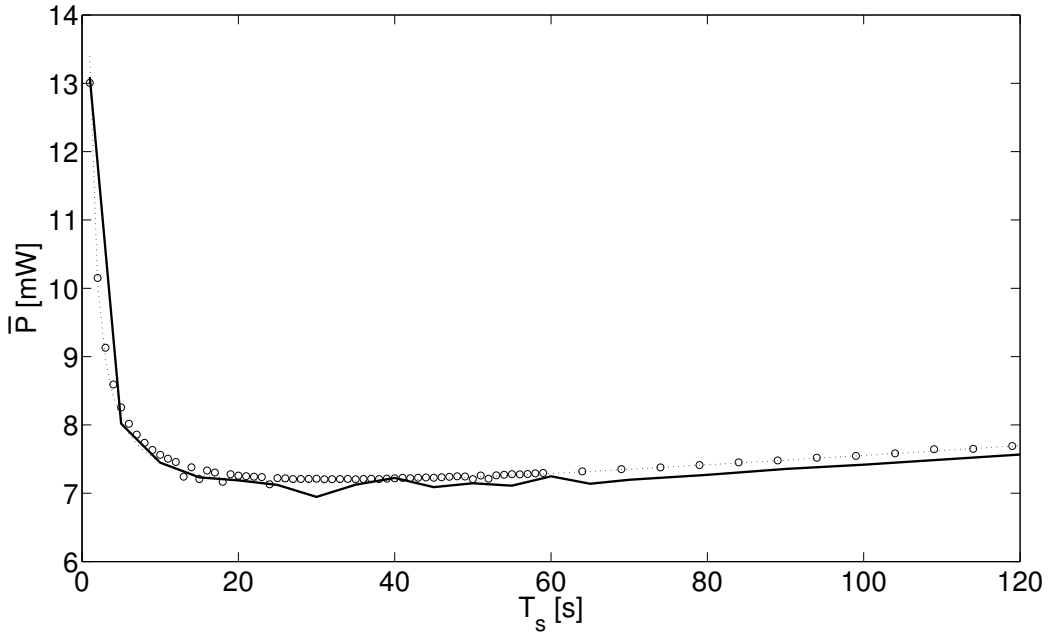

Figure 5.5: Average power consumption as a function of the synchronization interval T_s for $\nu_{max} = 22$ ppm (namely the estimated maximum relative frequency skew in the considered WSN). In the picture, the solid line refers to the measurement values collected using the setup shown in Figure 5.4, the dashed line results from (5.1), and the bubble points correspond to simulations. The optimal synchronization period is $T_s^* = 29.5$ s.

equal to the initial value. Then, all nodes were programmed to collect a humidity sample every $T_m = 0.6$ s and to run the synchronization procedure for different periods in the range $[1, 120]$ s. The voltage supply of a single WSN node as well as its current drain were measured using two digital multimeters (DMMs) Agilent 34411A (one used as a voltmeter and the other as an amperometer) connected to the PC, as shown in Figure 5.4. The integration time of both DMMs was set to 100 power line cycles (i.e., 2 s) in order to exploit the best instrument resolution and to maximize normal mode noise rejection. The duration of each experiment was about $10T_s$ seconds. Since the battery voltage is approximately constant, each average power consumption value can be estimated as the product of the average current drain and the average supply voltage, i.e. $\overline{P} = \bar{I}\bar{V}$. All measurement results are plotted in Figure 5.5 (solid line). Notice that the shape of the curve is in good agreement with the simulation results shown in Figure 5.2, although the values are smaller. The standard uncertainty associated with $\overline{P}$ was calculated

using a procedure similar to that described in [81], i.e.:

$$u(\overline{P}) = \sqrt{\bar{I}^2 u^2(\bar{V}) + \bar{V}^2 u^2(\bar{I})} \tag{5.6}$$

where $u^2(\bar{V})$ and $u^2(\bar{I})$ are the combined standard uncertainties related to current and voltage measurements, respectively. Both $u^2(\bar{V})$ and $u^2(\bar{I})$ include random and systematic contributions, as stated in [81]. The former ones can be estimated using the typical *type A* evaluation procedure and are particularly significant in the case of short measurement times (e.g., for small value of T_s). Conversely, the systematic contributions are independent of measurement duration and can be estimated from instrument specifications by following a *type B* evaluation procedure [23]. In our experiments, the maximum standard uncertainty value resulting from (5.6) is 0.3 mW, which is adequate because it is about one order of magnitude smaller than the measured values.

In order to perform a final check about the correctness of the proposed model we also compared the collected measurement results with the theoretical curve given by (5.5) and with the corresponding simulations. The values of the parameters in (5.1) were estimated with the help of a digital oscilloscope Agilent DSO7032A. In particular, such values are $d_{on} \approx 2.5$ ms, $d_m \approx 76$ ms, $d_s \approx 126$ ms, $p_{on} \approx 7.5$ mW, $p_r \approx p_m \approx 52$ mW, $p_0 \approx 2.8$ μW and $p_s \approx 53$ mW. Both the theoretical curve (dotted line) and the simulation results (bubble points) shown in Figure 5.5 are in perfect accordance with the experiments, thus confirming the validity of the model.

Chapter 6

Combustible Gases and Early Fire Detection

Fires or toxic gas leakages may have grave consequences like significant pecuniary loss or even lead to human victims. In this chapter we present an autonomous wireless sensor system for early fire and gas leak detection. The system consists of two modules: a gas sensor module and an energy scavenging module. The operation of the gas sensor module is based on the pyrolysis product detection which makes it possible to detect fire before inflammation. In addition, the on board gas sensor can identify the type of leaking gas. A generic energy scavenging module (see Chapter see Chapter 4), able to handle both alternating current and direct current based ambient energy sources, provides the power supply for the gas sensor module. The harvested energy is stored in two energy buffers of different kind, and is delivered to the sensor node in accordance to an efficient energy supply switching algorithm. At the end of this Chapter we demonstrate the experimental results on gas detection, energy consumption evaluation, and show how to ensure the system autonomous operation.

This research was carried out in collaboration with Denis Spirjakin, Mikhail Ivanov, and Prof. Alexander Baranov (Moscow Aviation Technological University, Russia). My contribution to this work was focused on experimental part.

6.1 Background and Motivation

Efficient environmental monitoring with sensor networks covering large territories and ensuring continuous monitoring of wildfires and toxic gases can be regarded as a problem of top priority [48]. From a technical point of view, the wild land fire problem, for instance, has not been solved yet [114].

Though satellites are able to provide wide area sensing, in terms of real-time spatial resolution and sensitivity, this method has considerable restraints. Moreover, it is associated with the exceptionally high deployment and operational costs, which remains the factor of decisive importance [76, 32].

Another well-known approach to monitoring wild fires or gases leakage is based on getting data on the emissions source. With the ultrasensitive instrumentation aboard the vehicles (airplanes, trailers, etc.), it is possible to acquire data in cross patterns [114]. Apart from the fact that this method requires operation and maintenance personnel to be involved in, spatial and time resolution is limited to a point measurement at the vehicle current location. On top of that, sensitivity in the case of airborne platforms should be extremely high so as to secure the high-altitude detection of ground emissions after the gases have propagated to a considerable distance from the sources.

During the last few years, the paradigm of Wireless Sensor Networks (WSN) [41] has been adopted to tackle this problem. Cheap and tiny wireless sensor devices which operate in a cooperative and autonomous manner deployed over a territory may detect hazardous gases and monitor wild fires [28]. There are two main strategies used by sensor nodes to detect a wild fire. The first strategy is based mostly on temperature, relative humidity and barometric pressure sensors [48, 78], whereas the second uses smoke detectors [36]. The systems of these kind described in [48, 114, 28], however, can only detect the direct flame. The system referred to in [78], for instance, is intended for detecting and localizing a small fire (about tens of square meters). This means that in most cases the sensor nodes with the capabilities of direct fire detection are not useful. For instance, the authors of [48] managed to record the flame passage in California, USA, using the WSN facilities, but the sensor nodes were rapidly destroyed and the fire was extinguished only at the final stage. Thus, the primary goal of sensor nodes is to detect fire at an early stage [36].

The high power consumption of combustible gas detection and fire monitoring systems [86, 102, 5, 80] is also a limiting factor for nodes deployment. The sensor nodes used for this specific application with on-board batteries can not really last for a long time. So, the second goal is to prolong the long-term operation of the sensor nodes.

6.2 System Overview and Principles of Operation

The system consists of two modules: the gas sensor module and the Energy Scavenging Module (ESM). Gas detection is performed by the gas sensor module, whereas the ESM is exploited as a power supply for it.

The main purpose of the designed gas sensor module is to allow one to get information about possible inflammation prior to the formation of smoke and the appearance of a direct flame [111] or to register the presence of a dangerous gas in the atmosphere.

We use a *temperature scanning procedure* of the sensing element of pre-fire (gas) detector to separate in time the processes which occur in the semiconductor layer's electronic subsystem from the relatively slow chemical processes which run on the catalytically active surface of the catalyst. This process increases the sensors' sensitivity and selectivity. The working principle of the pre-fire sensor operation is based upon the detection of pyrolysis products due to smoldering or overheating of combustible and electric insulating materials, primarily carbon oxide (CO) and hydrogen (H_2).

The goal of the ESM is to mediate between an energy harvesting component and the gas sensor module, providing a flexible modular system, and to increase the gas sensor module lifetime. This module supports various energy scavenging technologies [99].

Figure 6.1 shows the hardware architecture of the developed system. The semiconductor sensor acts as a sensing element to determine the gas composition of the environment. The data-conversion line (sensor - operational amplifier - microcontroller) converts the gas-composition dependent parameters of the sensor to digital format/code. A microcontroller-based drive/control circuit sets the required operation modes of the node, implements the mechanism for obtaining the digital code from the analog-to-digital conversion line, and provides preliminary data processing and compilation for the data wireless transmission module circuit.

The ESM architecture contains seven main blocks. The first block DC/DC(1) operates as

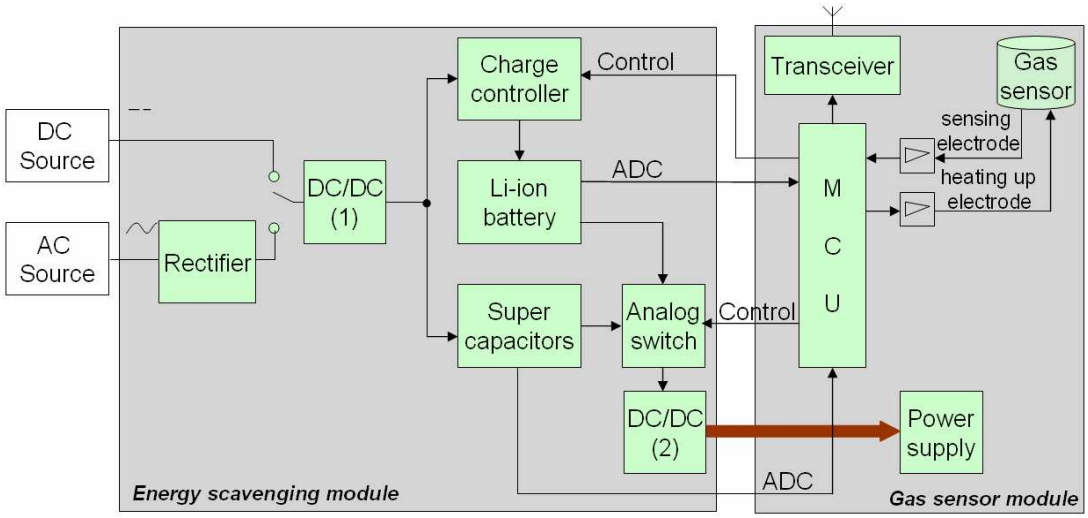

Figure 6.1: Architecture of autonomous gas detection system.

a transformer. It is supplied either by the "AC-based" ambient source (i.e., noise, vibrations) through the Rectifier, or by the "DC-based" ambient source (i.e., solar radiation, thermal energy). The purpose of the DC/DC(1) block is to provide the ESM with a stable input voltage. Super capacitors, wired in series with each other and in parallel with the DC/DC(1) block, operate as a buffer. In other words, if the ambient source is available the system exploits environmental energy transformed to the necessary voltage level by the DC/DC(1) block. Otherwise, the system can be supplied by the super capacitors in case they are charged. The sensor node monitors the voltage level available in the super capacitors to verify if they can be a power supply for the system or not. The third power supply depicted in the system architecture is a Li-ion battery. The Li-ion battery is a backup power supply. The charging/discharging state of the battery is controlled by the Charge Controller which, in turn, is managed by the sensor node. Thus, the ESM is designed in such a way that a sensor node can be supplied by one of three energy sources: super capacitors, Li-ion battery or directly by the environmental energy source. However, a user must specify the type of ambient source (both AC-based and DC-based harvesting components are supported, although the two cannot be present at the same time) using "jumpers" before the sensor network deployment. Before the DC/DC(2) block generates the required output voltage to supply the sensor node, the Analog switch chooses the power supply on the basis of the voltage level available and power management algorithm described in

the next paragraph. The Analog Switch and the Charge Controller are managed by the onboard sensor node microcontroller.

6.3 Gas Sensor Module Implementation

The gas sensor module is shown in Figure 6.2. The main unit of the gas sensor module is a MicroController Unit (MCU). For the designed node we have chosen the 8-bit MCU ADuC836. It has low power consumption, power supply monitor (both features are important for energy constrained WSNs), two analog-to-digital converters, enough Flash/EE memory and an on-chip debug system.

Figure 6.2: Gas sensor module.

The MCU communicates with the gas sensor through two OPA2340 operational amplifiers, optimized for low voltage operation. One amplifier is connected to the sensing electrode of the sensor whereas the second one is connected to the heating up electrode. Typically, a gas sensor is the main energy sink in an autonomous embedded system. To overcome this problem, our gas sensor module uses a pulse mode of gas measurement. Pulse Width Modulation (PWM) signals with a 1kHz frequency and 3.3 V amplitude go from the MCU control circuit to the sensor's heating unit with a 30-second interval. As a result, the sensing element warms up. If there are flammable, toxic, or explosive gases of various compositions and concentrations in the

atmosphere, the conductivity of the semiconductor sensor's sensitive layer changes. By adopting this mode, it was possible to reduce the total power consumption of the gas sensor from more than 100 mW (typical of continuous operation mode [18]) to the level of 30 mW, thereby enabling the long-time battery supply of the sensor. Semiconductor and thermo-catalytic sensors made using aluminum-oxide membranes [113, 112] are a perfect fit for the pulse mode of measurement.

The measured data (i.e., the presence of gas, its composition and concentration) converted by the Analog-to-Digital Converter (ADC) to a digit serial code go to the MCU control circuit where they are preprocessed. The resulting data is passed to the wireless transmission module TG-ETRX2, a ZigBee standard [55] module using the 2.4 GHz frequency channel.

The implementation of the ESM is described in Section 4.2.

6.4 Experimental Results

In this section we present the experimental results on gases detection and sensitivity of the system. We also investigate energy consumption at various heating profiles of the sensor, evaluate the Energy Scaveging Module (ESM) with respect to the gas sensor module, and discuss our experimental system deployment.

6.4.1 Pre-fire Detector Response

The pre-fire detector response experiment describes how the sensor node sets the pre-fire detector in operating mode. In addition, we show the difference in time response between a bulk-substrate sensor [83] and the sensor (used in this work) made using an aluminum-oxide membrane [113].

Figure 6.3a shows the oscilloscope display of the voltage pulses heating the sensor's sensitive layer. The pulse frequency is 675 Hz which ensures effective control of the sensor temperature making feedback possible. The heating time depends on the preset temperature, the heating pulse duration and the sensor design.

For a semiconductor sensor made on an aluminum-oxide membrane using micromachining technology (Figure 6.3c), the Pulse Width Modulation (PWM) time for a 450 °C operating temperature of the sensor is of the order of 500 ms (cf. 1.5s for a bulk-substrate sensor, Figure 6.3b). Control of PWM pulse duration ensures various heating/cooling profiles including gradual (smooth) heating of the sensor.

6.4.2 Gases Detection

Figure 6.4 shows the sensor's response in the absence and presence of 0.2% methane, alcohol fume and wood pyrolysis.

After analyzing the obtained results, one can see that, in the absence of toxic and inflammable gases, variation in temperature of the semiconductor sensitive layer (the sensor is ON) leads to an increase of its conductivity and, respectively, a drop of its voltage. After termination of the heating pulse (the sensor is OFF), the layer's conductivity goes back to the initial value (Figure 6.4, curve 1). Wood pyrolysis is accompanied by $_2$ and emission in the atmosphere. With their liberation, after termination of the heating pulse, the sensitive layer's conductivity rapidly drops at the first stage. However, after a considerably longer time, it steps up to the initial level (Figure 6.4, curve 2).

If the sensitive layer interacts with methane (as it can be seen from Figure 6.4, curve 3), a noticeable drop in the sensitive layer's conductivity will occur, while its recovery will take several seconds. In the case of alcohol fume measurement, we have the experimental relationship of the 3-rd type. The sensitive layer's conductivity remains practically unchanged, but getting back to initial conductivity after heating termination requires maximum time.

One may conclude from the above that, provided the sensitive layer's conductivity is measured within 0.2-0.5 s after heating termination, it is possible to separate mixed gases and define gas types and concentrations.

6.4.3 Sensitivity Investigation

Sensitivity plays significant role in reliability of fire and gas detection. At the same time, sensitivity impacts on the system energy consumption: the more the sensor is heated up, the more sensitive it is, hence more energy it requires. In this section we investigate sensitivity of the sensor in terms of temperature (PWM of heating pulses) and figure out if heating with high temperature is really essential. Figure 6.5 shows how the temperature of sensor‘s sensitive layer depends on various heating modes. It is not recommended to heat the sensor to higher then 550 °C since the sensor can be damaged at 600 °C. This graph shows that heating the sensor with a PWM of more than 85% is dangerous.

Figure 6.6 presents how the sensor response to gases depends on PWM heating modes. The

higher the temperature of the sensitive layer and the higher the concentration of the combustible gas (methane), the better the response. Based on the magnitudes of these curves we graphed Figure 6.7.

Figure 6.7 represents the sensor sensitivity. Zero sensitivity corresponds to sensor response to the air at appointed temperature. The graph shows that the optimal sensitivity is reached at the 75-80% PWM.

The next subsection will investigate how various modes of PWM affect on energy consumption, so that we could determine the most energy efficient and reliable in terms of sensitivity heating profile of the sensor.

6.4.4 Energy Consumption

The main power sink in the system is the gas sensor. Much energy is being spent during its heating. That is why it is important to investigate energy consumption of the gas sensor.

Pulse Width Modulation (PWM) is a technique for electrical power control between fully *on* and fully *off* states. Obviously, to heat the sensing element up to a predefined temperature applying various duty cycles will take various ranges of time, and hence consume various levels of energy. In this section we will demonstrate how a user may perform an efficient power management of the gas sensor node using various heating profiles for the sensor.

The sensor resistance is a function of temperature [73] and can be computed as:

$$R = R_0 \cdot (1 + \alpha \cdot (T - 20°C)) \tag{6.1}$$

where R_0 is the resistance of the heating layer of the semiconductor sensor (12 Ohm) under normal conditions, α is the temperature coefficient of resistance (0,0027 $^1/°$C in our case), T is the temperature (in °C). The resistances of the sensor at 450 °C (R_{450}) and 550 °C (R_{550}) is 25.9 Ohm and 29.1 Ohm respectively. Because the resistance increases with temperature, the current decreases when the sensor is heated up (see Figure 6.3d). The initial current can be calculated using Ohm's law:

$$I_{start} = \frac{U}{R} = 275mA \tag{6.2}$$

where U is a voltage during heating (3.3 V in our case). The currents I_{end450} and I_{end550} at R_{450} and R_{550} are 127 mA and 113 mA respectively.

Next, using the *trapezoidal rule* and knowing empirically the time required for the sensor heating (see Tables 6.1, 6.2) till its "on-position" we calculated the magnitude of the charge q_{450} necessary to heat the sensor up using a constant current: 38 mA·s. Hence, energy consumption during the constant sensor heating up till 450 °C can be calculated as:

$$E_{450} = q_{450} \cdot U = 125mJ \tag{6.3}$$

However, to calculate the energy consumption under the same conditions, but using another duty cycle, 80% for instance, we cannot simply multiply E_{450} by 0.8. In fact, empirically, we established that the heating up time for sensors at various duty cycles does not change linearly.

PWM, %	**Heating time, s**	**Energy consumption, mJ**	**Gain in energy as against 100%, %**
20	2.1	251	-100
30	0.91	163	-30
40	0.6	143	-14.3
50	0.43	128	-2.4
60	0.33	118	5.7
70	0.26	109	13.3
80	0.24	115	8.6
100	0.21	125	0

Table 6.1: Gas sensor module energy consumption during the sensor heating up to 450 °C.

Table 6.1 and table 6.2 show the energy consumption of the gas sensor module at the time of sensor heating and represents how the heating time of the sensor depends on the duty cycle of the heating pulses. According to our experiments, a 70% duty cycle is the most efficient for both 450 °C and 550 °C heating profiles. Negative numbers in the "Gain in energy" column of Table 1 relate to long term of heating. For low duty cycles, the heating layer cools down enough, so that next heating pulse must heat it up till the previous level and heat it up to the next level as well. Based on data presented in Tables 6.1 and 6.2 the system may be adjusted by the user

PWM, %	**Heating time, s**	**Energy consumption, mJ**	**Gain in energy as against 100%, %**
20	5	576	-150
30	2.5	432	-87.5
40	1.2	277	-20
50	0.8	230	0
60	0.62	214	7
70	0.5	202	12.5
80	0.45	207	10
100	0.4	230	0

Table 6.2: Gas sensor module energy consumption during the sensor heating up to 550 °C.

in accordance with application requirements.

6.4.5 ESM as a Power Supply

To evaluate the ESM with respect to the gas sensor module we used the ESM as a power supply for the gas sensor node.

The solar panel BP SX305M was installed indoors in the electronic lab with bright internal illumination. Then we connected the solar panel wires to the ESM power connector. The multimeter was connected directly to the pins of the super capacitors taking into consideration that they are wired in series. The "jumper" was installed in *DC ambient source* position.

Figure 6.8 shows the voltage level on supercapacitors during its charging and discharging. The purpose of this experiment is to demonstrate that the super capacitors could be promptly charged from the ambient source, and that they are able to support an uninterrupted operation of the gas sensor module in real conditions. The voltage was stable at approximately 1.1 V. Two super capacitors were fully charged in 1464 seconds (≈25 minutes). Next, we connected the gas sensor module to the ESM. The gas module was programmed to scan ambient atmosphere and send the measured data once every five minutes. Because of the load, the super capacitors started discharging and reached the 1.6 voltage level (the module reset happened) in 1863 seconds (≈31 minutes). Starting from this voltage level the ESM can not provide 5 V for the proper operation

of the sensor module. Using the sleep mode of the MCU and modem increases the module lifetime to 4290 seconds (≈72 minutes).

To discharge the Li-ion battery (Figure 6.9) we connected the gas sensor module to the ESM and run the application of ambient atmosphere sensing and transmission of measured data once per ten seconds. Over a period of time equal to 176 minutes the battery was discharged from 4.18 V to 2.8 V. The graph shown in Figure 6 shows also the charging behavior of the Li-ion battery when supplied by the solar panel. It took 334 minutes to fully charge the battery (750 mAh) from 2.8V to its maximum voltage (4.2 V).

6.4.6 System Deployment

Having carried out the experiments on sensor's response, sensitivity, and energy consumption evaluation it was essential to prove the system autonomy.

The autonomous wireless sensor system for combustible gases detection and wild fire monitoring was deployed in the campus of Moscow Aviation Technological University in October, 2009. The experimental setup is depicted in Figure 6.10 and includes the gas sensor module, the energy scavenging module, solar panel BP SX305M, laptop (since then called "host") to control the gas detection system, a methane-cylinder and piece of wood with a lighter to emulate the gas leakage or pyrolysis.

Between the system and the host (distance 15 meters) was organized a wireless network using the ZigBee standard [55]. A specifically designed "friendly" software interface shown in Figure 6.11 aims to facilitate the network management control. The screenshot shows that one sensor node, i.e., our system, was able to connect to the host. If the gas concentration is less than 2%, and, in this case, is equal to 0.26%, the "Sensor 1" window will be green. "OK" means that the system has enough energy for further operation. In case a gas concentration exceeds the 2% threshold, the window will turn red and an alarm will go off.

Every 30 minutes we emulated either methane leakage or generated pyrolysis y setting the piece of wood on fire. The system measured the atmosphere and transmitted data to the host each 15 minutes.

We applied 450 °C, 70% duty cycle of heating pulses profile. From 8 a.m. to 8 p.m. we monitored the voltage level on the super capacitors and the rechargeable battery of the energy scavenging module, and tracked the voltage generated by the solar panel. The three graphs

shown in Figure 6.12 represent the voltage state during the 12-hour period.

At the time of the experiment it was a cloudless and sunny day. The battery was discharged from 4.2 V to 3.88 V (discharge cut-off voltage is 2.75 V) during the night. The super capacitors started to charge from the solar panel before 8 a.m. as it is depicted in the graph. Because the solar radiation is not strong in the morning, the battery had to be replenished, and because the system was operating - the caps were charging slowly. However, at around 4 p.m. they were fully charged (whereas the battery was fully charged at around 3 p.m.), so that the system had a chance to be supplied directly from the solar panel. Starting from 6 p.m. the solar panel could not generate enough energy to support the operation from the ambient source that led to a smooth super capacitors discharge. At the end of our experiment the battery was still fully charged. We should note that there is a lack of strong solar radiation in Russia during autumn, but it was sufficient for the stable system operation.

After that, we also determined the time spent for data inquiry (the last measured state of atmosphere) from the system. The host received the response from the gas sensor module in 1.359 seconds.

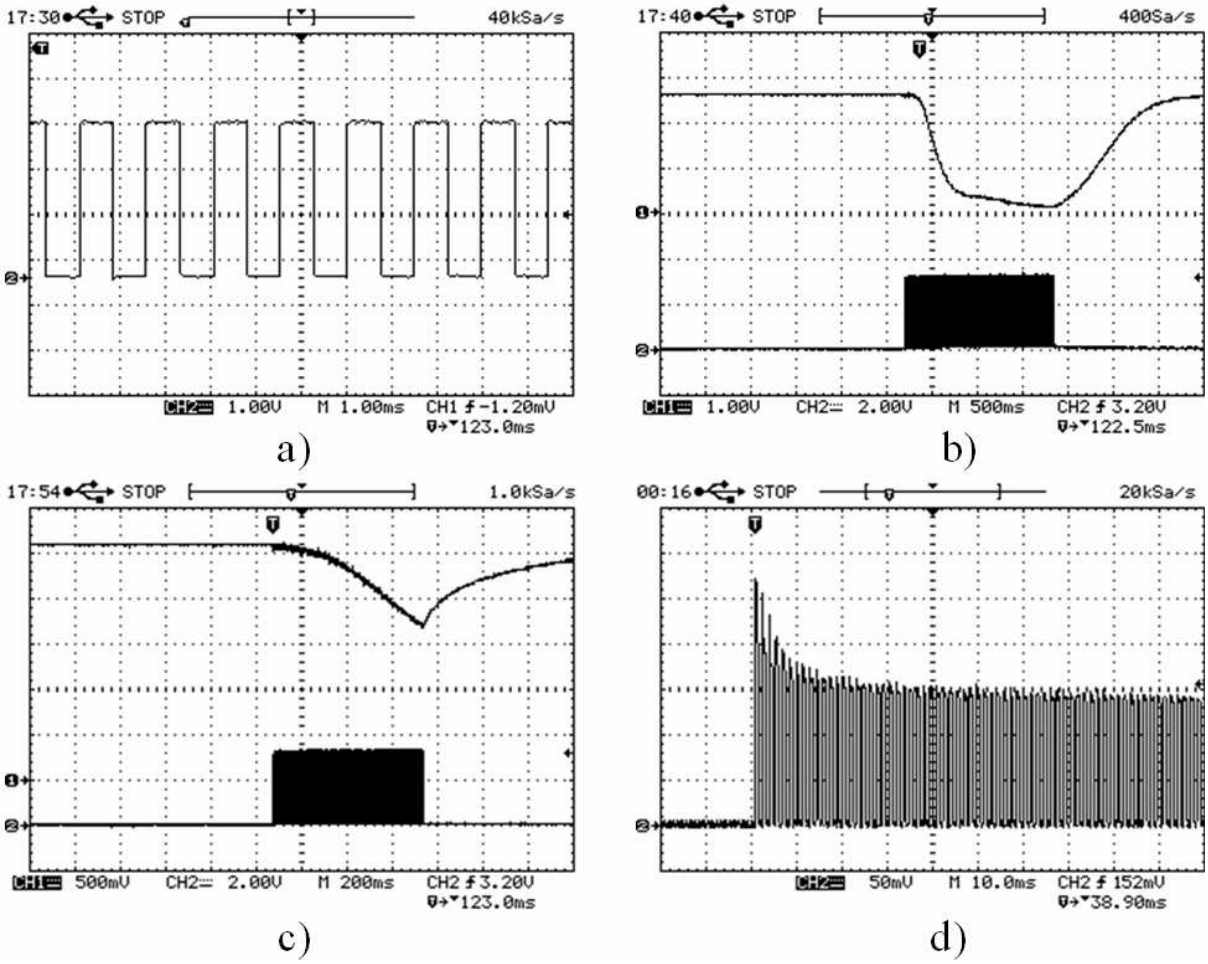

Figure 6.3: (a) 50% PWM generated by microcontroller, (b) variation in the sensitive layer's resistance of the bulk-substrate sensor in relative units (upper graph) and PWM generated by microcontroller (lower graph), (c) variation in the sensitive layer's resistance (in relative units) of the sensor made using the aluminum-oxide membrane (upper graph) and PWM generated by microcontroller (lower graph), and (d) variation in current running through the sensor's (the sensor made using the aluminum-oxide membrane) heating element.

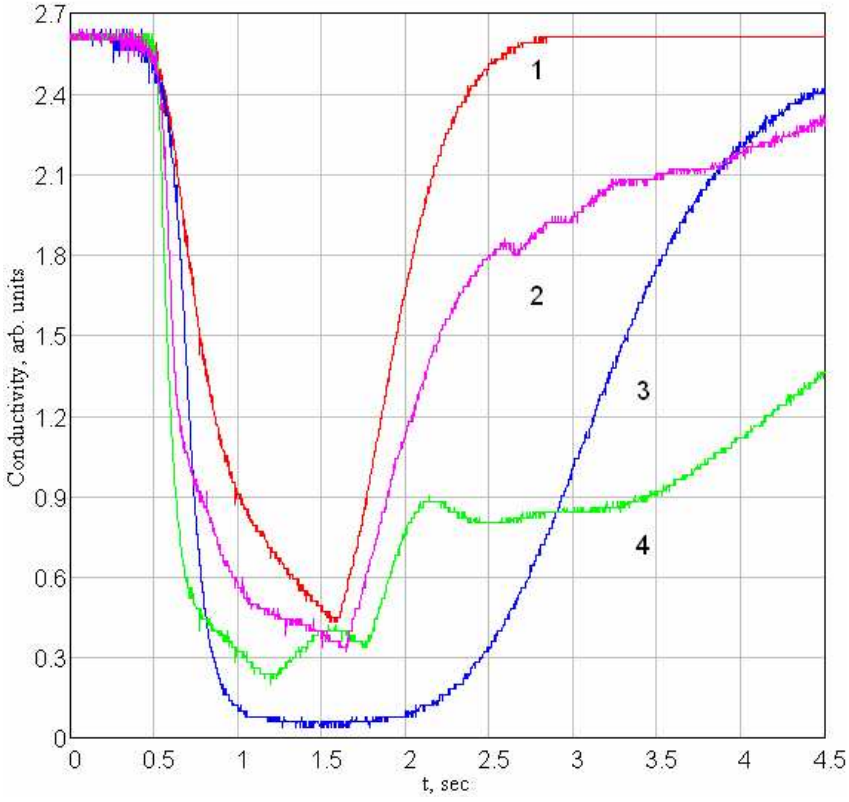

Figure 6.4: Sensor's response to various gases at 500 °C: 1 - the air; 2 - pyrolysis; 3 - 0.2% methane; 4 - alcohol fume.

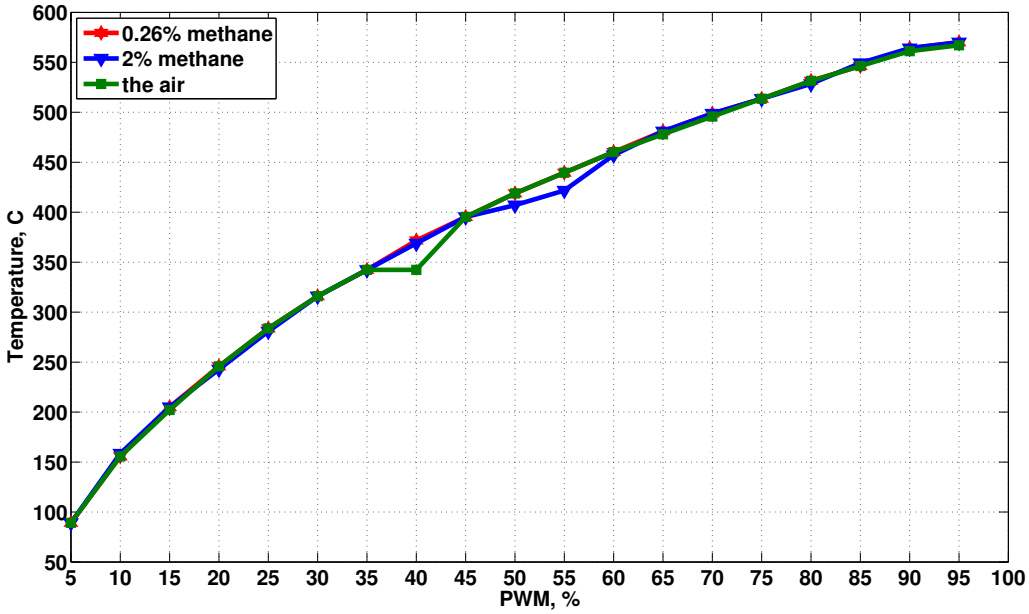

Figure 6.5: Temperature of sensitive layer against various PWM modes under normal conditions.

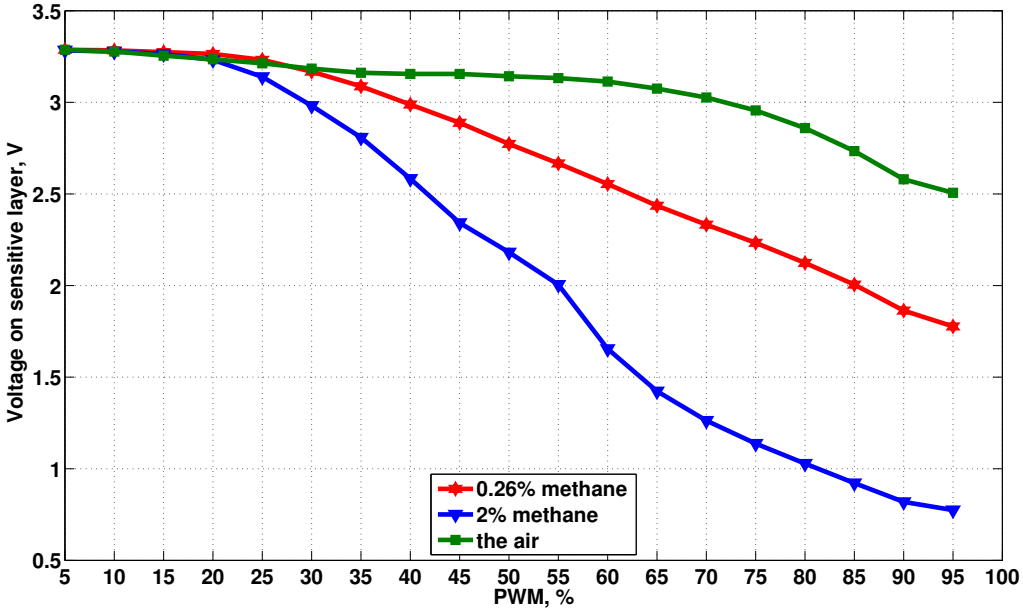

Figure 6.6: Sensor response against various PWM heating modes.

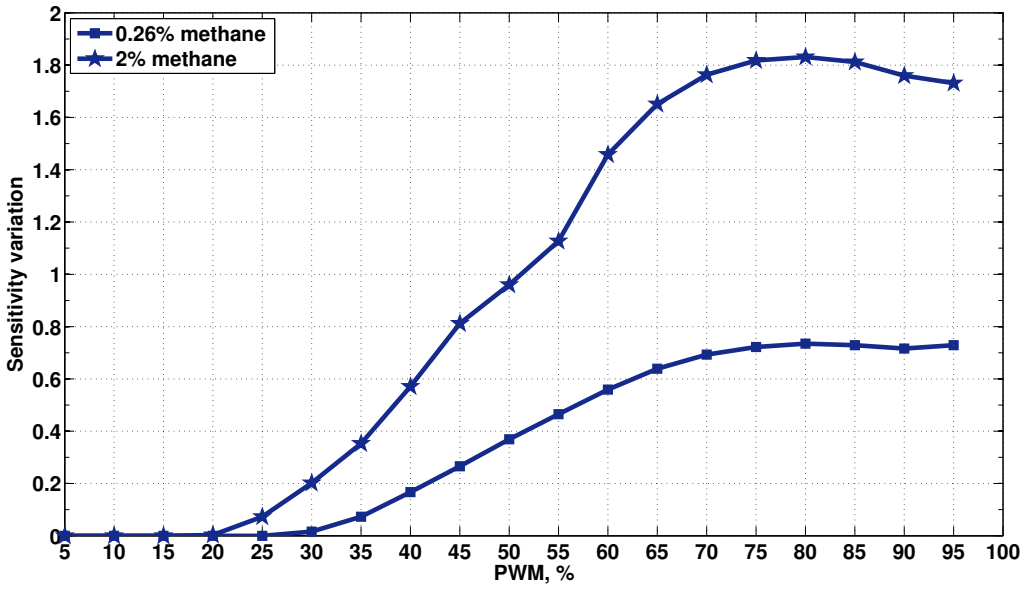

Figure 6.7: Sensitivity of the sensor.

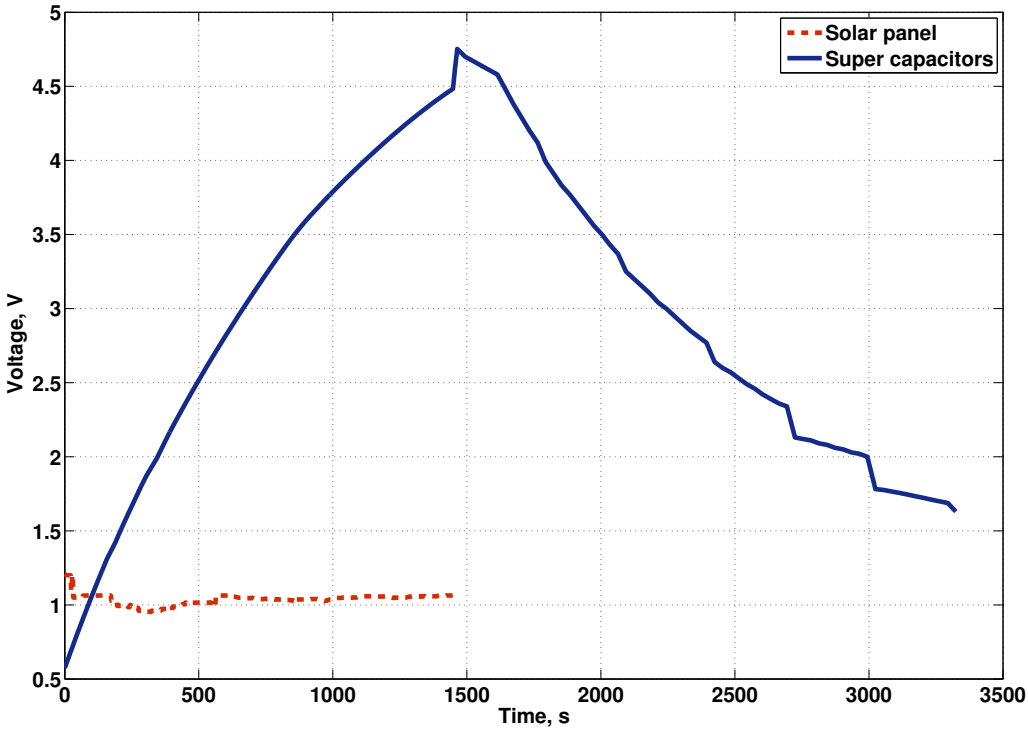

Figure 6.8: The charging and discharging of super capacitors.

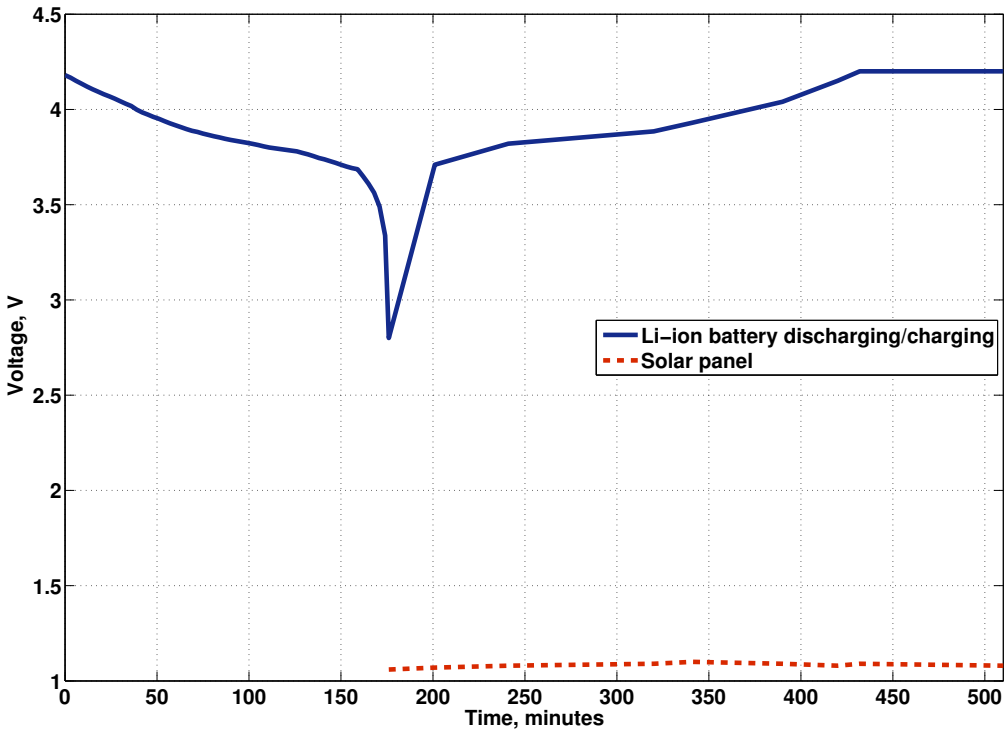

Figure 6.9: The charging and discharging of Li-ion battery.

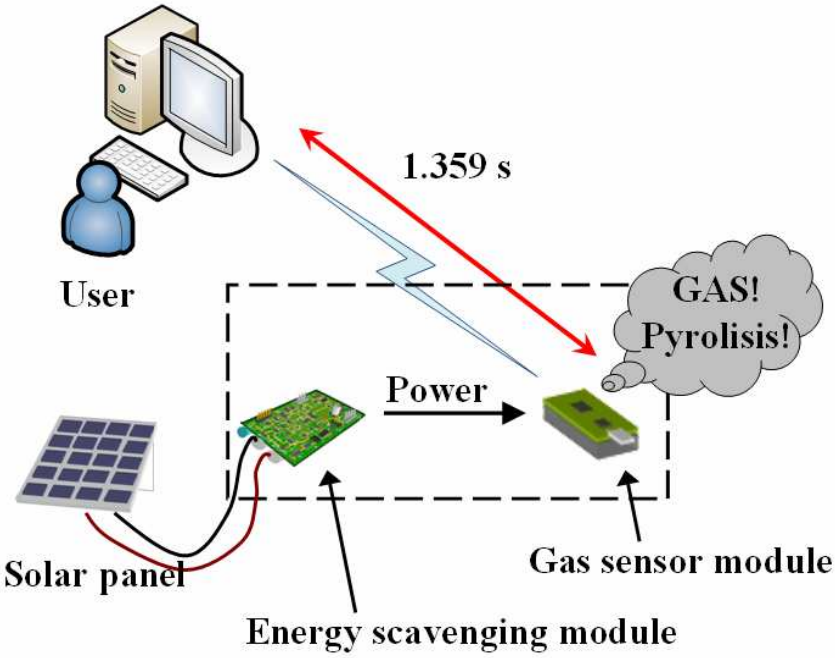

Figure 6.10: Schematic representation of the experimental deployment.

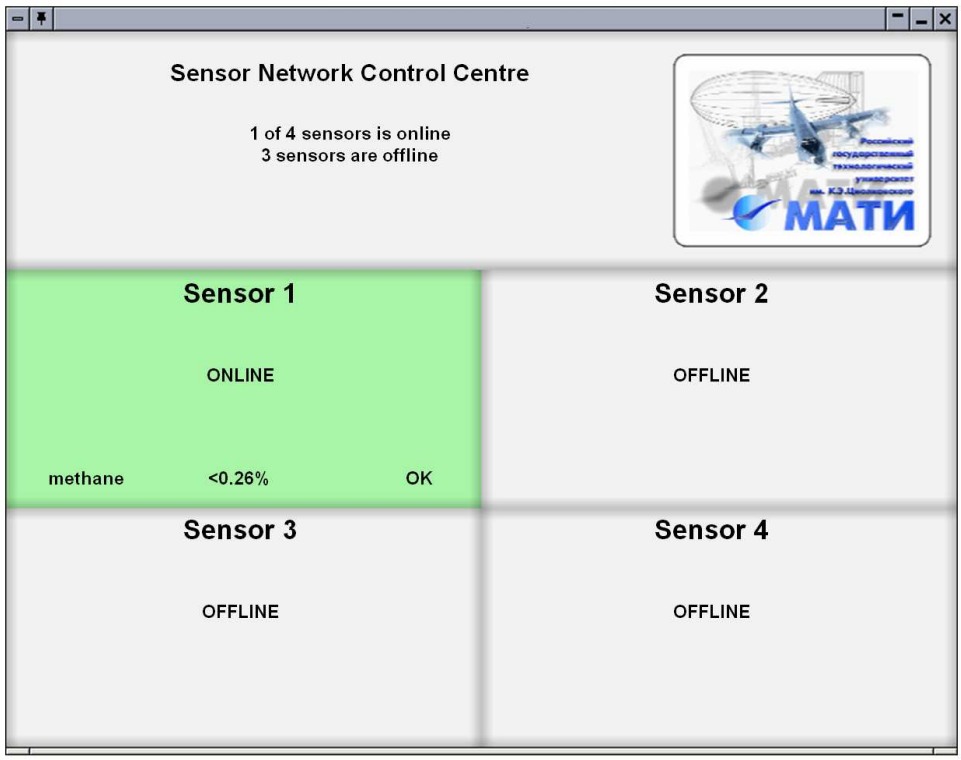

Figure 6.11: Interface for network control.

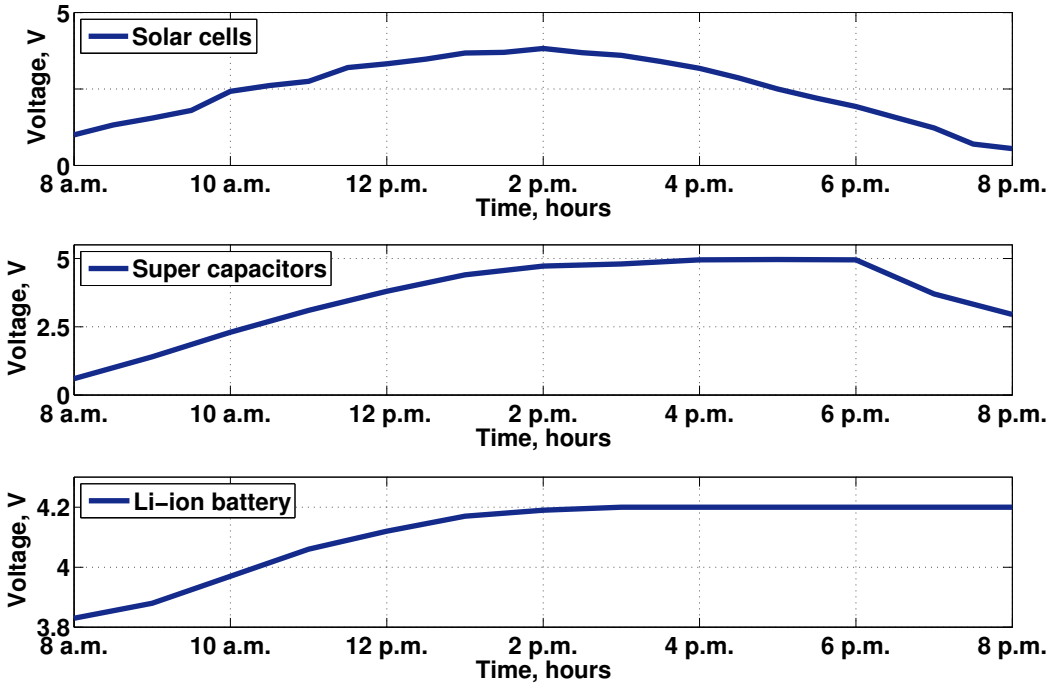

Figure 6.12: Voltage level on solar panel, super capacitors, and Li-ion battery during 12-hour period.

Chapter 7

Powering of Wireless Brain-Machine Interfaces

A brain-machine interface (BMI) is a direct communication pathway between a brain and an external device. BMIs have been designed to assist, augment or repair human cognitive or sensory-motor functions. It should be noted, however, that the term BMI is quite often (and erroneously) confused with the neuroprosthetic applications that aim at restoring damaged hearing, sight and movement. The remarkable cortical plasticity of the brain allows signals from implanted prostheses to be handled (after their adaptation) by the brain as natural sensor. First neuroprosthetic devices implanted in humans appeared in mid 1990s, after years of experiments on animals.

The main difference between BMIs and neuroprosthetics is that the latter typically implies connection of the nervous system to a device, whereas BMIs usually connect the brain (or nervous system) with a computer system (see Figure 7.1). In fact, practical neuroprosthetics can be linked to any part of the nervous system (e.g. peripheral nerves), while the term BMI normally defines a narrower class of systems interfaced with the central nervous system.

Figure 7.1 shows how a completely-implantable BMI may restore the limb mobility of paralysed patients or amputees. Though the details of this system need to be thoroughly elaborated in future research, it is clear by now that the BMI for human clinical applications should be encased in the patient's body to the highest possible degree. This may be achieved through wireless telemetry which seems to offer an effective solution of the problem. The prosthesis

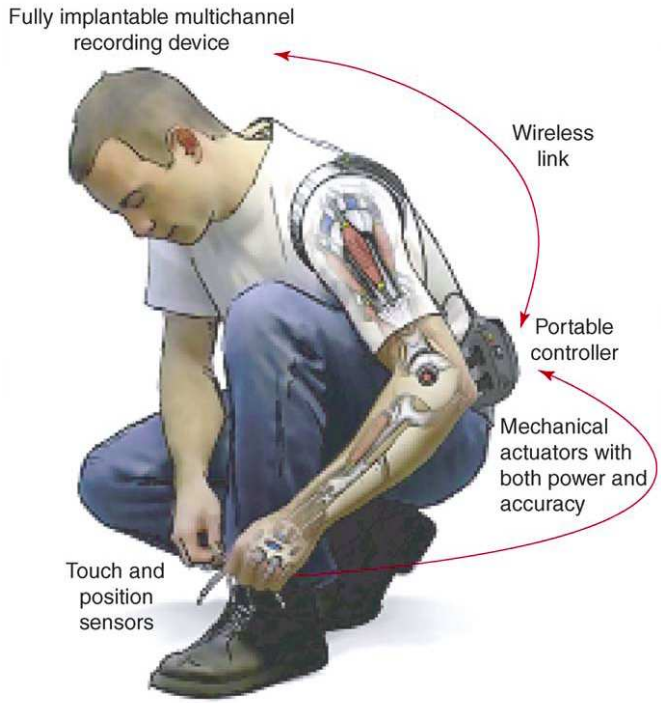

Figure 7.1: How a fully-implantable BMI could restore limb mobility in paralyzed subjects or amputees [75].

should not only have the functionality of the human arm or leg in terms of power and accuracy provided by the actuators, but also be equipped with the touch and position sensors to transmit signals back to the subject's brain.

7.1 Powering via RF

The research goal is to determine the maximum available power for a given overall size constraint (1 10 mm) and build systems in standard 65nm CMOS that deliver this power to the implant. Basically, our work is focused on a power supply for wireless neural implant presented in Figure 7.2.

The options for powering the implants are:

- Batteries;
- Energy scavenging inside the body;

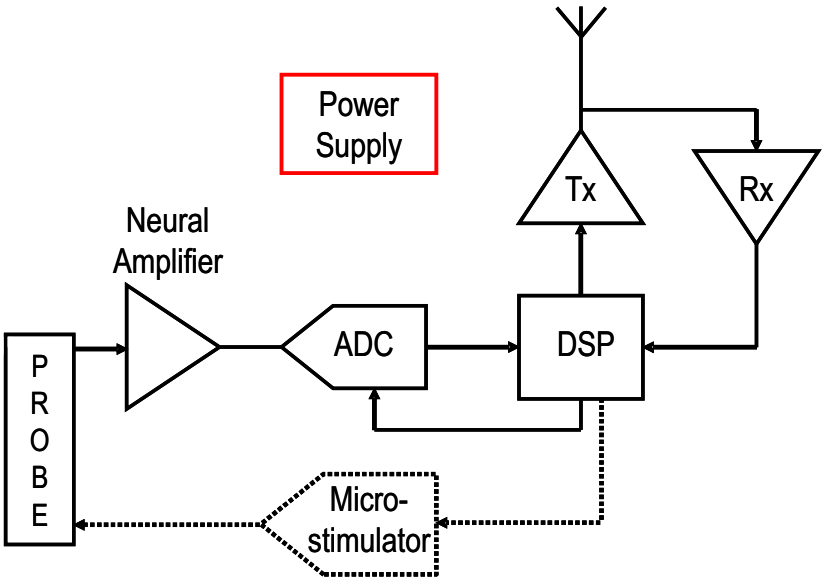

Figure 7.2: A generic wireless neural implant.

- Powering via RF;

Each option has some problems. The batteries, for instance, have the problems with the size and replacement that makes it difficult their use inside the skull. Energy scavenging inside the body is a relatively young reserch area and it can not provide the implant with the necessary power. Powering via RF is the most appropriate option: it is a versatile technique where the energy source is outside the body. However, this kind of power source has the challenges and opportunities:

- Extractable power depends heavily on size of the receive antenna;
- Current research on wireless BMIs exceed recommended limits on EM radiation drastically. Thus, the top bound of transmitting frequency is limited up to 10 MHz in accordance with [33];
- Smallest antenna for wireless power transfer so far 5x5 mm^2;
- Received power is expected to be very low for smaller antennas;
- Advancements in ultra-low power building blocks likely to make powering implants via smaller antennas feasible;

To power an implant we will use the Powering via RF option. We will design RF transceiver based on Class-E amplifier.

7.2 E-Class Amplifier

Class-E amplifier shown in Figure 7.3 is a power switching amplifier with high efficiency which is theoretically equal to 100%. The active switching device of e-class amplifier is a transistor. During RF period it aims to get as lower voltage-current product as possible. The desirable waveforms of transistor voltage and current are depicted in Figure 7.4.

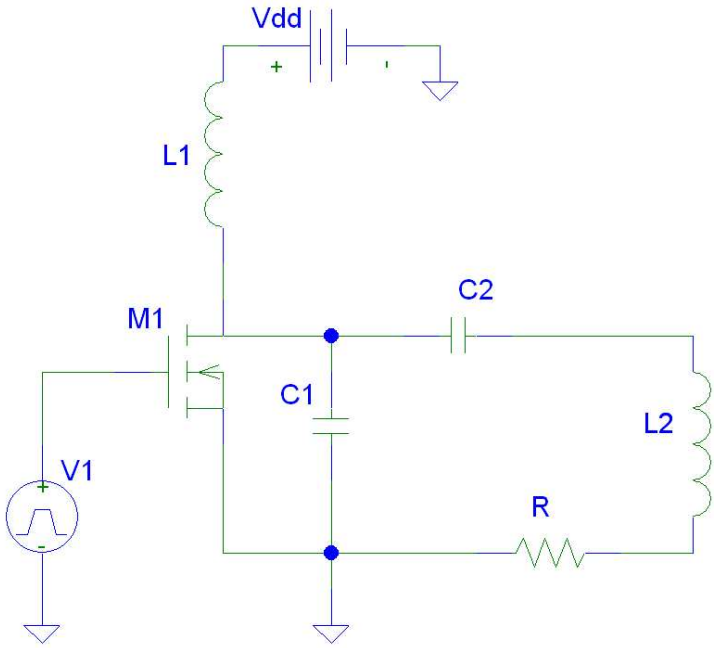

Figure 7.3: E-Class amplifier.

To tune the e-class amplifier is a non trivial task. However, [103] provides the electrical engineers with the detailed tuning instructions.

7.2.1 E-Class Amplifier Design

[103] can be considered as a perfect guidance to Class-E amplifier design and tuning. In the beginning of Class-E amplifier (see Figure 7.3) design we fixed several values:

$f = 10$ MHz, hence $\omega = 2\pi f = 62.8$ radian/s;

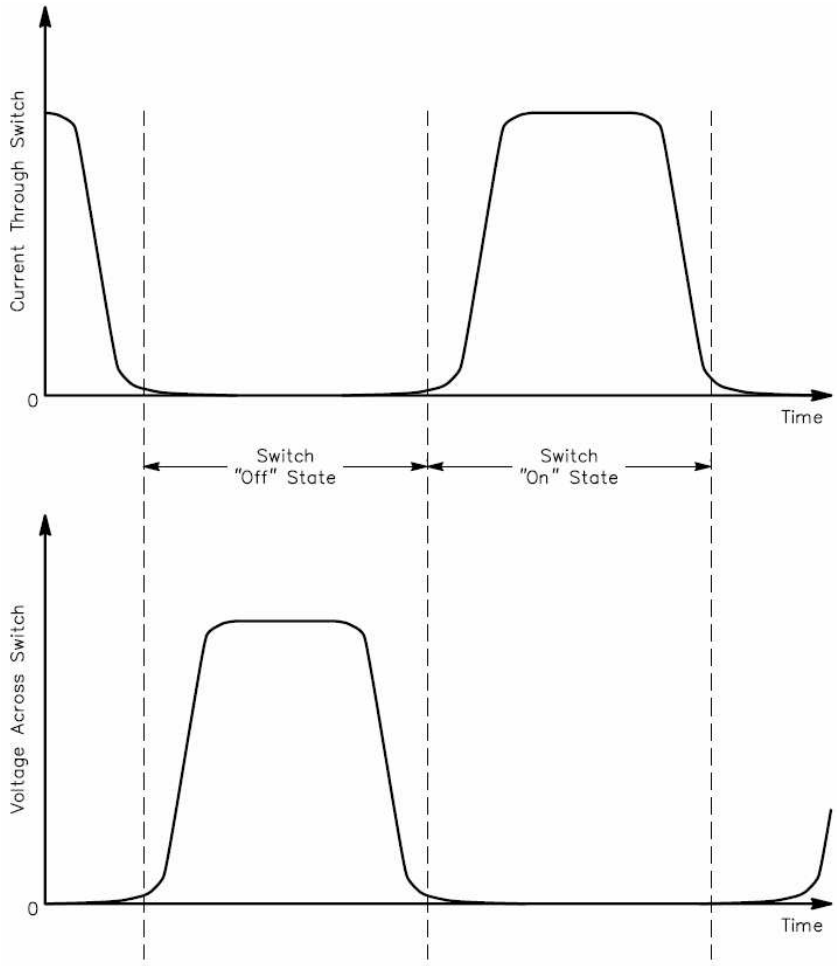

Figure 7.4: Desirable transistor waveforms of voltage and current in e-class amplifier [103].

$Q_L = 60$;

$L_1 = L_2 = 10\mu$H.

Applying equations (7.1)-(7.3) adopted from [103] we managed to calculate the residuary values of R, C_1, C_2 of Class-E amplifier represented in Figure 7.3.

$$L_2 = \frac{Q_L R}{2\pi f} \tag{7.1}$$

$$C_1 = \frac{1}{34.2219 f R}\left(0.99866 + \frac{0.91424}{Q_L} - \frac{1.03175}{Q_L^2}\right) + \frac{0.6}{(2\pi f)^2 L_1} \tag{7.2}$$

$$C_2 = \frac{1}{2\pi f R}\left(\frac{1}{Q_L - 0.104823}\right)\left(1.00121 + \frac{1.01468}{Q_L - 1.7879}\right) - \frac{0.2}{(2\pi f)^2 L_1} \tag{7.3}$$

However, [34] helps to reduce the equations (7.2) and (7.3) to simplified versions (7.4) and (7.5) respectively, where C_2 in (7.5) can be expressed via L_2:

$$C_1 = \frac{1}{2\pi f R} - C_{ds} \tag{7.4}$$

where C_{ds} is a drain to source capacitance of the transistor switch.

$$L_2 = \frac{1}{(2\pi f)^2 C_2} + \frac{1.152R}{2\pi f} \tag{7.5}$$

As an active transistor switch we have chosen a MOSFET IRF510.

Having calculated the necessary values we carried out the first experiment in Cadence environment. The voltage waveform at the time of transistor switching was far from ideal. The theoretical class-E amplifier voltage waveform is depicted in Figure 7.5, where V_d is a drain voltage and V_G is a gate voltage of the MOSFET transistor.

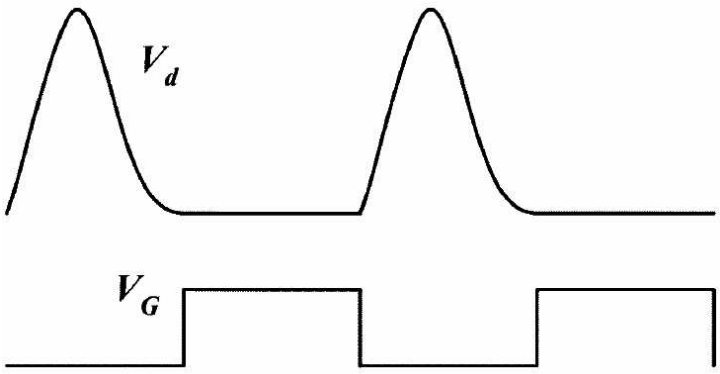

Figure 7.5: Class-E amplifier voltage [70].

To obtain voltage waveform as it is shown in Figure 7.5 we have applied Class-E amplifier tuning instructions from [103].

7.3 Inductive Power Link

Section 7.2 gives an idea how the Class-E amplifier operates and provides us with the design support. Figure 7.6 shows the "external" and "internal" circuits connected by inductive link. "External" circuit is assumed to be outside the skull and generates RF pulses. "Internal" circuit will be inside the skull, it will receive RF pulses and supply a probe.

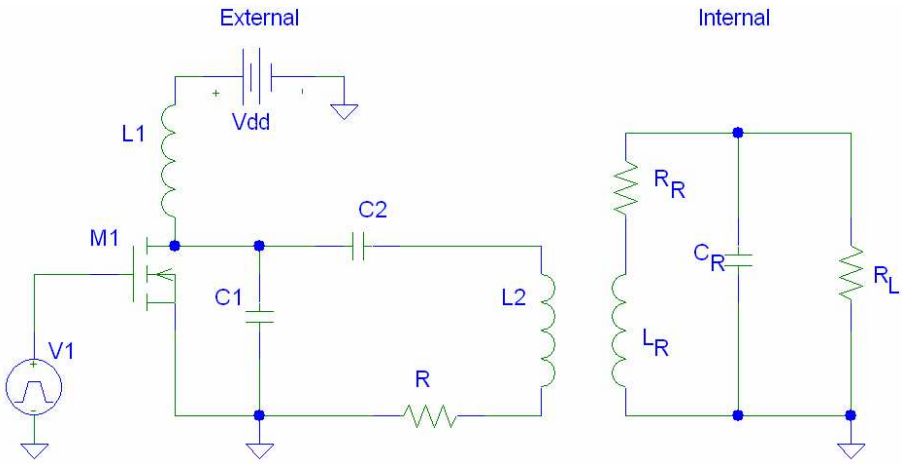

Figure 7.6: Inductive power link.

In accordance with the equivalent diagrams shown and described in details in Figure 1 of [62] we managed to simplify the circuit depicted in Figure 7.6 into the circuit depicted in Figure 7.7.

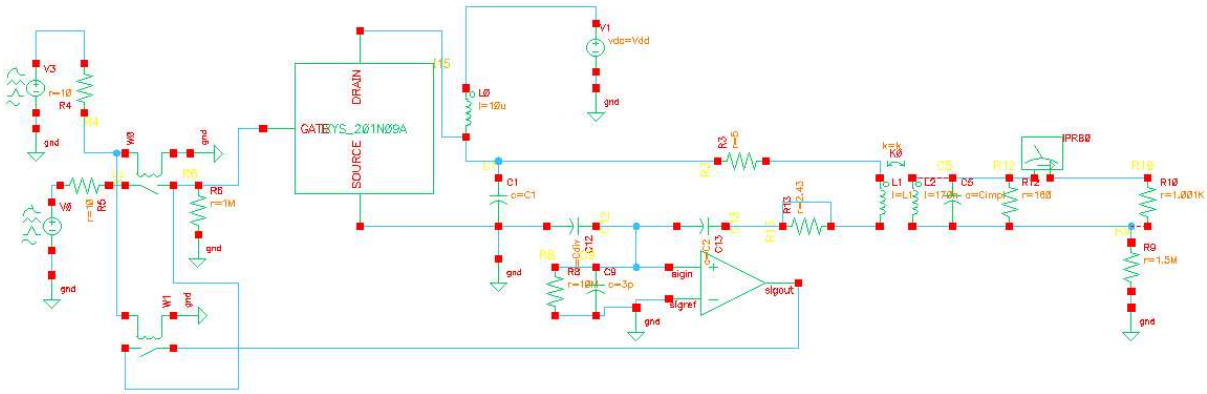

Figure 7.7: The final circuit in Cadence environment.

We can define the receive efficiency η_R of the receive ("internal") circuit using formula (7.6).

$$\eta_R = \frac{1}{1 + \frac{R_L}{Q_R^2 R_R}} \tag{7.6}$$

7.4 Receive Circuit vs Receive Circuit Transformed into Resistor

In this Section we describe the experiments carried out with the entire system (external+internal circuits) and the system where receive circuit transferred into a resistor added in external circuit.

According to [62] we have transformed the receive circuit into the equivalent resistor $R13$=2.43 Ohm (see Figures 7.7, 7.8). The goal of this section, actually, is to estimate the total efficiency of the entire system: with the normal receive circuit and with the transformed receive circuit.

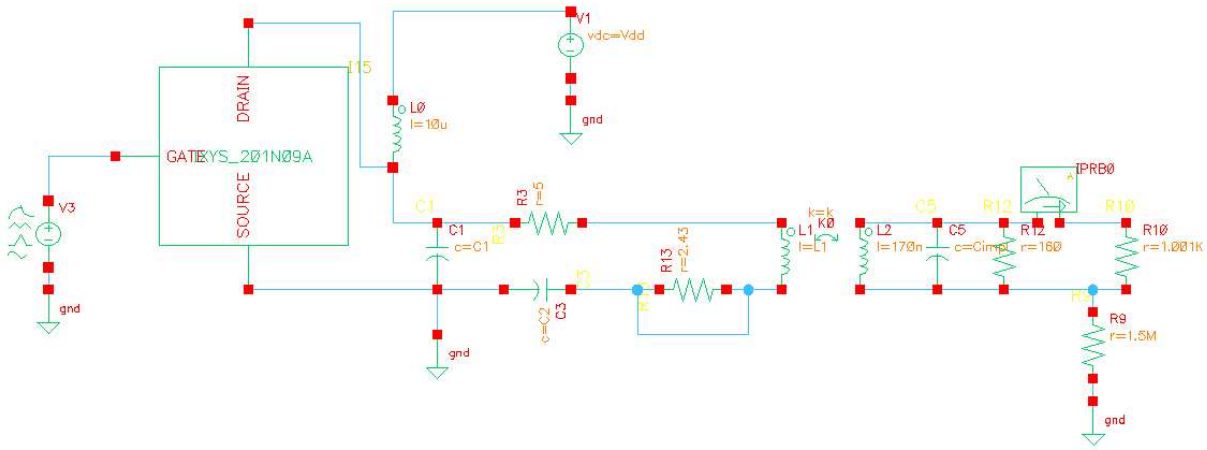

Figure 7.8: Inductive power link without the voltage feedback.

7.5 Voltage Feedback

[117] describes a self-oscillating class-E transmitter with an emphasis on voltage feedback scheme. Voltage feedback circuit carries out the following activity:

- it stabilizes the Class-E operation for various transmitter coil inductance values [117];
- it solves the problem of mismatch between the switching frequency of the driving device and the resonant frequency of the load network [117].

The main component of the feedback circuit is a comparator. Our choice is AD8611. However, it consumes much power. To supply the comparator a voltage divider is required. C_2 and C_{div} shown in Figure 7.7 represent the voltage divider which generates 5V to supply the comparator in the feedback circuit. The values of C_2 and C_{div} can be defined using formula (7.7) and Figure 7.9.

$$V_{out} = V_{in}\frac{C_2}{C_2 + C_{div}} \tag{7.7}$$

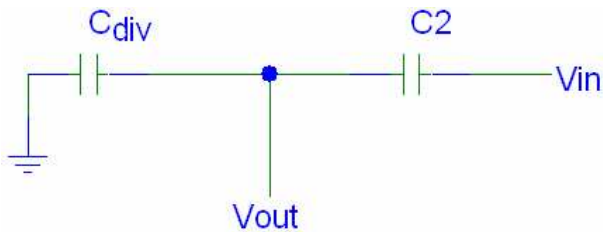

Figure 7.9: Capacitive voltage divider.

Due to high power consumption of comparator in the feedback circuit there is an idea to substitute it with the inverter built with discrete transistors. To do that it will be necessary to change the entire system polarity though.

7.6 Final Results

The final circuit is presented in Figure 7.7. In this version the active switching transistor has been changed from IRF510 to DE150-201N09A by IXYS which was designed specially for RF applications. The criterions for transistor search were: high switching frequency and low output capacitance.

The circuit values are:

$k = 0.01$;

$C_{impl} = 16.65$ nF ;

$C_{div} = 101$ pF ;

$L_0 = 10\ \mu H$;

$L_1 = 30\ \mu H$;

$f = 3.088$ MHz ;

V_{dd} = 3 V ;

trf = 4 ns ;

C_2 = 2.87 nF ;

C_1 = 760 pF ;

Voltage source parameters:

Source type: pulse ;

Zero value: 0 V ;

One value: 6 V ;

Period of waveform: $1/f$ s ;

Rise time=Fall time: trf s ;

Pulse width: $1/(2*f)$-trf s ;

The total efficiency (see Table 7.1) was determined as an average power dissipated on the load resistance in receive circuit divided into the average power generated by power supply. To get the efficiency in percentage we multiplied this result by 100.

Table 7.1 shows how the total efficiency depends on C_2 value. Figure 7.10 represents this dependence graphically. Thus, the adjustment of C_2 value on +28% changes the total efficiency on 7.8%. However, the adjustment of C_2 value on -28% changes the total efficiency on 18%.

The adjustment of C_1 on ±28% changes the total efficiency on 10.6%. So, the C_1 values in the range of 700-800 pF provide us with the stable efficiency of 2.8%.

Table 7.2 shows how L_1 (see Figure 7.7) value adjustments affect on the total efficiency.

Figure 7.11 presents the Class-E voltage waveform for the circuit depicted in Figure 7.7, and values (C2 = 2.87 nF, eff. = 2.83%).

This work was performed during my visit to UC Berkeley, USA, in the fall 2008.

C_2, **nF**	**Efficiency, %**
1.67	1.72
2.07	2.32
2.47	2.70
2.87	**2.83 Max.**
3.27	2.76
3.67	2.61
4.07	2.45
4.47	2.29

Table 7.1: The efficiency dependance on C_2 (Figure 7.7).

Value change, %	$\mathbf{L}_1$, μ**H**	**Efficiency, %**
±1	30.3	2.454
±1	29.7	2.452
±10	33.0	0.240
±10	27.0	0.190
±28	38.4	0.047
±28	21.6	0.016

Table 7.2: The efficiency dependance on L_1 (Figure 7.7).

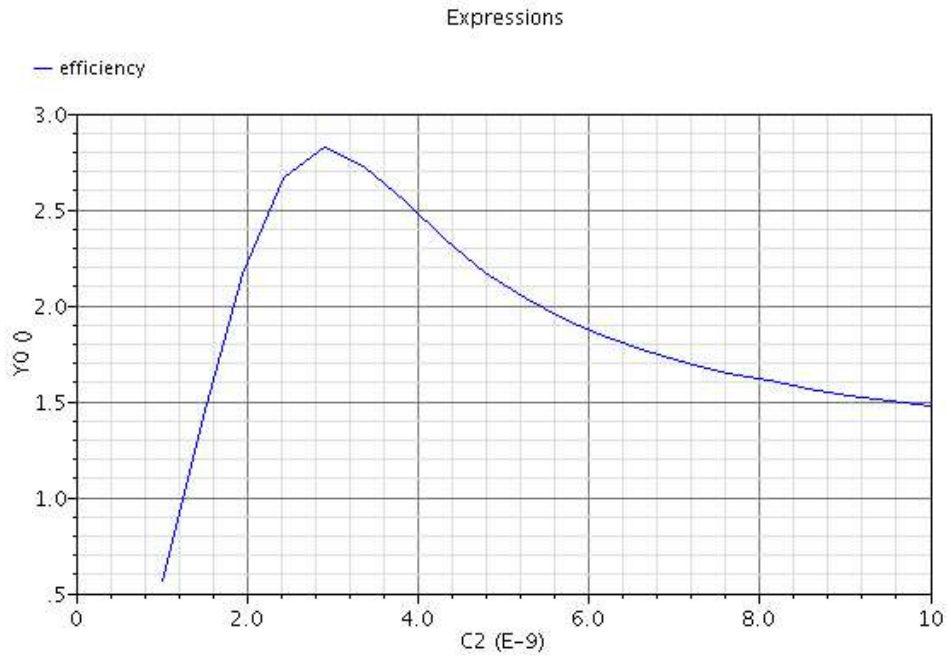

Figure 7.10: The efficiency dependance on C_2 (Figure 7.7).

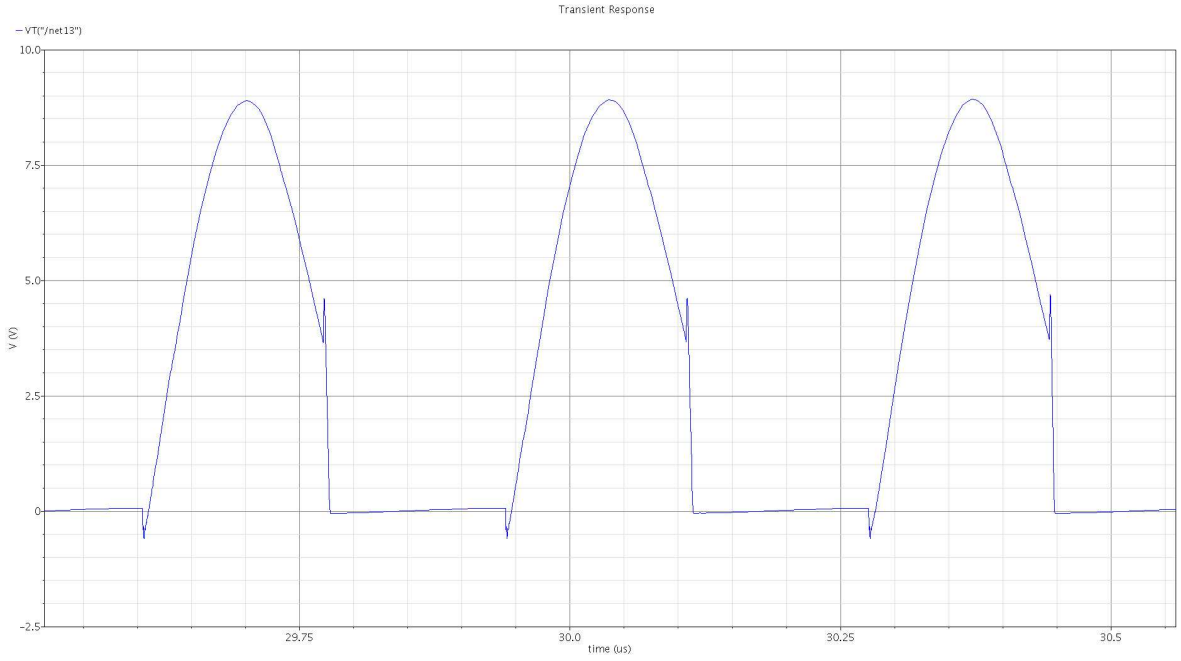

Figure 7.11: The Class-E voltage waveform.

Chapter 8

Conclusion

8.1 Summary

The field of wireless sensing holds the potential to make the way of human life better. By developing smaller and cheaper sensor devices and improving their technical parameters, we may significantly increase the range of their future applications. However, there are plenty of power and technological obstacles which have not been resolved yet.

In this doctorate thesis, we have addressed the lifetime problem of Wireless Sensor Network (WSN) and demonstrated how to achieve long-term operation of WSN starting from the design level. Due to WSN energy resources limitation, energy consumption is one of the most constraining requirement for design and implementation of WSNs. Thus, power analysis is a critical step in the sensor network development process. In this work we presented a flexible and extensible simulation framework to estimate power consumption of sensor network applications for arbitrary hardware (HW) platforms. This framework allows designers of sensor networks to estimate power consumption of the explored HW platform which permits the selection of an optimal HW solution and software (SW) implementation for the specific projects.

Besides, we discussed in the thesis how to exploit the WSN energy resources efficiently at HW and SW levels as well. At HW level we proposed a flexible architecture of energy scavenging module for a wireless sensor node. This module enables the energy harvesting of different kind (vibration, acoustic noise, solar radiation, temperature difference). Harvested energy can be stored in a primary energy buffer or a secondary one. The primary energy buffer is either the

supercapacitors in the package wired in series (EnTick platform) or the printed supercapacitor which is fabricated using "direct write" technology (TreBer platform) [29]. A rechargeable Li-ion battery serves as a secondary energy buffer. In case both buffers are fully charged the sensor node can be supplied from the ambient source directly.

The contribution of the thesis at software level refers to the communication part of the sensor node. Radio on a board is the most power consuming part. Thus, to decrease the average power dissipated by a WSN node we devised an analytical criterion which determines the value of the optimal node synchronization period where the node consumes less energy without undermining communication reliability.

At the end of the thesis, we try to prove that the research conducted during the PhD course will be useful for society or at least for one person as in case with the medical research. This research described our vision on the powering of wireless brain-machine interfaces. The final system which is still under construction will help handicapped persons to improve the quality of their life and feel themselves as full persons.

Two other application scenarios are connected with the implemented energy scavenging module. In the first application we apply the WSN paradigm to the entertainment area, and in particular to the domain of Paintball. The second one refers to a wireless sensor platform for monitoring of combustible, dangerous gases, and early fire detection. The platform operation is based on the pyrolysis product detection which makes it possible to prevent fire before inflammation.

8.2 Future Work

While the methodologies and solutions presented in this thesis achieve encouraging results, they are still not ideal and can be improved in the following directions.

As for the future work for the methodology of power consumption evaluation of an individual node of the WSN we plan to include simulation of heterogeneous networks by extending the presented methodology with additional network layer and provide ability to model and simulate a network of nodes. Furthermore, the customizable battery model will be implemented (using the SystemC language) as a part of the framework. It will allow quicker analysis of power consumption and lifetime estimation.

The next prototype of the energy scavenging module is going be with both printed capacitor and rechargeable battery. These technologies will help decrease the weight and volume of the module. Besides that, we intend to decrease the size of the powering board for wireless brain-machine interfaces from 5x5 mm to 1x1 mm device. However, this changes require additional research in limitations on wireless power transfer for human and precise modelling of the device.

The analytical criterion derived in Chapter 5 is quite sensitive to the absolute value of the relative frequency offset of the least accurate local oscillator (ν_{max}). Also, the frequency skew may differ considerably from node to node. To overcome this issue, nextgeneration synchronization algorithms should be able to estimate the value of ν_{max} at runtime and to adjust the duration of the optimal synchronization period accordingly.

Bibliography

[1] Atmel avr cpu. http://www.atmel.com/.

[2] Concentrated photovoltaic technology. www.solfocus.com/technology.html.

[3] Cooper bussmann. www.cooperbussmann.com.

[4] Eemb. www.eemb.com.

[5] Jlm innovation, zigsens platform. www.jlmi.de/products/zigsens, 2009.

[6] Maxim. www.maxim-ic.com.

[7] Micrel. www.micrel.com.

[8] National semiconductor. www.national.com.

[9] nesc. http://nescc.sourceforge.net/.

[10] NiMH battery specifiation. http://www.eemb.com/NI-MH/NI-CD/battery.html.

[11] Ns-2. http://www.isi.edu/nsnam/ns/.

[12] Oak ridge national laboratory. http://www.ornl.gov, 2009.

[13] Omnet++. http://www.omnetpp.org/.

[14] Otcl. http://bmrc.berkeley.edu/research/cmt/cmtdoc/otcl/.

[15] Perpetuum platform. http://www.perpetuum.co.uk.

[16] Prism solar technologies. www.prismsolar.com.

[17] Ptolemy ii. http://ptolemy.berkeley.edu/ptolemyII/.

[18] Sensor specification. www.figarosensor.com/products/2610pdf.pdf.

[19] Siemens solar panels. http://siemen.co.uk/qt2.html.

[20] Simpliciti protocol. http://focus.ti.com/docs/toolsw/folders/print/simpliciti.html.

[21] Tinyos. http://www.tinyos.net/.

[22] Visualsense. http://ptolemy.eecs.berkeley.edu/visualsense/.

[23] *ISO/IEC Guide 98:1995, Guide to the Expression of Uncertainty in Measurement*. Geneva, Switzerland, 1995.

[24] F. D. Barlow A. Eishabini-Riad. *Thin Film Technology Handbook*. McGraw Hill, 1998.

[25] A. Ageev, D.Macii, and A.Flammini. Towards an adaptive synchronization policy for wireles sensor networks. In *Proceedings of the International IEEE Symposium on Precision Clock Synchronization for Measurement, Control and Communication*, pages 115–120, Ann Arbor, Michigan, USA, September 2008.

[26] A. Ageev, D. Macii, and D. Petri. Synchronization uncertainty contributions in wireless sensor networks. In *International Instrumentation and Measurement Technology Conference (I2MTC)*, pages 1986–1991, Victoria, BC, Canada, May 2008.

[27] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *IEEE Communications Magazine*, 40(8):102–114, 2002.

[28] T. Antoine-Santoni, J. F. Santucci, E. De Gentili, and B. Costa. Wildfire impact on deterministic deployment of a wireless sensor network by a discrete event simulation. In *14th IEEE Mediterranean Electrotechnical Conference*, pages 204–209, Ajaccio, France, 2008.

[29] C. B. Arnold and A. Pique. Laser direct-write processing. *Mrs Bulletin*, 32:9–12, January 2007.

[30] Alvise Bonivento, Luca P. Carloni, and Alberto L. Sangiovanni-Vincentelli. Platform based design for wireless sensor networks. *MONET*, 11:469–485, 2006.

[31] S. Buller, M. Thele, R. W. D. Doncker, and E. Karden. Impedancebased simulation models of supercapacitors and li-ion batteries for power electronic applications. *IEEE transactions on Industrial Applications*, 41(3):742–747, May/June 2005.

[32] R. Burgan, R. Klaver, and J. Klaver. Fuel models and fire potential from satellite and surface observations. *International Journal of Wildland Fire*, 8:159–170, 1998.

[33] IEEE Standart C95.1-2005. *IEEE Standard for Safety Levels with Respect to Human Exposure to Radio Frequency Electromagnetic Fields, 3 kHz to 300 GHz.* IEEE, New York, USA, April 2006.

[34] W. H. Cantrell. Circuit to aid tuning of class-e amplifier. In *Proceedings of the IEEE MTT-S International Microwave Symposium Digest*, volume 2, pages 787–790, Boston, USA, September 2000.

[35] Luca P. Carloni, Fernando De Bernardinis, Alberto L. Sangiovanni-Vincentelli, and Marco Sgroi. The art and science of integrated systems design. In *Proceedings of the* 28^{th} *European Solid-State Circuits Conference, ESSCIRC 2002*, pages 19–30, Firenze, Italy, September 2002.

[36] Z. Chaczko and F. Ahmad. Wireless sensor network based system for fire endangered areas. In *3rd International Conference on Information Technology and Applications*, pages 203–207, Australia, 2005.

[37] Chih Min Chao and Yung Chang Chang. A power-efficient timing synchronization protocol for wireless sensor networks. *Journal of Information Science and Engineering*, 23(4):985–997, 2007.

[38] Y. H. Chee, M. Koplow, M. Mark, N. Pletcher, M. Seeman, F. Burghardt, D. Steingart, J. Rabaey, P. Wright, and S. Sanders. Picocube: A 1cm3 sensor node powered by harvested energy. In *Design Automation Conference*, pages 114–119, CA Anaheim, USA, June 2008.

[39] M. Chen and G. Rincon-Mora. Accurate electrical battery model capable of predicting runtime and i-v performance. In *IEEE Transactions on Energy Conversion*, volume 21, pages 504–511, June 2006.

[40] Elaine Cheong, Edward A. Lee, and Yang Zhao. Viptos: A graphical development and simulation environment fortinyos-based wireless sensor networks.

[41] C.-Y. Chong and S. P. Kumar. Sensor networks: Evolution, opportunities, and challenges. In *Proceedings of the IEEE*, pages 1247–1256, August 2003.

[42] B. Conway. *Electrochemical Supercapacitors*. Scientific Fundamentals and Technological Applications. Kluwer Academic: New York, 1999.

[43] Crossbow. Crossbow sensor modules: Mica2, micaz, telosb. http://www.xbow.com/Products, 2009.

[44] Cymbet. White paper: Zero power wireless sensors. http://www.cymbet.com, 2009.

[45] H. Dai and R. Han. Tsync: A lightweight bidirectional time synchronization service for wireless sensor networks. *ACM SIGMOBILE Mobile Computing and Communication Review*, 8:125–139, 2004.

[46] S. Davis. *Basics of Design: Battery Power Management*. Supplement to Electronic Design. 2004.

[47] D.W. Dennis, V. S. Battaglia, and A.Belanger. Electrochemical modeling of lithium polymer batteries. *J. Power Source*, 110(2):310–320, August 2002.

[48] D. M. Doolin and N. Sitar. Wireless sensors for wildfire monitoring. In *Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems*, San Diego CA, USA, 2005.

[49] N. J. Dudney. Solid-state thin-film rechargeable batteries. *Materials Science and Engineering*, 116:245–249, February 2005.

[50] N. J. Dudney, J. B. Bates, and B. J. Neudecker. Solid-state thin-film rechargeable batteries. *Encyclopedia of Materials: Science and Technology*, page Article 32, 2001.

[51] P. Dutta, M. Grimmer, A. Arora, S. Bibyk, and D. Culler. Design of a wireless sensor network platform for detecting rare, random and ephemeral events. In *Fourth International Symposium on Information Processing in Sensor Networks*, pages 497–502, CA Los Angeles, USA, April 2005.

[52] P. Dutta, J. Hui, J. Jeong, S. Kim, C. Sharp, J. Taneja, G. Tolle, K. Whitehouse, and D. Culler. Trio: Enabling sustainable and scalable outdoor wireless sensor network deployments. In *Fifthth International Conference on Information Processing in Sensor Network*, pages 407–415, Nashville, Tennessee, USA, 2006.

[53] J. Elson, L. Girod, and D. Estrin. Fine-grained network time synchronization using reference broadcasts. In *Fifth Symposium on Operating Systems Design and Implementation*, pages 147–163, Boston, Massachusetts, USA, 2002.

[54] J. Elson, L. Girod, and D. Estrin. Fine-grained network time synchronization using reference broadcasts. In *Proceedings of the Fifth Symposium on Operating Systems Design and Implementation (OSDI 2002)*, volume 36, pages 147–163, 2002.

[55] S. C. Ergen. Zigbee/ieee 802.15.4 summary. 2004.

[56] S. A. Forsyth, J. M. Pringle, and D. R. MacFarlane. Ionic liquids - an overview. *Australian Journal of Chemistry*, 57:113–119, 2004.

[57] S. Ganeriwal, R. Kumar, and M. Srivastava. Timing-sync protocol for sensor networks. In *ACM Sen-Sys*, pages 138–149, Los Angeles, California, USA, 2003.

[58] L. Gao, S. Liu, and R. A. Dougal. Dynamic lithium-ion battery model for system simulation. *IEEE Transactions Compon. Packag. Technol.*, 25(3):495–505, September 2002.

[59] Thorsten Grötker, Stan Liao, Grant Martin, and Stuart Swan. *System Design with SystemC*. Kluwer Academic Publishers, Norwell, MA, 2002.

[60] S. C. Hageman. Simple pspice models let you simulate common battery types. *Electronics Design, Strategy, News*, pages 17–132, October 1993.

[61] M. S. Halper and J. C. Ellenbogen. *Supercapacitors: A Brief Overview*. MITRE, McLean, Virginia, 2006.

[62] R. R. Harrison. Designing efficient inductive power links for implantable devices. In *IEEE International Symposium on Circuits and Systems*, pages 2080–2083, 2007.

[63] J. P. Hart and S. A. Wring. Recent developments in the design and application of screen-printed electrochemical sensors for biomedical, environmental and 115 industrial analyses. *TrAC Trends in Analytical Chemistry*, 16:89–103, 1997.

[64] C. C. Ho, D. A. Steingart, J. W. Evans, and P. K. Wright. Tailoring electrochemical capacitor energy storage using direct write dispenser printing. *ECS Transactions*, 16:35–47, October 2008.

[65] B. Hoffmann-Wellenhof, H. Lichtenegger, and J. Collins. *Global Positioning System. Theory and Practice*. Springer Wien New York, 5th edition, 2001.

[66] J. Hsu, S. Zahedi, A. Kansal, M. Srivastava, and V. Raghunathan. Adaptive duty cycling for energy harvesting systems. In *International Symposium on Low Power Electronics and Design*, pages 180–185, Tegernsee, Germany, October 2006.

[67] Dr. S.S. Iyengar. LSU sensorsimulator user manual, February 2006.

[68] X. Jiang, J. Polastre, and D. Culler. Perpetual environmentally powered sensor networks. In *Fourth International Symposium on Information Processing in Sensor Network*, pages 463–468, CA Los Angeles, USA, April 2005.

[69] X. Jiang, J. Taneja, J. Ortiz, A. Tavakoli, P. Dutta, J. Jeong, D. Culler, P. Levis, and S. Schenker. An architecture for energy management in wireless sensor networks. In *Special Issue on the Workshop on Wireless Sensor Network Architecture*, volume 4, pages 31–36, April 2007.

[70] G. A. Kendir, L. Wentai, G. Wang, M. Sivaprakasam, R. Bashirullah, M. S. Humayun, and J. D. Weiland. An optimal design methodology for inductive power link with class-e amplifier. In *IEEE transactions on Circuits and Systems*, volume 52, pages 857–866, 2005.

[71] H. A. Kiehne. *Battery Technology Handbook*. Marcel Dekker: New York, 2003.

[72] B. Krishnamachari. *Networking Wireless Sensors*. Cambridge University Press, New York, USA, 2005.

[73] H. Kuchling. *Handbook of physics [Russian translation]*. Mir, Moscow, 1982.

[74] B. Kusy, P. Dutta, P. Levis, M. Maroti, A. Ledeczi, and D. Culler. Elapsed time on arrival: a simple and versatile primitive for canonical time synchronization services. *International Journal of Ad Hoc and Ubiquitous Computing*, 1:239–251, 2006.

[75] M. A. Lebedev and M. A. L. Nicolelis. Brain-machine interfaces: past, present and future. *TRENDS in Neurosciences*, 29(9):536–546, 2006.

[76] B. Leblon. Monitoring forest fire danger with remote sensing. *Natural Hazards journal*, 35:343–359, 2005.

[77] A. Lewandowski and M. Galinski. Carbon - ionic liquid double-layer capacitors. *Journal of Physics and Chemistry of Solids*, 65:281–286, 2004.

[78] Y. Li, Z. Wang, and Y. Song. Wireless sensor network design for wildfire monitoring. In *6th World Congress on Intelligent Control and Automation*, pages 109–113, Dalian, China, 2006.

[79] D. Linden and T. B. Reddy. *Handbook of Batteries*. McGraw-Hill Professional: New York, 2002.

[80] F. Linnarsson, P. Cheng, and B. Oelmann. Sentio: A hardware platform for rapid prototyping of wireless sensor networks. In *The 32-nd Annual Conference of the IEEE Industrial Electronics Society*, pages 3002–3006, Paris, France, November 2006.

[81] D. Macii and D. Petri. An effective power consumption measurement procedure for bluetooth wireless modules. *IEEE Transactions on Instrumentation and Measurement*, 56(4):1355–1364, 2007.

[82] D. Macii, A. Zanotto, F. Leonardi, and D. Petri. A high-level model for estimating power consumption of bluetooth devices. In *Instrumentation and Measurement Technology Conference Proceedings*, pages 72–77, Victoria, British Columbia, Canada, May 2008.

[83] V. V. Malyshev and A. V. Pislyakov. Investigation of gas-sensitivity of sensor structures to carbon monoxide in a wide range of temperature, concentration and humidity of gas medium. *Sensors and Actuators B: Chemical*, 123:71–81, 2007.

[84] M. Maroti, B. Kusy, G. Simon, and A. Ledeczi. The flooding time synchronization protocol. In *Second ACM International Conference on Embedded Networked Sensor Systems*, pages 39–49, Baltimore, Maryland, USA, 2004.

[85] C. Moser, L. Thiele, D. Brunelli, and L. Benini. Adaptive power management in energy harvesting systems. In *Design, Automation and Test in Europe Conference and Exhibition*, pages 1–6, Nice, France, April 2007.

[86] S. Nakano, K. Yokosawa, Y. Goto, and K. Tsukada. Hydrogen gas detection system prototype with wireless sensor networks. In *IEEE Sensors*, pages 159–162, USA, 2005.

[87] G. Nazri and G. Pistoia. *Lithium Batteries: Science and Technology.* Kluwer Academic Publishers: Boston, 2004.

[88] OrCAD. *OrCAD PSpice A/D.* Online Manual, USA, October 1998.

[89] C. Park and P. H. Chou. Ambimax: Autonomous energy harvesting platform for multi-supply wireless sensor nodes. In *Third Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks*, pages 168–177, VA Reston, USA, September 2006.

[90] A. Phipps, F. Liu, L. Cattafesta, M. Sheplak, and T. Nishida. Demonstartion of a wireless, self-powered, electroacoustic liner system. *The Journal of the Acoustic Society of America*, 125(2):873–881, February 2009.

[91] Alessandro Pinto, Luca P. Carloni, and Alberto L. Sangiovanni-Vincentelli. Cosi: A framework for the design of interconnection networks. *Design & Test of Computers, IEEE*, 25(5):402–415, Sept.-Oct. 2008.

[92] V. Raghunathan and P. Chou. Design and power management of energy harvesting embedded systems. In *International Symposium on Low Power Electronics and Design*, pages 369–374, Tegernsee, Germany, 2006.

[93] V. Raghunathan, A. Kansal, J. Hsu, J. Friedman, and M. Srivastava. Design considerations for solar energy harvesting wireless embedded systems. In *Fourth International Symposium*

on Information Processing in Sensor Networks, pages 457–462, CA Los Angeles, USA, April 2005.

[94] D. Rakhmatov, S. Vrudhula, and D. A. Wallach. A model for battery lifetime analysis for organizing applications on a pocket computer. *IEEE transactions on VLSI Systems*, 11:1019–1030, December 2003.

[95] K. Romer. Time synchronization in ad-hoc networks. In *International Symposium on Mobile Ad Hoc Networking Computing*, pages 173–182, Long Beach, California, USA, 2001.

[96] K. Romer, P. Blum, and L. Meier. *Time Synchronization and Calibration in Wireless Sensor Networks*. Wiley and Sons, 2005.

[97] P. Rong and M. Pedram. An analytical model for predicting the remaining battery capacity of lithium-ion batteries. In *Proceedings Design, Automation, and Test in Europe Conference and Exhibition*, pages 1148–1149, Munich, Germany, March 2003.

[98] S. Roundy, P. Wright, and J. Rabaey. A study of low level vibrations as a power source for wireless sensor nodes. *Computer Communications*, 26(11):1131–1144, July 2003.

[99] S. Roundy, P. K. Wright, and J. M. Rabaey. *Energy Scavenging for Wireless Sensor Networks*. Kluwer Academic Publishers Group, 2003.

[100] Alberto L. Sangiovanni-Vincentelli. Defining platform-based design. *EEdesign*, February 2002.

[101] F. Simjee and P. H. Chou. Everlast: Long-life, supercapacitor-operated wireles sensor node. In *International Symposium on Low Power Electronics and Design*, pages 197–202, Tegernsee, Germany, October 2006.

[102] S. So and G. Wysocki. Ultra efficient laser spectroscopic trace-gas sensors for sensor networks and portable chemical analysis. In *Innovation Forum 2009*, Princeton school of engineering and applied science, USA, 2009.

[103] Nathan O. Sokal. Class-e rf power amplifiers. *QEX*, pages 9–20, 2001.

[104] A. Somov, V. Sachidananda, and R. Passerone. A self-powered module with localization and tracking system for paintball. *Lecture Notes In Computer Science; Proceedings of the 3rd International Workshop on Self-Organizing Systems*, 5343:182–193.

[105] L. Song and J. W. Evans. Electrochemical-thermal model of lithium polymer batteries. *J. Electrochemical Society*, 147:2086–2095, 2000.

[106] T. Starner. Human-powered wearable computing. *IBM System Journal*, 36:618–629, 1996.

[107] D. A. Steingart, C. C. Ho, J. Salminen, J. W. Evans, and P. K. Wright. Dispenser printing of solid polymer - ionic liquid electrolytes for lithium ion cells. In *6th International IEEE Conference on Polymers and Adhesives in Microelectronics and Photonics*, pages 261–264, Japan, 2007.

[108] SystemC. http://www.systemc.org.

[109] Ben L. Titzer and Daniel K. Lee. Avrora: Scalable sensor network simulation with precise timing.

[110] R. Torah, P. Glynne-Jones, M. Tudor, T. O'Donnell, S. Roy, and S. Beeby. Self-powered autonomous wireless sensor node using vibration energy harvesting. *Measurement Science and Technology*, 19(12):1–8, December 2008.

[111] A. A. Vasiliev, V. I. Filipov, I. M. Olikhov, and et al. Gas sensors for the detection of pyrolysis of combustible gases. In *8-th International Meeting on Chemical Sensors*, Switzerland, 2000.

[112] A. A. Vasiliev, S. Yu. Gogish-Klushin, D. Yu. Kharitonov, and et al. A novel approach to the micromachining sensors: the manufacturing of thin alumina membrane chips. In *Eurosensors XVI*, Czech republic, 2002.

[113] A.A. Vasiliev, R.G. Pavelko, S. Yu. Gogish-Klushin, D. Yu. Kharitonov, O.S. Gogish-Klushina, A.V. Sokolov, A.V. Pisliakov, and N.N. Samotaev. Alumina mems platform for impulse semiconductor and ir optic gas sensors. *Sensors and Actuators B: Chemical*, 32:216–223, 2008.

[114] A. Volokitina, M. Sofronov, and T. Sofronova. Topical scientific and practical issues of wildland fire problem. *Mitigation and Adaptation Strategies for Global Change journal*, 13:661–674, 2008.

[115] A. Wang, S-H. Cho, C.G. Sodini, and A.P. Chandrakasan. Energy-efficient modulation and mac for asymmetric microsensor systems. In *International Symposium on Low-Power Electronics and Design (ISPLED)*, pages 106–111, Huntington Beach, CA, USA, August 2001.

[116] P. Zhang, C. M. Sadler, S. A. Lyon, and M. Martonosi. Hardware design experience in zebranet. In *Second International Conference on Embedded Networked Sensor Systems*, pages 227–238, Baltimore, USA, 2004.

[117] B. Ziaie, S. C. Rose, M. D. Nardin, and K. Najafi. A self-oscillating detuning-insensitive class-e transmitter for implantable microsystems. In *IEEE Transactions on Biomedical Engineering*, volume 48, pages 397–400, 2001.