Thesis for the Degree of Licentiate
Sundsvall 2011

Enabling Autonomous Environmental Measurement Systems with Low-Power Wireless Sensor Networks

Sebastian Bader

Supervisor: Professor Bengt Oelmann

Electronics Design Division, in the
Department of Information Technology and Media
Mid Sweden University, SE-851 70 Sundsvall, Sweden

ISSN 1652-8948
Mid Sweden University Licentiate Thesis 54

ISBN 978-91-86694-14-2

Mittuniversitetet
MID SWEDEN UNIVERSITY

Akademisk avhandling som med tillstånd av Mittuniversitetet i Sundsvall framläggs till offentlig för avläggande av licentiatexamen i elektronik **måndagen den 10 januari 2011**, klockan **13.00** i sal **O111**, Mittuniversitetet Sundsvall. Seminariet kommer att hållas på engelska.

Enabling Autonomous Environmental Measurement Systems with Low-Power Wireless Sensor Networks

Sebastian Bader

Electronics Design Division, in the
Department of Information Technology and Media
Mid Sweden University, SE-851 70 Sundsvall
Sweden

Telephone: +46 (0)60 148495

Printed by Kopieringen Mittuniversitetet, Sundsvall, Sweden, 2011

Abstract

Wireless Sensor Networks appear as a technology, which provides the basis for a broad field of applications, drawing interest in various areas. On the one hand, they appear to allow the next step in computer networks, building large collections of simple objects, exchanging information with respect to their environment or their own state. On the other hand, their ability to sense and communicate without a fixed physical infrastructure makes them an attractive technology to be used for measurement systems.

Although the interest in Wireless Sensor Network research is increasing, and new concepts and applications are being demonstrated, several fundamental issues remain unsolved. While many of these issues do not require to be solved for proof-of-concept designs, they are important issues to be addressed when referring to the long-term operation of these systems. One of these issues is the system's lifetime, which relates to the lifetime of the nodes, upon which the system is composed.

This thesis focuses on node lifetime extension based on energy management. While some constraints and results might hold true from a more general perspective, the main application target involves environmental measurement systems based on Wireless Sensor Networks. Lifetime extension possibilities, which are the result of application characteristics, by (i) reducing energy consumption and (ii) utilizing energy harvesting are to be presented. For energy consumption, we show how precise task scheduling due to node synchronization, combined with methods such as duty cycling and power domains, can optimize the overall energy use. With reference to the energy supply, the focus lies on solar-based solutions with special attention placed on their feasibility at locations with limited solar radiation. Further dimensioning of these systems is addressed.

It will be shown, that for the presented application scenarios, near-perpetual node lifetime can be obtained. This is achieved by focusing on efficient resource usage and by means of a carefully designed energy supply.

Acknowledgements

There are many people who, directly and indirectly, have helped create this thesis and contributed in the work it stands for. I hereby want to thank all of them, and apologize that I will not be able to name every one of them.

First of all, I want to thank my supervisor Bengt Oelmann. I am very grateful to be your student, to learn from your insight and for all our discussions and talks. Thank you for giving me the freedom to choose what I like, but also the guidance for what is worth pursuing. I enjoy our meetings, from many of which I learned to question my own point of views.

I further would like to thank my colleagues at the Electronics Design Division for technical and organizational support, as well as occasional distractions. Cheng, thank you for infecting me with your fascination for sensors and embedded systems, as well as for never doing something half-way. David, thank you for your fika-support and the occasional evening out. Stefan, thank you for sharing and listening when organizational frustrations come up.

Moreover, I want to thank my family and friends for every bit of support, understanding and distraction. I am glad to be able to say that my life feels balanced because of you.

Finally, I would like to thank Solène. I am thankful about every moment we spend together. Thank you for being there for me and for all your love. I am glad you keep me from working at some time, but you understand to let me work at others.

Acronyms

CCD	Charge-Coupled Device
MEMS	Microelectromechanical System
GSM	Global System for Mobile Communications
WSN	Wireless Sensor Network
EM	Environmental Monitoring
ESN	Environmental Sensor Network
GPRS	General Packet Radio Service
WLAN	Wireless Local Area Network
RF	Radio Frequency
GDI	Great-Duck-Island
WAN	Wide Area Network
FPGA	Field Programmable Gate Array
DSP	Digital Signal Processor
ISM	Industrial, Scientific and Medical
LDO	Low Dropout
NDIR	Nondispersive Infrared
IC	Integrated Circuit
TDMA	Time Division Multiple Access

RTS	Request-to-Send
CTS	Clear-to-Send
TPSN	Timing-Sync Protocol for Sensor Networks
RBS	Reference Broadcast Synchronization
FTSP	Flooding Time Synchronization Protocol
LTS	Lightweight Time Synchronization
PBS	Pairwise Broadcast Synchronization
TDP	Time Diffusion Protocol
SFD	Start Frame Delimiter
RTC	Real-Time Clock
EH	Energy Harvesting
DLC	Double-Layer Capacitor
Li-Ion	Lithium-Ion
MPP	Maximum Power Point
MPPT	Maximum Power Point Tracking
PnO	Perturb-and-Observe
MOSFET	Metal-Oxide-Semiconductor Field-Effect Transistor
EPR	Equivalent Parallel Resistance

Contents

List of Figures

List of Tables

List of Papers

This thesis is mainly based on the following publications:

Paper I **Adpative Synchronization for Duty Cycling in Environmental Wireless Sensor Networks**
Sebastian Bader, Bengt Oelmann
2009 International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)

Paper II **Enabling Battery-Less Wireless Sensor Operation Using Solar Energy Harvesting at Locations with Limited Solar Radiation**
Sebastian Bader, Bengt Oelmann
2010 Fourth International Conference on Sensor Technologies and Applications

Paper III **A Method for Dimensioning Micro-Scale Solar Energy Harvesting Systems Based on Energy Level Simulations**
Sebastian Bader, Torsten Schölzel, Bengt Oelmann
Accepted for publication in *2010 International Conference on Embedded and Ubiquitous Computing (EUC)*

1 Introduction

Sensors are an omnipresent technology, which have the ability to augment human senses by converting physical values into simpler perceivable and processable signals. However, most of the present day sensors are deeply embedded in everyday objects and it is often the case that people are not even aware of their existence, as the manner in which they are used evolves. While it was the case previously that sensors were used as a direct interface between physical world and human perception (see Fig. 1.1b), today sensor data is more often combined and processed, introducing additional abstraction steps to the chain as depicted in Figs. 1.1c and d. A common everyday-example is the mobile phone, where the typical user is not interested in the pure data of microphone, CCD camera, MEMS accelerometer, the GSM-modem itself, and many more, but in the functions that these sensors provide, making the use of the device more efficient, comfortable or exciting.

Several computer science visions go even further, introducing ideas such as ubiquitous or pervasive computing [1, 2], smart dust [3] or the *Internet of things*. All of these have a common basis which involves the implementation of a digital interface to the real-world, thus creating responsiveness in everyday objects. This interface is usually supposed to be built of small and inexpensive computers, capable of communicating with each other and which have the ability to sense their ambient environment.

These scenarios have had a tremendous impact on the development of a new networking concept, consisting of distributed embedded systems which are generally referred to as Wireless Sensor Network (WSN). Based on the advantages the WSN-concept brings to a vast amount of different applications, interest in the corresponding technology is high. Ideally, a WSN allows for the deployment of a large amount of sensor nodes, which configure themselves, depending on the network topology and neighborhood situation. After sensing their physical environment and processing the obtained data locally, nodes communicate their data (or an extract) towards a network sink, where

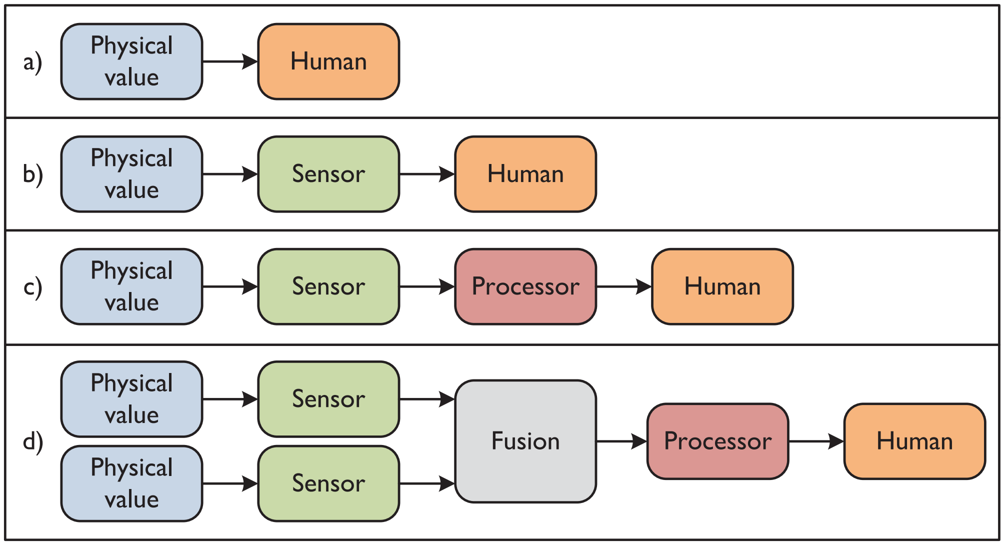


Figure 1.1: Evolution of sensing

data is further processed and made available for readout. As transmitted data should find the *best* route towards its destination automatically, the network can be remotely controlled and therefore be handled as one large measurement instrument.

Typical application areas for WSNs include industrial control systems, motion sensing, as well as the monitoring of structures or environments. Depending on the application challenges and constraints, WSNs can adopt different forms, use different technologies and communicate through different network topologies, making the design of WSNs highly application-specific. However, the necessity to involve energy conservation due to limited energy resources, is an issue in almost all application-areas [4].

1.1 Environmental Monitoring

Although EM can mean the monitoring of any kind of environment, it is most often defined as the observation and study of natural environments. The foundation of EM is the collection of data, which enables a better understanding of our natural surrounding to be gained by means of observation. Scientifically, Environmental Monitoring includes the fields of Physics, Chemistry and Biology. However, as more technologies, especially for data acquisition, become involved, so do the number of technical fields of study.

The motivation based on the ever increasing world population, means that Environmental Monitoring is not limited to the understanding of environments, but also includes monitoring for preservation reasons. EM plays a key-role to show the effects of human behavior on the environment and to disclose its limits. Typical applications, in addition to purely environmental science purposes, include the protection of water supplies, radioactive waste treatment, air pollution monitoring, natural resource protection, weather forecasting and enumeration and monitoring of species [5].

Environmental Monitoring strives to determine the status of a changing environment by analyzing a representative sample of the environment. As such, data acquisition forms a major part of EM. The data acquisition system in use has to allow for the collection of representative samples, which includes concerns such as the intrusiveness of the measurement system itself, sampling accuracy or sample storage. The impact of these concerns depends, on the one hand, on the application (e.g., the sensitivity of the observed physical value to be externally influenced) and, on the other hand, on the type of data acquisition system used.

Typical sampling techniques are grab samples, (remote) sampling stations and remote sensing. Grab sampling is the manual removal of a sample from the environment for further analysis. While this is not performed as frequently as was the case previously due to technological progress, in some cases, grab sampling is still used to allow for random sampling or more complex analysis than is possible in the field. One immense drawback of grab sampling is the involvement of human beings, which leads to time and cost issues in addition to high invasiveness. Sampling stations refer to sensor systems deployed in the environment of interest, monitoring the surrounding continuously or in defined intervals. These systems store the measurement samples taken locally or transfer them by means of GSM or satellite communication, thus significantly reducing the amount of human labor. Remote sensing usually means the sensing of environments from a distance by satellite or aircraft, involving imagery or radiation detection. Because of this distance application, remote sensing allows for the covering of large areas and the monitoring of inaccessible or dangerous environments, but this does, however, usually lack local resolution.

1.2 WSN for Environmental Monitoring

EM-applications are based on the development from data to information to knowledge [5]. Hence, the more meaningful data is obtained, the more knowledge that can be derived. Because data that is gathered through measurement and observation, the measurement system capabilities of WSNs offer several advantages to the field of Environmental Monitoring. Probably the most fundamental is the autonomy of data aggregation. While traditional sampling methods demand increased labor input for larger amounts of samples (e.g., sampling at several locations in the same area), an ideal WSN observes the environment at multiple locations and automatically transmits the data to a gathering point via the networked infrastructure. Furthermore, the autonomous sampling allows for the unobtrusive observation of phenomena and for monitoring in harsh locations and under extreme conditions [6]. Because the sensing networks are usually directly connected to the operator via the Internet or some type of local connection, data is gathered in real-time or near-real-time. This enables problems to be detected at an earlier stage than in systems with local storage and manual downloading at the end of an acquisition period. In addition, the remote connection to the sensor network means removal of distance between scientist and the monitored site [6], as the researcher can directly observe what is happening at a particular area of interest.

Nevertheless, Environmental Monitoring is an extensive area and different applications impose different requirements on the Wireless Sensor Network. A very useful classification of WSN-applications is made by Barrenetxea et al., who divide them into *time-driven*, *event-driven* and *query-driven* sensor networks [7]. However, as sensing in most applications is time-driven (e.g., by continuous or periodically sampling of the attached sensing devices), the classification mainly describes the network activity in the system. Within these, time-driven applications usually transmit their sensor readings periodically, which is typically used in data gathering applications, such as [8, 9]. However, event-driven sensor networks attempt to minimize the actual ongoing traffic and the flooding of the gathering point with meaningless information. These types of systems observe the area of interest, transmitting sensor information only on those occasions when particular events occur, such as a fire [10] or volcanic eruption [11]. Query-driven systems store gathered information locally and communicate it on request. This type of sensor network can for example be useful in logistics or home applications, but is not very common in applications of Environmental Monitoring.

System requirements of the different application classes differ tremendously. While it is the case that time-driven sensor networks are usually more organized, especially in terms of network traffic, event-driven applications behave more randomly and in an unforeseeable manner. Because of these differences, it is not usual system designs, particularly in communication protocols, for both classes to be interchangeable. The work presented in this document targets time-driven applications in Environmental Monitoring. A typical application scenario, including network architecture and system challenges is described below.

1.2.1 Application Scenario under Scope

The application scenario which is the focus of this work involves data gathering environmental sensor networks. These applications are manifold in the field of Environmental Monitoring and are usually used to aggregate as much data as possible. Examples include studies of water quality [12], microclimates of glaciers [7, 13] or rainforests [8], as well as the measurement of light intensity under shrubs [14]. In turn, the data thus collected acts as the input in order to provide a better understanding, modeling or site status observation. Furthermore, the applications dealt with in this case are time-driven applications, where physical values are measured and gathered continuously (or semi-continuously). As opposed to the event-driven sensor networks, time-driven systems possess only limited data processing within the sensor node units. While event-driven applications communicate the occurrence of events, and therefore must translate data into information, time-driven applications typically transfer pure sensor data.

As a result of this, typical wireless sensor nodes in the network can be rather simple devices, thus reducing system complexity, size and cost. Additionally, large amounts of these simple devices are necessary in order to cover large-scale areas, while maintaining required spatial resolution. The number of nodes opens up several challenges which must be addressed in these applications, such as the handling and visualization of data, communication and configuration protocols, as well as deployment and maintenance

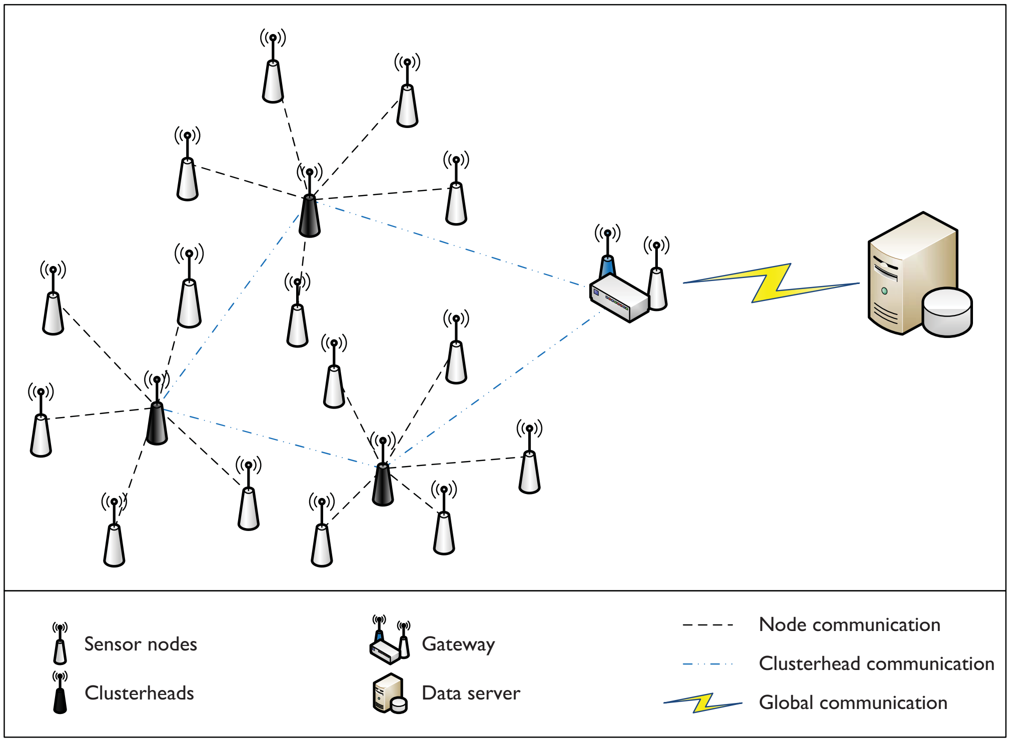


Figure 1.2: Typical network architecture of environmental monitoring applications under scope

of the network. Further details with reference to these research challenges will be given in section 1.4.

Fig. 1.2 shows a possible network architecture for data gathering WSNs in Environmental Monitoring, and the type of network architecture we address in this work. As many Environmental Sensor Networks (ESNs) are deployed in remote locations (or at least at a distance from the operator), a distinction between local and global domain is made. The local sensor field contains the sensor nodes, which are monitoring the deployment site, while also being connected to the global domain via some kind of gateway. The network topology in the sensor field is a hybrid between a star and mesh network, as depicted in Fig. 1.2. It will be further referenced to as a cluster-star-network.

Consequently, another distinction is made between normal sensor nodes and clusterheads. While normal sensor nodes are communicating with *their* clusterhead only (i.e., operating in a single-hop star-cluster), clusterheads also communicate with each other in a multi-hop-fashion to route data towards the gateway. This structure allows the majority of devices in the network to be simple, and only a few nodes are required to be more *intelligent*, thus requiring more resources.

Nevertheless, this topology structure does contain some disadvantages which are related to the node-task-distinction. Although usually not *single-point-of-failures*, the malfunctioning of a clusterhead can result in limited functionality within the sensor field. While some sensor nodes, initially connected to the malfunctioning clusterhead, might reconnect to another cluster, it is particularly the case that the border-nodes (i.e., sensor nodes at the outer border of the sensor field) might be out of communication range to any other clusterhead. Further on, the malfunctioning clusterhead might interrupt routes from other clusterheads to a gateway, hence leading to the separation of the sensor field. Some of these disadvantages might be overcome by proper deployment planning, such as redundancy in communication routes or backup-clusterheads.

The global domain usually consists of a data processing and storage unit, such as a data server. From here, data is converted into a representable form and made available to the operator of the network and environmental researchers, typically via inter-/intranet. The link between the sensor field and global domain is usually established via gateways using GSM/GPRS or Wireless Local Area Network (WLAN) capabilities. Additionally, in some cases special long-distance RF-links are implemented [13, 11].

1.2.2 Related System Examples

There have been several wireless sensor systems in the literature, which have been designed for and used in Environmental Monitoring. A subset of these systems - chosen for their impact on the community and which bear a close similarity to the targeted application in this work - are presented below. Additionally, in Table 1.1, a summarizing overview of these and additional systems is given, which provides, as fas as is possible, an acceptable comparison.

Great-Duck-Island: One of the first large-scale applications of WSN in Environmental Monitoring was the implementation of a WSN on Great-Duck-Island (GDI) [15, 16]. The system was designed and deployed in cooperation

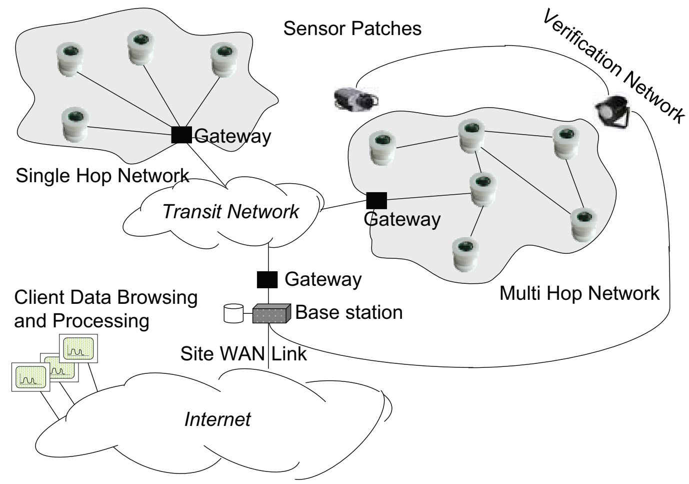


Figure 1.3: System architecture of the Wireless Sensor Network deployed on Great-Duck-Island in 2003 [15]

with both technology and biology researchers. Its main purpose was the monitoring of seabird nesting, which included the monitoring of animal behavior and the nesting environment conditions.

The motivation behind the use of the Wireless Sensor Networks for this application was twofold. The biologists involved valued the possibility to monitor the environment unobtrusively, while, for the technological researchers involved, they saw an opportunity to analyze the vision of WSNs in a real-world application.

The presented system was organized in a tiered architecture (see Fig. 1.3), consisting of sensor nodes deployed in patches, communicating via a local gateway through a transit network with the base station. This, in turn, was WAN-enabled and therefore allowed remote data collection and network management. Mainwaring et al. conducted two deployment runs, an initial deployment in the summer of 2002 [16], and a deployment of an updated system one year later [15]. During these deployments many experiences were made, and an extensive list of *lessons-learned* and *issues-to-be-solved* were provided in their publications.

GlacsWeb: A WSN-based system for the monitoring of glaciers is targeted in the GlacsWeb project [13]. The system is used as a tool to support the understanding and modeling of the behavior of glaciers. The main requirements included non-intrusive, automated, robust and long-term operation which involved reasonable costs.

The system architecture of GlacsWeb is quite similar to the architecture deployed on GDI, as it utilizes hierarchical layers. While sensor nodes are deployed on and in the glacier, base stations manage the nodes and relay their data towards the reference station. The reference station connects the sensor network with the researchers in Southampton. There are a variety of challenges and requirements for the different layers. While the main challenges for the sensor nodes are the packaging, power consumption and size, gateway issues relate more to the computation and communication reliability.

The project members conducted several deployments on glaciers, testing the developed system in the real world. Since the initially presented deployment in Norway in 2003 [13], from which the first results on system behavior and performance were gained, the system has been further developed and updated systems have been deployed in Norway and Iceland [17, 18]. Many issues regarding sensor network design and deployment, particularly those relating to extreme deployment sites, are presented and solutions are given in the related literature.

Volcano Monitoring: A different type of extreme environment is targeted in the volcano monitoring system presented in [19, 11]. In this application WSNs are equipped with low-frequency acoustic sensors to monitor volcanic activity.

While traditional systems involved local storage of data, which thus required a manual collection of the sampled data for further processing, the WSN-based system allows real-time monitoring of the activity over wireless links. In addition to the continuous monitoring of the volcanic activity, the researchers implemented an event-detection mechanism to reduce the amount of data which had to be communicated and processed.

The system has been deployed several times on active volcanoes. From initial proof-of-concept in 2004, where the system has been compared to an existing wired sensor-array, system improvements have been evaluated in 2005 and 2007.

While demanding different (i.e., particularly higher) resource requirements due to the nature of the application, the system presented in these works addressed signal processing problems concerning event-detection and sensor sample quality, that have not been previously targeted.

Luster: A Wireless Sensor Network for environmental research, called LUSTER, is presented in [14]. The motivation behind this system was based on a specific application, but it was designed in a more general fashion, thus allowing reuse and extension.

The application being looked at was the investigation of light conditions under shrub thickets. This leads to some application-specific design challenges. Nevertheless, several general problems were addressed, mainly the targeting of an autonomous and reliable operation in remote and harsh environments.

LUSTER includes solutions to problems, such as distributed network storage and in-deployment network observation. While targeting problems due to intermittent or faulty communications with the former solution, the latter allows for the validation of the network operation at runtime. Due to network-wide storage, the user can request missing data from the storage nodes or the same data can be used for error evaluation after collection. To allow a runtime validation of the network, Selavo et al. developed a tool, named SeeDTV [20].

SensorScope: The SensorScope project [7, 9] is another example of large-scale Environmental Monitoring applications. The system developed in this project is a flexible Wireless Sensor Network with high quality sensing stations, targeted to replace traditional sensing stations, which are costly and large in size.

Typical application scenarios for this system include large-scale data collection with high spatial resolution for modeling, prediction and risk assessment. In the project, multiple low-cost stations with limited accuracy are preferred over single high-accuracy, but costly stations can only provide data of limited resolution.

Since the initial test deployment in 2006, the system has been developed further and deployed several times. Lessons which have been learned from the design of the system and the different deployment runs, including harsh and remote environments, have been shared with the community in related publications.

1.3 Lifetime of WSNs

Although application constraints and therefore system design parameters do vary considerably between the EM-systems presented previously, the desire for a long system lifetime is a common goal. While the lifetime of Wireless Sensor Networks is usually a design parameter in all targeted applications, it is of particular importance in Environmental Monitoring. The reason for this is deeply embedded in the advantages that Wireless Sensor Networks provide in this application area.

As described in section 1.2, WSNs enable autonomous sensing on a large scale. Ideally, a large number of sensor nodes are spread over a wide area (possibly at a remote or harsh location), organizing themselves and communicating their sensor data back to the researcher without further attendance. A key parameter for allowing the nodes to fulfill this task is that there must always be a sufficient energy supply. If there is a requirement to regularly re-enter the deployment site in order to exchange or recharge the batteries the this defeats the object with regards to the advantage initially offered by the system.

Because of the importance of the system lifetime in the final application, lifetime has been used extensively as a evaluation parameter in the designs at all levels in sensor networks. However, providing a definition for the WSN-lifetime at a general level is very difficult. Several definitions have been proposed in the literature and a comprehensive coverage of these definitions (or definition-classes) is given in [22].

1.3.1 Definitions and Influencing Parameters

When discussing the lifetime of sensor networks, it is important to differentiate between the lifetime of the sensor nodes and the lifetime of the entire sensing system. While the two lifetime parameters are connected to each other, they have to be targeted on different levels.

Name	Year	Nodes	Size	Sensors [a]	Energy Source	Lifetime/Duration	Literature
GDI	2002 2003	32 150	large	l, t, h, p, pir	battery	days – month	[16, 15]
Glacsweb	2003-2005 2008	9 12	medium	p, ti, t p, ti, t, c, r, st, h	battery	days – years	[13, 17, 18]
Redwood Tree	2004	33	small	t, h, l	battery	44 days	[21]
Volcano	2004 2005	3 16	small large	is is, s	battery	days weeks	[19, 11]
Sensorscope	2006-2010	6 – 97	large	h, t, m, sr, pr, w	solar + battery	days – years	[7, 9]
Springbrook	2008	10	large	w, t, h, m, lw	solar + battery		[8]
Luster		19	medium	l	solar + battery	hours – weeks	[14]

Table 1.1: Overview of deployed Environmental Monitoring systems based on Wireless Sensor Networks

[a] l - light; t - temperature; h - humidity; p - pressure; pir - passive infrared; ti - tilt; c - conductivity; r - reflectivity; st - strain; is - infra sound; s - seismic; m - soil moisture; w - wind speed+direction; lw - leaf wetness; sr - solar radiation; pr - precipitation

Sensor node liveliness can usually be determined directly from the operational status of a sensor node at a given time. Typically, the lifetime of a sensor node is defined by its energy reservoir and its energy consumption over time. Limited energy resources are thus usually the only limiting factor for the nodes lifetime (ignoring hardware defects at this point). Nevertheless, operational interruptions can certainly occur during the lifetime of a sensor node, which might require additional consideration.

In comparison, Wireless Sensor Network lifetime is not as easy to determine without taking more application constraints into account. Typical definitions are based on the availability of sensor nodes, the coverage of the monitored terrain or the connectivity of the network. In addition, combinations and extensions of these factors might also be placed into the definition of a sensor network lifetime [22]. Since sensor nodes are the building blocks of Wireless Sensor Networks, the lifetime of the two are connected, but to what extent they are connected can vary from application to application.

Different scenarios regarding the WSN lifetime definition based on alive sensor nodes include the *n-of-n*, *k-of-n* or *m-in-k-of-n* scenarios. The first two cases are straight-forward solutions, where *n-of-n* means all nodes have to be alive, while in *k-of-n* at least *k*-nodes have to be alive. If the case is not met because more nodes are dead than are allowed, then the entire sensor network is considered as being dead. However, one can easily distinguish situations, in which the malfunctioning of one node will have a different affect on the network. An example is given in Figure 1.4, where the node-death in a) obviously affects overall sensor network operation to a lesser extent than that of b). This is not handled in the above mentioned definitions.

In an *m-in-k-of-n* scenario this shortcoming is addressed by introducing different classes of nodes. This case defines critical (m) and non-critical (k) nodes in the network. In addition, the sensor network lifetime is defined as the time period where *m* critical and *k* non-critical nodes of the overall *n* nodes are still alive. While the introduction of different classifications of nodes allows for improvement, aspects exist which are still not dealt with. In Figure 1.4b) and 1.4c) the death of a node from the same class would still introduce a different result. While in b) the node death also results in the loss of data from child nodes, in c) some (maybe even all) child nodes may find another transmission route. However, if and to what extent this is possible varies according to the environmental conditions and is difficult to predict

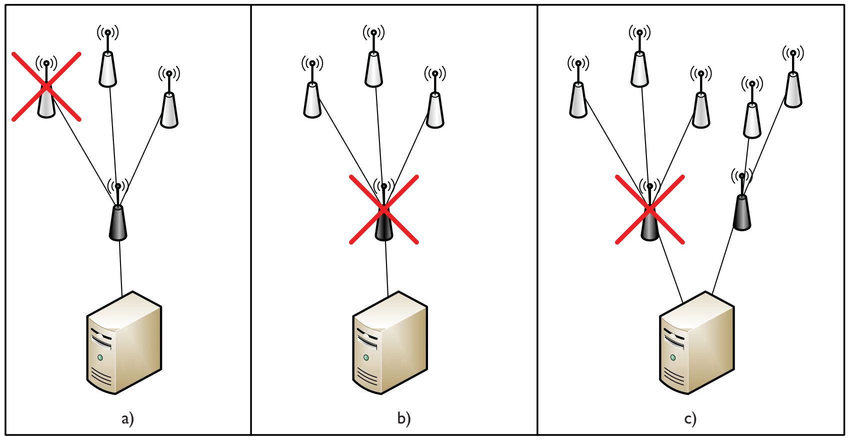


Figure 1.4: Scenario overview of the effect of one node death on the whole sensor network in different situations

prior to deployment.

Another problem is the process of classifying nodes prior to deployment. Determining the number of child nodes in an unknown terrain is almost impossible if dynamic routing is allowed. Further critical and non-critical determination might be performed by other factors, such as sensor value importance. This might be possible in some applications, but certainly not in all. While in a control process the role of a sensor can be predefined and therefore its data delivery importance is given, in EM-applications the importance of data from a single location is undetermined and hence classification is impossible.

Other definitions are based on characteristics such as connectivity and coverage. Coverage generally describes the covered area or phenomena determined by sensors. As long as the interested area/phenomena are covered by a predefined amount of sensors, the sensor network is able to fulfill its task and the system is defined as being alive. Connectivity in sensor networks typically refers to connection with the data sink, rather than the overall connectivity. These two parameters, however, have to go hand in hand. While coverage certainly is important as the sensor data is the reason for deploying a sensor network in the first place, its value is nearly meaningless if it cannot be communicated to the operator. Nevertheless, lifetime definitions based on these values have similar challenges as those based on the number of alive nodes. Predefining when connectivity will be lost, or how important the coverage of a certain location really is, is very difficult if not impossible.

1.4 Problem Formulation and Contributions

Wireless Sensor Networks have the possibility to become a new measurement standard in Environmental Monitoring applications. Enabling autonomous measurements on a large scale, but possibly with high spatial resolution, makes this technology an attractive solution for manifold problems.

However, there are still many challenges to be addressed and solved before WSNs can fulfill their promised vision. While these challenges cover several different areas, such as user-friendliness, self-configuration or data visualization, one very broad area is the lifetime of these systems. Lifetime demands can vary from application to application, but in almost every application within the area of Environmental Monitoring a long lifetime is desired. As mentioned previously, the definition of lifetime in Wireless Sensor Networks is not easy to obtain and this work does not attempt to provide a general definition with regards to these systems. In this case, only a common foundation of lifetime will be addressed, namely the energy management of sensor nodes within the sensor network.

Although WSN-lifetime can have different influencing factors, as discussed previously, the majority of these factors can be reduced or translated into energy demands. While, for example, connectivity is no energy challenge on its own, connectivity of a sensor node can usually be improved by increasing communication output power, hence increasing energy consumption. Similarly coverage problems can be addressed by increasing sampling rates (temporal coverage), or by adding mobility to sensor nodes (spatial coverage). Both factors lead to higher energy demands on the sensor node itself.

Energy management can be addressed both, at the system level and at the node level. As the final energy resource is located and handled at the node level, this is the targeted level in this work. For this, however, the connection between the sensor node life and the sensor network life is of importance. This relation is in some way unidirectional. A Wireless Sensor Network can be considered dead, while all sensor nodes are still alive (e.g., due to interrupted communication links). On the other hand it is impossible to consider your WSN alive, while all sensor nodes are dead (e.g., they are out of energy). Accordingly, sensor node liveliness (at least to some extend) can be considered as being a prerequisite for WSN liveliness.

In this thesis energy management of Wireless Sensor Networks is addressed with reference to the sensor node level. By performing this, the groundwork to enable a long WSN lifetime is provided without the need to define the lifetime of sensor networks on a general level. Energy management is analyzed and conducted on the hardware and software level, including the reduction of energy consumption as well as the incrementation of supplied energy by means of energy harvesting. The focus is based on the application area of Environmental Monitoring and particularly on the application scenario presented in section 1.2.1.

The following list provides an overview of the main contributions due to this work:

1. Analysis and experimental verification of synchronized communication as a method to reduce overall energy consumption by allowing efficient resource use, such as low duty-cycles.

2. Experimental analysis of temperature influence on synchronized duty-cycling and integration of a temperature drift compensation technique.

3. Investigation of micro-scale energy harvesting architectures with a focus on solar energy harvesting in locations with challenging light conditions.

4. Proposition of energy level based simulations as a tool for dimensioning the above mentioned solar harvesting systems. Demonstrative implementation of a model, verification of the model in experimental deployment and an analysis of its simulation capabilities.

1.5 Thesis outline

The remainder of this thesis is organized as follows. Chapter 2 and 3 form the main part of the thesis, followed by conclusions in chapter 4. The focus of chapter 2 lies on the reduction of node-level energy consumption, addressing node-level consumers, consumption reduction mechanisms and methods to efficiently implement these mechanisms. The main targets are the hardware platform and communication costs.

Chapter 3 addresses the power supply of nodes within the system. The limitations of traditional systems (e.g., batteries) are given and solar energy harvesting is presented as a feasible alternative, providing a durable energy supply. A particular focus is given to locations involving limited solar radiation and strong condition variations. Furthermore, the dimensioning of energy harvesters is targeted and a method based on energy level simulations is introduced.

Chapter 4 summarizes the thesis content, discusses proposed results and completes the work with future research perspectives.

In addition, the papers which act as the foundation for this work, are appended to this document.

2 Extending Lifetime by Reducing Energy Consumption

2.1 Consumers in Wireless Sensor Networks

The basic architecture of wireless sensor nodes has not significantly changed during the last decade. It usually contains modules for computation, communication, sensing and power management. Application-specific tasks can require some additional functions, however, in most cases these functions can be classified as belonging to one of the basic modules, mentioned previously. An abstract overview of the hardware architecture of general sensor nodes is provided in Figure 2.1.

In the following, the sensor node modules are analyzed individually. Their tasks and typical implementations are presented, as well as an exposure of their impact on the node energy consumption.

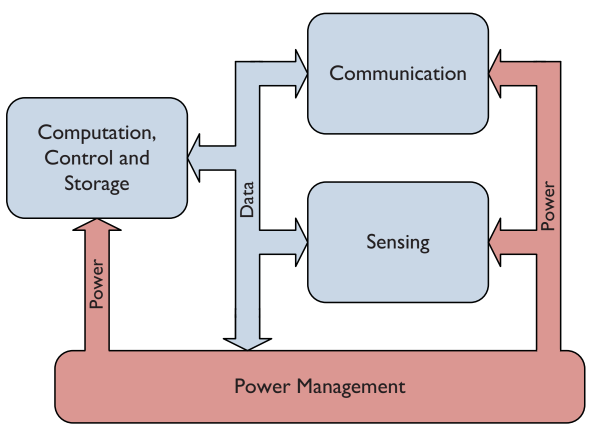


Figure 2.1: Abstract architecture of typical sensor nodes, used in Wireless Sensor Networks

2.1.1 Computation Module

The computation module of a sensor node usually has several tasks to fulfill. It controls the other components on the platform, processes and stores data, and provides an interface to the user/programmer. While the implementation of the computational module will truly depend on the specific application, the node used for most implementations consist of some sort of low-power microcontroller. This is due to the long lifetime expectations of the system and the typically limited energy supply available.

Popular choices for these microcontrollers include Atmel's ATmega series [23, 24], Texas Instruments MSP430 [25, 26, 27], as well as PIC controllers from Microchip [28]. However, for processing intensive applications, instead of or additional to the microcontroller, Field Programmable Gate Arrays (FPGAs) or Digital Signal Processors (DSPs) may be used.

The microcontroller is usually responsible for the application, meaning it is programmed with some sequential code, taking control over the processes necessary in order to fulfill the application tasks. In this case, the controller will determine which components have to be active at a specific time and will control their activities by supplying commands to them. It will usually obtain a response from other modules in the form of communication packets, or sensor samples, storing these in memory, processing them or handing them onward for further activities.

Because active operation consumes a considerable amount of power (i.e., normally in the order of hundreds of $\mu A\,MHz^{-1}$), most microcontrollers offer a series of operating modes, allowing the system to save energy. Thus, at times when certain operations are not required, the controller can change to a lower operating mode, disabling a subset of its circuitry. Nonetheless, to allow this operation, the microcontroller needs to facilitate means to wake itself up again (or get woken up externally) in order to continue normal operation when necessary.

2.1.2 Communication Module

In a similar manner to the computational module, the communication module implementation also depends to some degree on the application. Nevertheless, it is also the case that the majority of systems use similar communication devices, namely low-power Radio Frequency (RF) transceiver, typically operating in the license-free ISM-band. Other communication methods, such as acoustical or optical are only used seldomly (e.g., for underwater sensor networks).

This communication module is used for the local communication between nodes in the sensor network. This means that the main purpose of this communication link is to transfer measured data from the sensor node to a common gathering point (i.e., the network sink), and in return to send commands towards the individual sensor nodes. In addition to this local communication link, a global communication module might be implemented. This module has the purpose of connecting the local sensor network to the outside world. Typical implementations of this global communication module are Wifi [16], long-range radio communication [13, 11] or GSM/GPRS [21, 8]. However, it is usual for a strictly limited number of nodes to have this capability as both power consumption and the price for these devices are excessive.

There have been some minor changes made to the implementation of the local communication module during the last decade. Although in the majority of case low-power radio transceiver are still used, there is a difference in their detail. During the initial introduction of Wireless Sensor Networks proprietary radio transceiver were used, mostly operating around the 900 MHz band, because mostly originating from the United States. Typical devices at this time were the RFM TR1000 or Chipcon's CC1000. Nowadays, most sensor nodes use transceivers according to the 802.15.4 Standard and operate in the 2.4 GHz ISM-band, allowing world-wide operation at the same frequency.

Communicating via RF, however, is costly for resource limited devices, such as sensor nodes. Even when operating in idle mode, and only listening to surrounding noise, energy consumption of the communication module is tremendous, easily reaching similar levels to those involved in transmitting data. Therefore, sensor nodes should disable their communication modules whenever possible, reducing power consumption by several orders of magnitude. However, waking up in time to receive a packet destined for this node can become the major problem.

2.1.3 Sensing Module

While the implementation of computation and communication modules change depending on the resource requirements of the application, it is definitely the case that the most application-specific part of a sensor node is its sensing module. A given application will be required to monitor certain physical parameters or detect specific events. This in turn requires specific sensor units that have the ability to fulfill these application demands.

Although any type of sensor is imaginable for WSN operation, the choice of sensors found in the literature is rather limited. Most work is limited to low-power sensors, with a typical example being temperature sensing. The possible reasons for this are limited application-oriented research, difficulties in performance comparison when there are different underlying assumptions, as well as the simplicity and availability of these sensing devices.

Nonetheless, this easily leads to assumptions within the community that do not generally hold true. One typical example being the negligible energy consumption of the sensing module. While this might hold true for above mentioned temperature sensors, there are many sensors (e.g., gas sensors or cameras) that have higher power consumption than the communication module or require tremendous warm-up times in order to give accurate readings.

2.1.4 Power Module

Underlying all other node modules is the power module. Its main task is as simple as it is important, namely in providing a stable power supply to all active components of the sensor node system. This means it converts the input from the energy source into acceptable levels in order to power the connected devices.

How this conversion actually apperas will generally depend on the type of energy source used for the sensor nodes. In some cases the sensor node might be able to receive power from the main power supply, requiring some kind of AC-DC conversion. However, especially in EM-WSN applications, this is seldom the case. The use of battery supplies is more popular and recently the use of harvesting energy from ambient energy sources, such as wind, temperature difference or sun has been involved. For these sources, power modules implement simpler DC-DC converters. For the sensor nodes these typically use Low Dropout (LDO) regulator, as well as buck or boost converter.

Additionally, the power module might include monitoring and control functions. Monitoring is again mainly dependent on the energy source in use. For example, when using a battery a typical desired monitoring function to be implemented is the determination of the battery charge level so as to predict the time of failure, etc. On the other hand, for energy harvesting systems, the observation of incoming energy level might be more important, due to their intermittent behavior. Control functions are implemented, for example, to react to monitored events or periodic tasks. A typical example for this is the shutting down of certain modules on the node to conserve energy, either because they are unnecessary or because reduced energy income was detected.

2.2 Duty-Cycling

As mentioned previously, most active components (e.g., microcontroller and RF-transceiver) provide different power states, with tremendous differences in their energy consumption. In reality the choice of operation states can be quite broad. For example, a typical MSP430 microcontroller from Texas Instruments offers seven different states (i.e., one active mode, five low power modes, as well as shutdown), allowing the optimization of energy conservation by picking the most appropriate operation mode at any given time. However, for simplification reasons, in the following, a division will only be made between *active state* and *inactive state*. This will provide the basic underlying principles, that can be extended by means of further operation state layering when required.

Duty-cycling is a common approach to reducing the average energy consumption in Wireless Sensor Networks. Its main principle is to achieve lower average power consumption by being inactive whenever possible (see Figure 2.2 for graphical representation). Defining the portion of time in the active state as T_{active} and the period in the inactive state as $T_{inactive}$, the duty cycle δ can be formally described as

$$\delta = \frac{T_{active}}{T_{active} + T_{inactive}}, \quad 0 < \delta < 1\,. \tag{2.1}$$

For this to be effective, the duty cycle δ should be considerably smaller than one. Furthermore, the power consumption levels in the respective states have to differ. In typical WSN applications, these are both the case. Specifically in Environmental Monitoring, sampling rates are rather low (i.e., typically in the order of minutes [6, 29, 9]). This leads to typically very low duty cycles in the network. Consumption level differences are reached by toggling between active and inactive states. An overview of the power consumption levels usually achieved in different module states is given in Table 2.1.

Taking these factors into account, duty-cycling can be applied to the different consumers of the sensor node. In general the average power consumption obtained by duty-cycling can be described as

$$P_\delta = \frac{P_{active} \cdot T_{active} + P_{inactive} \cdot T_{inactive}}{T} \tag{2.2}$$

$$= \delta \cdot P_{active} + (1 - \delta) \cdot P_{inactive}\,. \tag{2.3}$$

Herein the cycle-time T is the sum of T_{active} and $T_{inactive}$ and δ as described in Equation 2.1. However, for each sensor node module different challenges have to be addressed. The simplest of these is, usually the com-

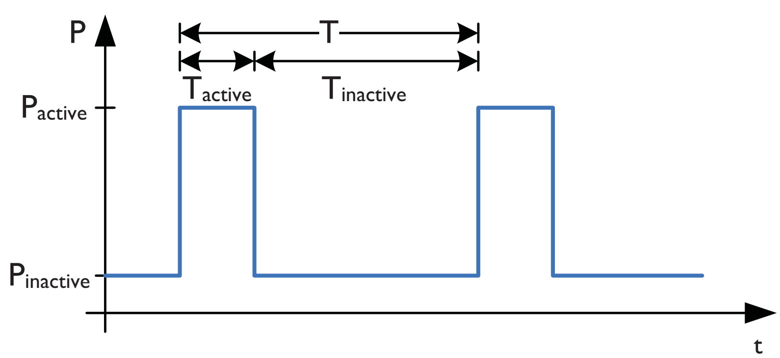


Figure 2.2: Graphical representation of the duty-cycling principle (simplified for bi-modal operation states)

Module	State: Operation	State: Classifier	Consumption
Communication	RX (RF)	Active	Tens of mA
	TX (RF)	Active	Tens of mA
	Sleep	Inactive	µA
Computation	Processing	Active	Hundreds of µA MHz^{-1}
	Memory access	Active	mA
	Sleep	Inactive	µA
Sensing	Sampling	Active	µA – hundreds of mA
	Warm-up	Active	µA – hundreds of mA
	Sleep*	Inactive	µA

Table 2.1: Operation states and consumption levels of typical sensor node modules for EM-Wireless Sensor Networks

putational unit, as this is the point at which the control takes place. The computational unit therefore can switch to a low-power mode whenever idle (i.e., waiting for the next event to occur). It only has to be possible to awaken the module from its low-power state as soon as the event occurs. This is usually obtained by interrupts. The resulting power consumption can be estimated by substituting computational unit values into equations 2.2 and 2.3

$$\begin{aligned} P_{comp,\delta} &= \frac{P_{comp,active} \cdot T_{comp,active} + P_{comp,inactive} \cdot T_{comp,inactive}}{T_{comp}} && (2.4) \\ &= \delta_{comp} \cdot P_{comp,active} + \left(1 - \delta_{comp}\right) \cdot P_{comp,inactive} \,. && (2.5) \end{aligned}$$

In this, however, $T_{comp,active}$, $T_{comp,inactive}$, and result of this δ_{comp}, are typically not known precisely. Because the active and inactive periods are not scheduled and the computation occurs only as necessary, it is difficult to predict the exact active and inactive time intervals. Nevertheless, quite accurate estimations can be made, when events that trigger computation are known and occur periodically.

However, it is usual for the sensing module activities to be typically scheduled by the computational unit. This is particularly true in periodic sampling applications, when the ratio of active to inactive time is rather predefined. However, many sensors do not have low-power states, which means that they have to be switched off at times when they are not required. This in turn leads to longer power-up intervals, as sensors might require some time to warm-up, before producing accurate results. Integrating this into the equations leads to an average power consumption of

$$P_{sens,\delta} = \frac{P_{sens,on} \cdot \left(T_{sens,warm} + T_{sens,samp}\right) + P_{sens,off} \cdot T_{sens,off}}{T_{sens}} \quad (2.6)$$

$$= \delta_{sens} \cdot P_{sens,on} + \left(1 - \delta_{sens}\right) \cdot P_{sens,off}\,, \quad (2.7)$$

with $T_{sens,warm}$ being the warm-up time and $T_{sens,samp}$ the sampling time. From this the duty cycle δ_{sens} results in

$$\delta_{sens} = \frac{T_{sens,warm} + T_{sens,samp}}{T_{sens,warm} + T_{sens,samp} + T_{sens,off}}, \quad 0 < \delta_{sens} < 1\,. \quad (2.8)$$

The warm-up time of sensors can vary tremendously between the different types of sensors. In gas sensors (e.g., Nondispersive Infrared (NDIR) CO2 sensors) it can easily reach tens of seconds. Hence for the sensor module, the sensor warm-up time has a large impact on the duty-cycling efficiency. Also consumption in the active state can be immensely different from sensor to sensor. Duty-cycling low-power sensors therefore might not be useful.

On the other hand, typically used communication transceivers in the radio frequency band show a considerable consumption difference between the active and inactive states. Therefore duty-cycling for these radio chips is desired in most cases. However, transmission and reception of data can only occur when the transceiver is in the active state. While this is not a big problem for the transmission of data, data reception is difficult to predict. At the same time, one does want to reduce the active time. A mechanism towards this, especially suited for periodic data gathering applications, is presented in more detail in section 2.4.

Completing the description set, for the communication module we can approximate duty-cycled power consumption as

$$P_{com,\delta} = \frac{P_{com,active} \cdot (T_{com,RX} + T_{com,TX}) + P_{com,inactive} \cdot T_{com,sleep}}{T_{com}} \tag{2.9}$$

$$= \delta_{com} \cdot P_{com,active} + (1 - \delta_{com}) \cdot P_{com,inactive}\,, \tag{2.10}$$

with

$$\delta_{com} = \frac{T_{com,RX} + T_{com,TX}}{T_{com,RX} + T_{com,TX} + T_{com,sleep}}, \quad 0 < \delta_{com} < 1\,. \tag{2.11}$$

In this case, $T_{com,RX}$ is the reception time, $T_{com,TX}$ the transmission time and $T_{com,sleep}$ the period in sleep state. Furthermore, transmission and reception power consumption are simplified to be the same, defined as $P_{com,active}$.

Superimposing these equations and adding some static consumption, such as for power supply ICs, a simplistic model can be built to estimate the node power consumption under different conditions. Figure 2.3 shows an example,

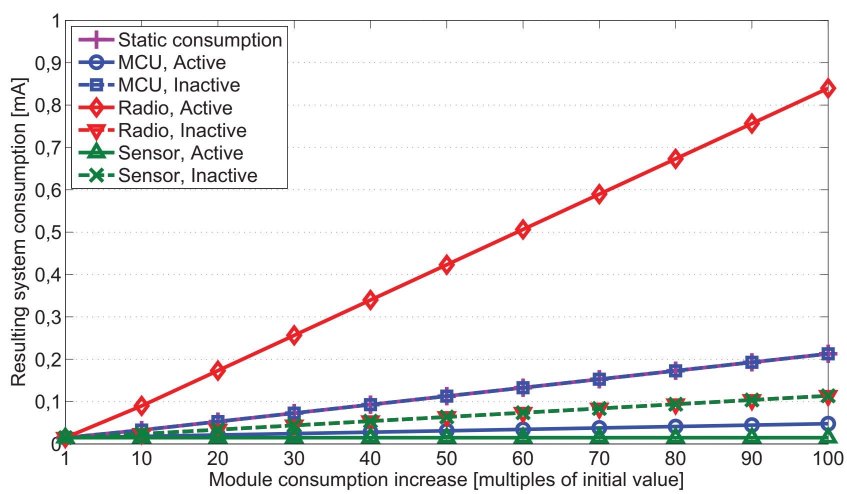


Figure 2.3: Influence of the consumption per module on the overall duty-cycled system consumption. Consumption increase can mean both, increase in amplitude and increase in time period

Module	State	Time	Power
Static		always	2 µA
Microcontroller	Active	10 ms	2 mA
	Inactive	60 s – 10 ms	2 µA
Radio	TX	10 ms	30 mA
	RX	10 ms	20 mA
	Inactive	60 s – 20 ms	1 mA
Sensor	Sampling	100 µs	100 µA
	Sleep	60 s – 100 µs	1 µA

Table 2.2: Assumptions for a typical low-power Environmental Monitoring sensor node, periodically gathering data in a duty-cycled fashion (based on Sentio-e² characteristics)

where the influence of module power consumption on system power consumption is analyzed. Module consumption increase is changed by multiples of the initial module consumption, which in turn is based on the duty-cycling equations as previously mentioned and the application assumptions as shown in Table 2.2. For these assumptions to provide real-world correlation, they have been based on measured values of Sentio-e² [30], a hardware node platform for use in Environmental Monitoring. Furthermore, a time frame length of one minute is chosen. Thus, every minute the microcontroller wakes up, takes one sample and this sample is transmitted somewhere. Also, one packet will be received, which could for example represent an acknowledgment for the transmitted data.

Figure 2.3 illustrates the well known fact with regards to the high communication cost in low-power sensor networks. This is shown in the solid red line in the graph and it is obvious that the system is sensible to even small increases of communication demands. However, the graph also demonstrates the impact of possible changes to current hardware. For example, it is possible to extract the information that increasing the power consumption of the microcontroller in the active state (e.g., increasing the clock frequency) does not have a significant impact on the system power consumption. In this scenario an increase by a factor of ten (i.e., theoretically an improvement from 4 MHz to 40 MHz) increases the overall system consumption from about 14.5 µA to about 17.5 µA, while not even considering a reduction in processing time.

2.3 Node-level Power Domains

As depicted in the last section, in addition to radio communication, in particular, the inactive state consumptions have an impact on the overall consumption of the duty-cycled sensor node. This is not very surprising, considering the typically low duty cycles (i.e., the long time periods in inactive state). Therefore the power consumption in these states should be as low as possible, to maintain the overall system consumption at as low a value as possible. This is reflected in the low-power modes, implemented in the present day microcontroller and radio ICs. These devices can be easily set to an operational state, where the power consumption is reduced by orders of magnitudes as compared to the active values.

However, not all devices offer low-power modes or actually consume considerable amounts of energy. Typical examples for these are high-performance co-processors (e.g., FPGAs or DSPs) and some types of sensors. These components should usually still be duty-cycled, in particular as they consume comparably large amounts of energy. A possible solution for this involves the implementation of power domains.

Power domains divide components into groups, typically used together, equipping them with their own controllable power supply. This allows the microcontroller, usually by a single GPIO pin change, to cut the connection from a power domain to the power source. Thus the consumption from that power domain is reduced to the leakage of the power supply unit in the off-state. For low-power LDO regulators, this can typically stay below 1 µA.

Figure 2.4 visually illustrates this principle. While the power domain A in this case has to be powered all the time in order to maintain the system control, all other power domains can be switched-off in a controlled fashion by power domain A. However, there are some aspects which should be considered before implementing massive amounts of power domains. First of all, the costs are increased by the addition of power supply components,

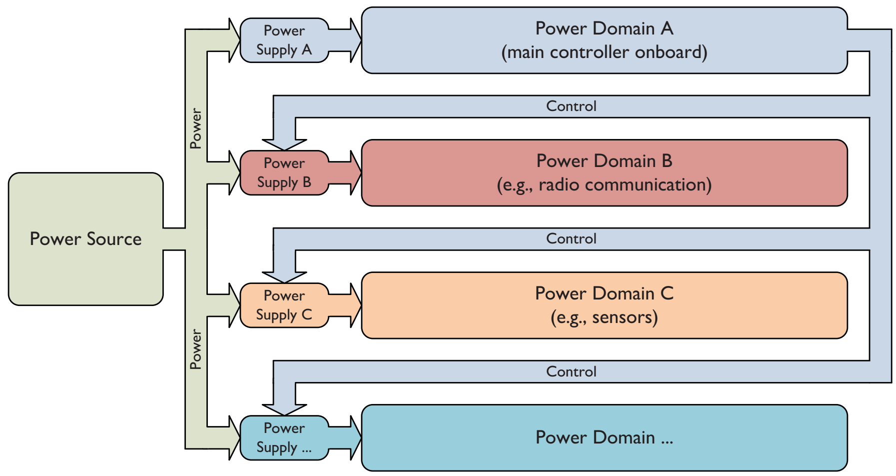


Figure 2.4: Principle of different power domains for dynamic adjustment of power consumption and on-demand performance

especially in cases where it is usually considered to be possible for two power domains to have been supplied by the same power supply. Furthermore, switching off a sub-circuit might not always be the most power-efficient solution. Startup times from shutdown are, for many devices, longer than switching between low-power and active mode. If the savings involved in disabling the power supply, compared to low-power modes, are only minor, they might be overpowered by the longer wakeup times for the devices when in use. Additionally, there is a risk of component damage involved, when communication between components of different power domains has to occur. Most Integrated Circuits do not allow signals of higher voltage than their supply voltage on their terminals. Thus, if the controller or any other active device attempts to communicate with an inactive device or a device of lower power supply, it might damage its circuitry. Preventing this from happening is not a significant problem (e.g., by implementing line breakers or level converters). However, if fixed in hardware, cost increase by additional components is unavoidable. On the other hand, prevention in software does not increase cost, but requires the software engineer to have some knowledge concerning this phenomena.

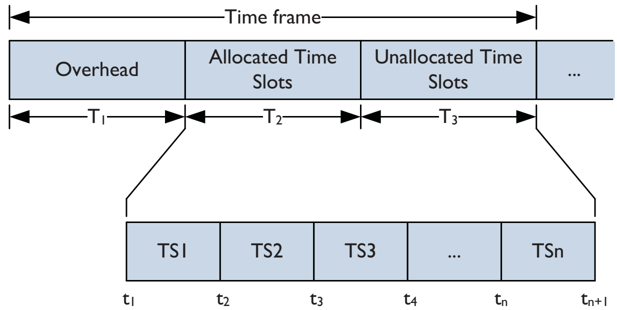


Figure 2.5: Typical structure of a TDMA based communication schedule

2.4 Synchronized Communication

While it is nearly impossible to reduce the consumption amplitude for the communication module on the node level, optimization of active time is possible to address on the software level. Choosing an appropriate communication protocol plays a major role in achieving this. The protocol has to reduce the time spent in transmission and reception states as much as possible, while maintaining a reliable communication.

In time-driven data gathering applications, sampling and communication occurs periodically. Unlike the case with event-driven systems, this means that the communication times are known beforehand. Thus, we believe, scheduled communication protocols will be the most efficient ones in these applications. A typical scheduled protocol is Time Division Multiple Access (TDMA), with its principle depicted in Figure 2.5.

Time frames are reoccurring in TDMA, and each time frame consists of the same structure for the time slots. Usually in the beginning of a time frame, there is a defined overhead period, which allows for changes to be made in the following schedule. The schedule is then divided into time slots, each allocated to one individual communication node. Depending on the implementation of the protocol, time slots can be of equal size or depending on the data length of the respective time slot holder. Some nodes might hold more than one time slot per time frame. Furthermore, a time frame usually contains more time slots than initially allocated, to allow for an easy adaptation to changes. The amount of unallocated time added is a trade-off between additional frame overhead and the number of time frame structure changes. Mostly it depends on the degree of dynamics in the communication network.

This section will particularly cover synchronous communication. We will look in greater detail into the communication demands of Wireless Sensor Networks and present reasons why synchronous communication methods offer advantages for the application under the present scope. The coverage will include what synchronization is, what requirements on the synchronization method are posed, and how synchronization can be achieved. Furthermore, we present a synchronization method for the specific application demands given, and demonstrate its performance under different conditions.

2.4.1 Communication in Environmental Monitoring Wireless Sensor Networks

In a similar manner to that for the classification of sensor network applications, as presented in section 1.2, communication behavior of WSNs can be differentiated. For the majority of the time protocols are grouped into *scheduled* communication methods and *on-demand* communication. Not surprisingly, scheduled communication protocols are more popular in time-driven applications, while event-driven applications more often employ on-demand protocols. This is due to the different impact of communication protocol factors on each application domain respectively. Table 2.3 provides an overview of typical design consideration factors for communication protocols. While latency, throughput and fairness can be considered rather traditional design considerations, energy is a factor that has gained attention with the introduction of battery-powered communication systems, such as Wireless Sensor Networks.

For on-demand protocols, latency and throughput are important design considerations, while scheduled communication protocols rather focus on energy-efficiency. The reason for this is, that if an event occurs and is detected by a node, the event should be reported as quickly as possible, without any long waiting times and network latencies. In particular when dealing with dangerous events, such as fires or earthquakes, the energy spent for reporting this event is of less importance, while providing warnings and alerts should occur as soon as possible. In these situations, fairness handling also

Factor	Description
Latency	Describes time delays in communication networks. However, an exact definition can vary depending on the viewpoint of the user. In WSNs latency can on the one hand describe time delays for pure packet transmission, such as the time interval from transmitting the packet at the sender's site to the reception of the packet at the destined receiver. On the other hand, and more often, latency in WSNs is the delay from sampling at the node to reception at the sink. Due to the shared medium and the possibly large number of nodes, latency in this way can be considerable large.
Throughput	Throughput is the amount of successfully transmitted data via a communication channel. It usually is measured in terms of bits s^{-1}, but might also be measured in number of packets per defined time period. In Wireless Sensor Networks nodal throughput is mostly used as a qualitative measure (i.e., low, medium, high), because quantitative description is rather difficult. It is influenced by packet delivery ratio, number of competing nodes in the same broadcast domain and communication overhead.
Fairness	Fairness is usually important when a large amount of communication nodes share a common transmission medium. As opposed to *first-come-first-served* approaches, in a fair communication protocol every node has a chance to transfer its data and nodes cannot block the channel. However, depending on the application, in WSNs fairness might intentionally be removed by providing priority to a subset of nodes or types of packets.
Energy Efficiency	Energy efficiency gains more attention, when the communication devices involved are not mains powered (e.g., battery powered). Often energy efficiency is related to the lifetime of the communication systems, and describes how much energy is spent for the transmission of a certain amount of data. Especially in networks with typically low traffic, such as WSNs, idle energy consumption has a larger impact on the overall consumption. In these networks, protocol optimization for reducing energy consumption, which mostly means reducing active time, is of great importance.

Table 2.3: Design consideration factors for WSN communication protocols

plays an important role in the communication protocol. Communication should intentionally be unfair, since it is desirable that nodes with important information have priority over other nodes.

On the contrary, time-driven applications, such as the periodical data gathering under the scope of the present work usually do not have these tight reporting constraints. Their data sampling and reporting is predefined and energy efficiency becomes the major design concern, because the duration of operation is of prime interest. Having tight communication schedules can reduce the active time of the involved sensor nodes and thereby reduce the energy consumption. An overview of the impact factors, adding to the communication energy consumption, is given in Table 2.4. While these factors contribute considerably to the energy consumption of the communication module, they do not actively contribute to the data transmission and thereby reduce energy efficiency. The impact of these factors can be reduced, in some cases even eliminated, by the use of scheduled communication protocols. However, to enable tight communication schedules a common notion of time within the network is important, which usually requires time synchronization between nodes.

2.4.2 Time Synchronization for Wireless Sensor Networks

There are several tasks in Wireless Sensor Networks, which demand or profit from a common notion of time between nodes. Among these are time-stamping of samples, collaborative sampling of an event at the same time, and the measurement of time related properties (e.g., velocity). To achieve this common notion of time, synchronization is mandatory. While for synchronization, due to increased communication, typically additional energy is consumed, synchronization can also be used to reduce energy consumption spent for communication in the network.

The need for time synchronization between sensor nodes is based on the non-ideal clock source, implemented on these devices. Ideally, the time retained by a sensor node $C(t)$, should represent the reference time t, thus

$$C(t) = t\,. \tag{2.12}$$

However, due to the non-ideal properties of the clock source, a relation of

Factor	Description
Collision	A collision is a superposition of two or more data transfers at the same time. At the receiver this leads to a non-decomposable reception and therefore to data loss. In non-scheduled communication collision is avoided by *listen-before-send* mechanisms or additional control sequences, such as Request-to-Send (RTS) and Clear-to-Send (CTS) packets. In scheduled protocols, this problem is solved by allocating unique time periods for communication to nodes respectively.
Overhead	Overhead is additional information that has to be sent with the data, e.g. to determine source and destination. Overhead increases in on-demand protocols when control sequencing is used. Depending on synchronization techniques, scheduled communication might also have increased overhead. In scheduled communication, however, source identification might be omitted by using time allocation instead.
Over-Hearing	Over-Hearing describes the problem of one node listening to communication, or part of the communication, dedicated to another node. While this problem is still applicable to on-demand protocols, in scheduled communication each node is aware when data is meant to be targeted to itself and when not.
Over-Emitting	Over-Emitting is the effect of one node transmitting data to another node, while the receiver is not listening. The result is similar to collisions, and once again it is solved by defining communication times in scheduled protocols.
Idle-Listening	Idle-Listening occurs when the receiver is waiting for a transmission, listening to the channel without traffic occurring. Consumption in this state is similar to active reception. On-demand protocols usually target this problem by listening periodically to the channel. Thus reducing, but not eliminating the impact. Scheduled communication ideally knows exactly when the transmission will start. However, in reality they are unable to eliminate idle-listening completely, because of the inaccuracy of the synchronization.

Table 2.4: Parasitic communication factors, that contribute to the energy consumption of WSN nodes

node i's clock to the reference time of

$$C_i(t) = \omega \cdot t + \phi \tag{2.13}$$

becomes a more appropriate description model. In this case, ω describes the clock skew or frequency offset of node i's clock source and ϕ is a phase offset. Using one node as the reference, we can further describe node clock relations as

$$C_j(t) = \omega_{ij} \cdot C_i(t) + \phi_{ij}, \quad i, j = 1, 2, \ldots, N, \tag{2.14}$$

where ω_{ij} and ϕ_{ij} are the clock skew and phase offset between node i and j respectively and N is the number of nodes in the network. Time synchronization between nodes is reaches when

$$C_j(t) = C_i(t), \quad i, j = 1, 2, \ldots, N. \tag{2.15}$$

It should be noted that this does not mean synchronization to real time, but synchronization of one node's clock to the clock of the reference node. Comparing equations 2.14 and 2.15, depicting the initial and the desired case respectively, it is obvious that the goal of synchronization is to compensate for clock skew and phase offset (i.e., $\phi_{ij} = 0$ and $\omega_{ij} = 1$).

While the phase offset ϕ typically results from the different start configuration or start-up times of nodes, a single compensation run can usually eliminate the impact. However, clock skew is not a static factor, but will vary over time. Typically this variation is divided into *short-term* and *long-term stability* [31]. Short-term stability is usually influenced by environmental factors, such as temperature, pressure or supply voltage, showing a relatively direct effect on the stability of the clock source. On the other hand, long-term effects will influence the clock skew rather slowly over time. A common example of a long-term effects is oscillator aging.

The time-dependency of clock skew has the consequence, that a single compensation of the frequency offset is not sufficient. Instead, regular resynchronization is mandatory in order to maintain synchronized node clocks. The resynchronization interval thus depends on the degree of environmental changes, the stability of the used clock source and the necessary synchronization accuracy to be achieved in the respective applications.

In general, perfect synchronization cannot be guaranteed, as the synchronization method itself will introduce a certain amount of inaccuracy. Typical ways of establishing synchronization in WSNs, sources of inaccuracy, design consideration and trade-off factors will be introduced in the following section.

Design Considerations and Sources of Inaccuracy

When implementing a synchronization method to reach a common notion of time within the network, there are several design factors that must be considered. The majority of these factors are related and therefore lead to trade-off situations. These trade-offs have to be optimized for each application or system case individually. Typical design considerations include synchronization accuracy, the energy cost of the synchronization method, in addition to scalability and behavior in dynamic networks.

Synchronization accuracy is the most obvious design factor, as synchronization is implemented for a purpose, namely to create a common notion of time. However, curtailment of synchronization accuracy might be acceptable when the remaining accuracy still results in it being sufficient for the targeted tasks. Synchronization accuracy is related to allocated resources, thus also to energy cost. Additionally scalability might be limited when very high accuracies is required to be reached. Synchronization accuracy is mainly limited by delays in the wireless communication. An overview of these delay components is given in Table 2.5 and their occurrence during a packet transmission is presented in Figure 2.6.

In energy constraint systems, such as Wireless Sensor Networks, the energy spent for synchronization can be a matter of consideration. Depending on the desired accuracy of the implementation, considerable amounts of energy can be spent in the communication or processing stages. In particular when synchronization is used to reduce energy consumption (e.g., by building communication schedules), limiting the energy overhead spent by the synchronization approach is of major importance. Usually energy cost considerations conflict with synchronization accuracy and might influence the reliability of the network in dynamic situations.

Scalability and dynamic network behavior are factors that should be considered in any sensor network, which is planned to be deployed. Scalability herein describes how the synchronization accuracy is influenced with the increasing size of the network. A typical problem is additionally introduced delay due to multi-hop synchronization. On the other hand, dynamic network behavior deals with situations when for example resynchronization appointments are missed.

Delay Component	Description
Send Time	Time period of handling the send request and preparing the packet transmission. Influenced by system load and overhead. Can be reduced by limiting system load by predicting time of resynchronization.
Access Time	Time of accessing the transmission medium. Influenced by network load. Usually high in contention-based protocols, while low for scheduled communication.
Transmission Time	Time to actually transmit the packet to the medium. Depends on the packet size, but can be estimated rather accurately.
Propagation Time	Time the message needs to propagate from sender to receiver. Depends on the distance between the two nodes. Comparably small in typical RF-based and short-range WSNs.
Reception Time	Time it takes to receive the packet at the receiver's side. Complementary to Transmission Time.
Receive Time	Time for processing the received packet until the application is informed. Complementary to Send Time.

Table 2.5: Delay factors in Wireless Sensor Network transmissions, influencing synchronization accuracy when not addressed

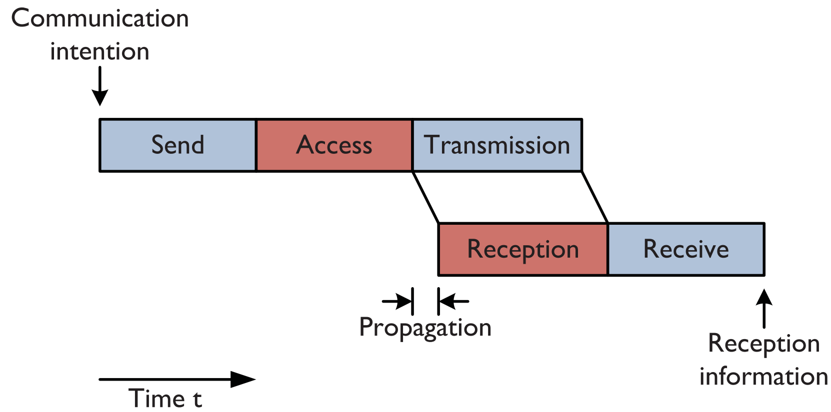


Figure 2.6: Occurrence of delay components during a packet transmission between two nodes in a Wireless Sensor Network

Time Synchronization Methods and Classification

In essence, time synchronization between sensor nodes is established by reporting one node's time to the other node(s). As introduced in Table 2.5, the transmission of packets occurs with a mainly unpredictable delay, reporting a time in the past at the receiver when used for synchronization. This principle is illustrated in Figure 2.7a and can be formally described by

$$C_B(t_2) = C_A(t_1) + \phi_{AB} + \delta\,, \tag{2.16}$$

wherein ϕ_{AB} is the clock offset between node A and B, and δ the sum of delay components as mentioned previously. As interested in ϕ_{AB}, δ acts as a parasitic effect on the synchronization. Methods to handle this are on the one hand, attempting to reduce the delay times in the system, and on the other estimating/calculating them.

A typical way of dealing with the latter case, is to use bidirectional communication to implement synchronization. This enables the estimation of the delay components by calculating round-trip delays. This method can be graphically demonstrated as shown in Figure 2.7b. Formally, it is possible to add the backward transmission

$$C_A(t_4) = C_B(t_3) + \phi_{BA} + \delta, \quad \phi_{BA} = -\phi_{AB} \tag{2.17}$$

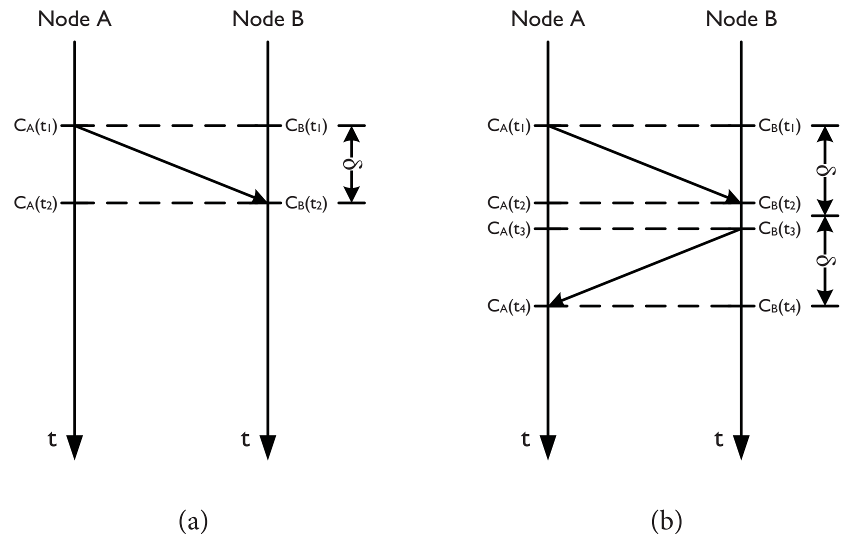


Figure 2.7: Underlying timeline of packet transmission for (a) unilateral synchronization method and (b) bidirectional synchronization method

to the description, and thus are able to estimate both, δ and ϕ_{AB} as

$$\delta = \frac{(C_B(t_2) - C_A(t_1)) + (C_A(t_4) - C_B(t_3))}{2} \tag{2.18}$$

and

$$\phi_{AB} = \frac{(C_B(t_2) - C_A(t_1)) - (C_A(t_4) - C_B(t_3))}{2}\,. \tag{2.19}$$

However, this introduces some estimation error, because delays are simplified to be of the same length for both communication directions and an additional delay between t_2 and t_3 is added to δ. Furthermore, we increase the number of transmissions and now the receiver has to start the synchronization process (or an even higher number of packets is needed). This might lead to energy overhead problems in systems with large number of nodes.

As the intention of using a common notion of time in the network varies from system to system, so does the design of synchronization algorithms. Due to the large amount and variance of solutions, common classification factors have been introduced, such as in [31, 32, 33]. A list of these classification factors is presented in Table 2.6, also summarizing the meaning of each factor.

Many time synchronization algorithms have been presented for use in Wireless Sensor Networks. The typically known ones within the community of time synchronization, are Timing-Sync Protocol for Sensor Networks (TPSN) [34], Reference Broadcast Synchronization (RBS) [35], Flooding Time Synchronization Protocol (FTSP) [36], Lightweight Time Synchronization (LTS) [37], Pairwise Broadcast Synchronization (PBS) [38] and Time Diffusion Protocol (TDP) [39]. Because of their intended use in WSN applications, energy efficiency is addressed in all of them. Nonetheless, as most of these synchronization mechanism are based on the bidirectional synchronization method, the number of packet transmissions to establish synchronization can be rather high. However, FTSP is based on the unilateral approach and therefore limits the number of packet transmissions. To address delay components, however, low level time-stamping is used and several packets are sent to allow for the estimation of clock-skew via linear regression.

When time synchronization has its main purpose in energy savings, such as in the case targeted here, the number of communication packets used for establishing a common notion of time, which essentially describe the energy overhead for the synchronization algorithm, becomes one of the major concerns. However, at the same time the protocol has to maintain a certain degree of accuracy, allowing for energy-efficient, but stable communication.

2.4.3 Synchronization Method for Efficient Duty-Cycling in Environmental Wireless Sensor Networks

As mentioned previously, typical sampling intervals in Environmental Monitoring-WSNs are in the order of minutes. This is due to the measurement of typically slowly changing parameters, such as temperature, gas concentrations or barometric pressure. In networks with a large number of sensor nodes this means, that new data is sampled at a rather slow rate, but once sampling is due, a large number of nodes want to communicate their data to a central collection point. The lack of synchronization in this situations has a twofold impact. On the one hand, sampling should usually occur at the same time as this will basically allow for an image of the monitored environment to be taken at a certain time. Lacking time synchronization in this case means, that data correlation cannot be guaranteed. On the other hand, the transmission of the sampled data has to be coordinated. In a common on-demand protocol the nodes would attempt to transmit their data directly after sampling, which would in the majority of cases lead to a massive number of collisions. Using a scheduled transmission without a common timebase only partially solves this issue, as different node clock skews will lead to overlapping transmission times.

A synchronization algorithm, targeting these issues in the present application, should have certain properties. Synchronization should be network-wide, as all end-nodes are involved in the communication process. The synchronization should allow communication at any time, thus an algorithm that maintains the nodes as synchronized is preferable over on-demand synchronizing solutions. Resulting accuracy is not the main target of the

Classification Factor	Description
Internal/External	Describes whether the algorithm synchronizes network nodes to some sort of external clock source or only amongst each other.
Always/On-demand	Describes whether nodes stay synchronized all the time or whether they are synchronized on-demand.
All/Subset	Describes whether all nodes will be synchronized or only a subset of nodes.
Rate/Offset	Describes whether nodes synchronize to the same clock rate or whether a clock offset is compensated.
Correction/Translation	Describes whether node clocks are corrected to the common time or whether the timescale is translated.

Table 2.6: Main classification factors of time synchronization algorithms in Wireless Sensor Networks

protocol, but the best achievable accuracy for a minimum energy spent is desired. As energy overhead is the main concern, the optimization goal for the solution is minimum energy, thus minimum communication.

The consequence regarding communication limitation, is that all pairwise synchronization schemes are not appropriate. Considering the communication minimization aim, a single synchronization message is the desired outcome. Using the broadcast capability of the RF-channel, the same message can be used to synchronize all nodes in the same broadcast domain. A single (unidirectional) synchronization message, does enable for the compensation of the phase offset between nodes, but it cannot target the clock skew. However, as periodic resynchronization is required to address temporary changes, the time interval between two synchronization messages can be used as a second parameter. This does allow for the additional compensation in relation to clock skew.

As depicted in Figure 2.8, in an interval-based synchronization method, message transfer occurs periodically and unidirectionally. The node acting as the clock reference periodically broadcasts a synchronization packet. This packet can be interpreted as a typical Start Frame Delimiter (SFD) for the synchronization frame. Theoretically, there is no content necessary within this message, but as the packet has to be transfered anyway, some global information might as well be included. If indeterministic delay components, such as access time or system load, are handled to be minimal, the transmission delay of the two following packets can be estimated as a constant value.

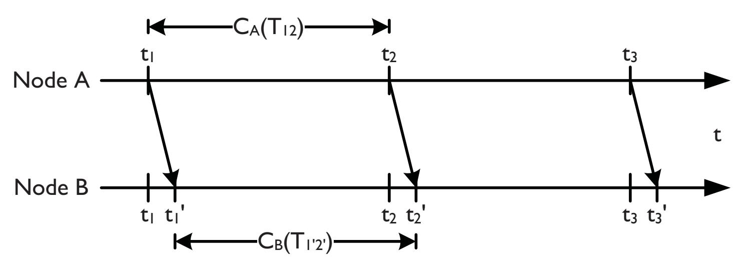


Figure 2.8: Timeline of an interval-measurement based synchronization method

Therefore, as indicated in Figure 2.8, we can define

$$\delta = t'_1 - t_1 = t'_2 - t_2 = t'_i - t_i \,. \tag{2.20}$$

This means, a synchronization message sent by node A at t_i will arrive an unknown time later at node B at the time t'_i, with

$$t'_i = t_i + \delta \,. \tag{2.21}$$

Furthermore, when defining a constant interval T between synchronization messages, node A will periodically send out synchronization messages with its own understanding of periodicity T, namely

$$C_A(T) = C_A(t_{i+1}) - C_A(t_i) \,. \tag{2.22}$$

Allowing node B to measure the interval between two consecutive synchronization messages will lead to a measured inter-packet length of

$$C_B(C_A(T)) = C_B(t'_{i+1}) - C_B(t'_i) \,. \tag{2.23}$$

As the delay components of both transmissions are estimated as being constant, the real-time length of both intervals will be the same, thus

$$C(C_A(T)) = C(C_B(C_A(T))) \,, \tag{2.24}$$

with

$$C(T) = T \,. \tag{2.25}$$

At this point, all receiver nodes only know their interpretation of the reference interval $C_A(T)$ and the ideal synchronization interval T. They do not know the real synchronization interval itself. However, as for communication purposes, internal synchronization is usually sufficient and the nodes can treat the measured interval as the ideal interval. This leads to the fact that the nodes do not synchronize to real time $C(t)$, but to the reference's interpretation of the real time $C_A(t)$. For the previous example, we can therefore redefine node B's point of view as

$$C_B(C_A(T)) := C_B(T) \,. \tag{2.26}$$

This will be false in the majority of cases and only holds true if

$$C_A(T) = C(T)\,. \tag{2.27}$$

However, for the purpose of internal synchronization, the redefinition does not introduce any error, as long as node A is defined as holding the reference clock.

Knowing both, T and $C_B(T)$, node B can simply calculate its difference from the reference clock

$$D(T) = T - C_B(T)\,, \tag{2.28}$$

wherein a positive $D(T)$ denotes a slow clock compared to the reference and a negative $D(T)$ a fast clock. Furthermore, by knowing the time interval T over which this clock difference occurred, node B has the possibility to estimate its drift rate relative to the reference node and thereby translate any point in time to its own timebase, using

$$C_B(t) = t - \frac{D(T) \cdot t}{T}\,. \tag{2.29}$$

This translation is useful, when it comes to communication schedules, as the synchronization reference, which should be the clusterhead of a communication cluster, can schedule communication in terms of its own timebase. All nodes willing to communicate will then translate their allocated communication times to their local timebase. Hence, all nodes will communicate in a schedule, based on the reference timebase, avoiding overlapping communications (i.e., collisions). Additionally, local measurement of the reference synchronization period allows for the spending of unused time in a low-power mode, while waking up precisely for the reception of the next synchronization message (i.e., the start of a new communication frame) or any other activity scheduled to be performed. A general time-line of the synchronization procedure is given in Figure 2.9. This basically provides a closer look at what happens each time a synchronization message is sent in Figure 2.8.

With ideal synchronization, waiting for the synchronization packet reception (marked blue in Figure 2.9) would not be necessary. However, be-

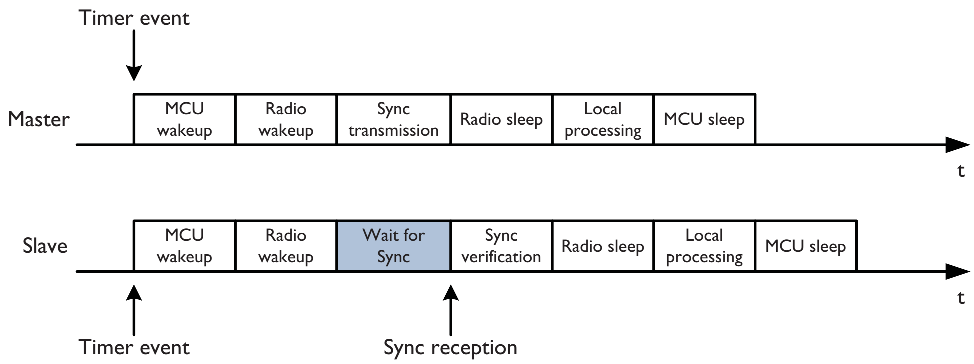


Figure 2.9: General sequence of tasks performed by synchronization master and slave at the beginning of each communication frame

cause there are variations in local clock frequencies due to environmental parameters, quantization errors in reading and setting timer events, as well as variations in component wakeup times, a certain guard-time is desired. This guard-time should guarantee, that the node is in a state allowing the reception of the synchronization message once it occurs. Nonetheless, at the same time the guard-time should be as short as possible, while fulfilling the before-mentioned task.

An important measure to define the length of the guard-time is the accuracy of the wake-up time. This value bases on, but is not limited to, the synchronization accuracy. Figure 2.10 provides the measurement results of the wake-up accuracy for consecutive synchronization messages. Synchronization packets are sent periodically from the reference and received at the slave according to Figure 2.8. Radio wake-up at the slave is scheduled to be completed at a defined time period T_{guard} before the synchronization packet is received. Wake-up accuracy is measured by measuring the deviations from the actual wake-up time to T_{guard}. In order to conduct the experiment, protocols for both master and slave have been implemented in Embedded-C code on the Sentio-e² platform. Measurements were performed using a time accurate logic analyzer [40] and have been repeated for different synchronization interval lengths. Statistical results of these measurements are summarized in Table 2.7. However, due to the limited number of samples, the estimation of the statistical values also has a limited confidence.

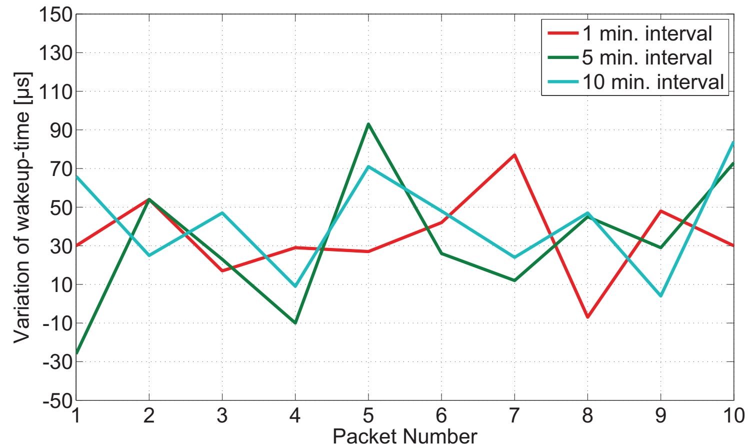


Figure 2.10: Deviation of measured to scheduled time interval between *radio-ready-to-receive* and synchronization message reception for different synchronization interval length

The major sources of inaccuracy in the synchronization result from the interval measurement process. Because the time measurements in low-power mode are taken with a energy conserving auxiliary oscillator, which is sourced from a 32 kHz tuning-fork crystal [41], the maximum time resolution achievable is about 30.5 µs. This means, that a time period might be measured up to one clock cycle (30.5 µs) shorter than it actually is. Additionally, this leads to a shift in the wake-up time, leading to a wake-up of up to one clock cycle earlier than expected. This is the main reason for the offset from the predefined wake-up period, shown in Figure 2.10 and Table 2.7. Moreover, adjustments can merely be taken in the number of clock cycles and thus allow no higher resolution. The results further show a maximum variation of about 100 µs (3 clock cycles) between two consecutive wake-ups. In addition deviation, does not appear to depend on the interval length, which is an expected result due to frequency offset compensation. However, these measurements were obtained under laboratory conditions (i.e., rather constant environmental parameters). In real deployments, increasing interval length also increases the probability of environmental conditions to change. This can have a negative impact on the accuracy due to frequency drift.

One of the main influences on crystal frequency drift, as mentioned at an earlier stage, is crystal temperature, which is directly influenced by environmental temperature. Typically used crystals for low-power time keeping are 32 kHz tuning-fork crystals, which are cut in such a way as to provide a parabolic temperature dependence. This leads to a temperature dependent clock frequency of

$$f_{32kHz}(T_{env}) = \left[1 - B \cdot (T_{env} - T_0)^2\right] \cdot f_{32kHz}(T_0)\,, \tag{2.30}$$

where T_{env} is the current environmental temperature, T_0 the turnover temperature and B the parabolic coefficient of the used crystal. Commonly mentioned parabolic coefficients are of the order of $-0.04 \cdot 10^{-6}\ ^{\circ}C^{-2}$ with a typical turnover temperature of 25 °C.

Figures 2.11a, 2.11c, 2.12a and 2.12c show the effect on wake-up deviation under changing temperature conditions for both, warm and cold temperature environments. While environmental temperature usually changes slowly, there are situations in which sudden temperature changes can occur. One example is depicted by the measurement results in Figure 2.13. In this case, a sensor node was enclosed in a typical housing unit made of light-colored plastic, then placed in direct sunlight. Similar situations might occur when a node was situated in the shade for a while, before being exposed to direct sunlight. The temperature inside the box was logged using an on-board temperature sensor with a rate of 1 Hz over a period of 40 min. It can be seen, that a rapid temperature change in the first minutes after exposure can be expected.

Knowledge of the temperature of the clock crystal enables compensation to be made for the drift occurring due to deviation from turnover temperature. Implementing a low-power temperature sensor on the sensor node is inex-

Interval length [min.]	Mean Value [µs]	Standard Deviation [µs]
1	35	22.56
5	32	36.07
10	42	26.69

Table 2.7: Summary of statistical results obtained from measurements shown in Figure 2.10

pensive and is often required for calibration or application purposes. When used for temperature drift compensation, however, close spatial location to the crystal should be ensured in order to achieve as accurate readings of crystal temperature as possible.

An initial measurement of clock drift over increasing temperature for the Sentio-e^2 platform has been conducted, using its on-board temperature sensor (see Figure 2.14). The graph shows a comparison of measured drift-rate and theoretical drift-rate according to datasheet parameters, provided by the crystal manufacturer [41]. As the match of these two showed some accuracy limitations, an estimation of clock drift based on a calibrated second order polynomial was further added. The estimation is obtained in the form

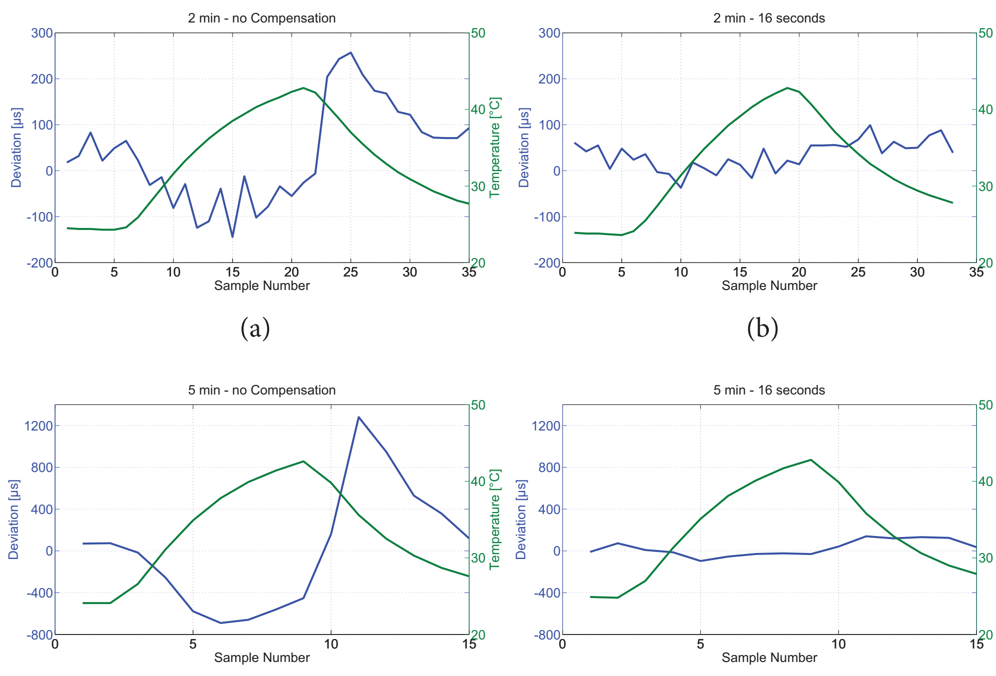


Figure 2.11: Measurement of wake-up accuracy under changing temperature conditions in the oven – (a) 2 min synch. interval, uncompensated; (b) 2 min synch. interval, compensated; (c) 5 min synch. interval, uncompensated; (d) 5 min synch. interval, compensated

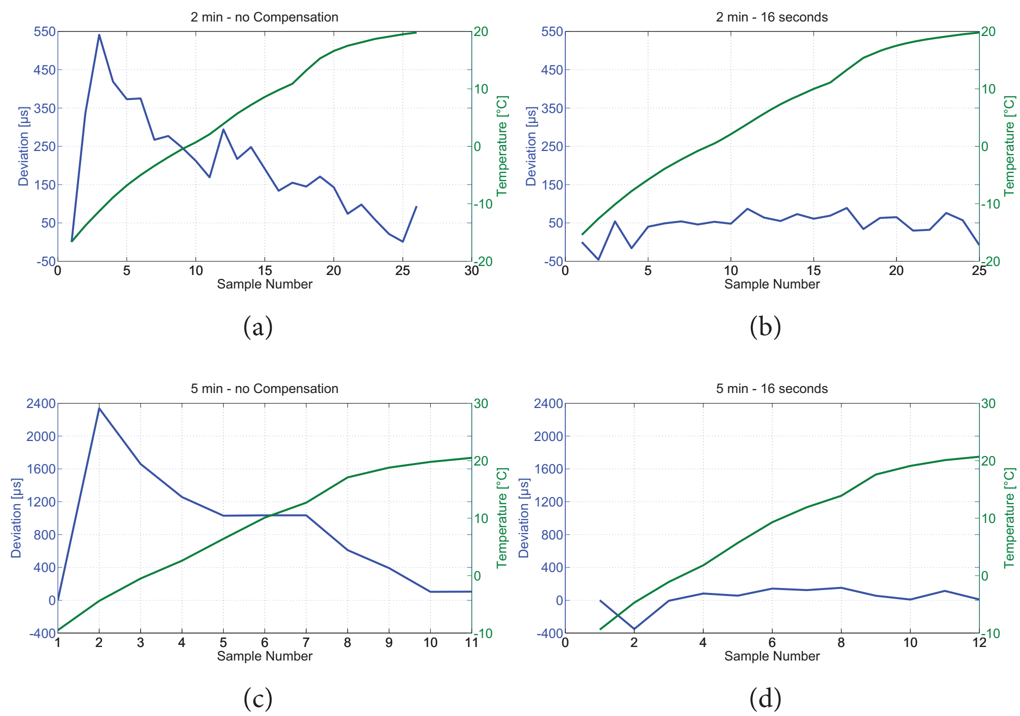


Figure 2.12: Measurement of wake-up accuracy under changing temperature conditions after cooling in the freezer – (a) 2 min synch. interval, uncompensated; (b) 2 min synch. interval, compensated; (c) 5 min synch. interval, uncompensated; (d) 5 min synch. interval, compensated

of

$$\Delta_{est}(T_{env}) = a \cdot (T_{env} - T_0)^2 + b \cdot (T_{env} - T_0) + c\,, \qquad (2.31)$$

where Δ_{est} is the estimated drift and a, b, c are the calibration parameters. The results show, that if high accuracy is required, an initial calibration run can provide accuracy within 1 ppm. However, in the case where the calibration is too time and cost expensive, datasheet values can be used to predict occurring clock drift with restricted accuracies. For measurements shown in Figures 2.11 and 2.12 a calibrated temperature compensation method has been implemented. Comparing these results with the uncompensated ones illustrates, that temperature dependent clock drift can almost completely

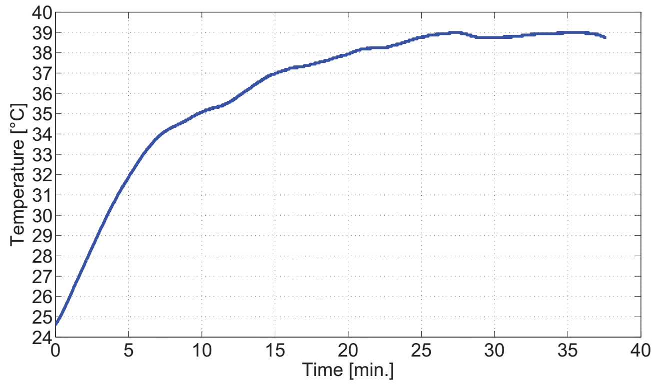


Figure 2.13: Measurement of in-box temperature of a light-colored plastic enclosure under direct sunlight in summer conditions; location Sundsvall/Sweden; measurement with on-board temperature sensor [42] of Sentio-e², 1 Hz

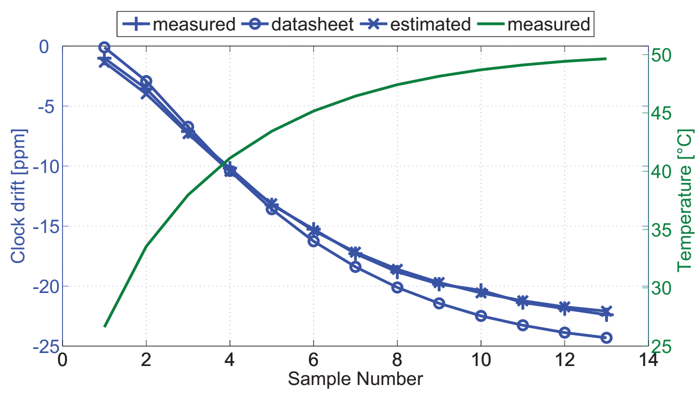


Figure 2.14: Comparison of measured, estimated and datasheet temperature drift behavior; measured with on-board temperature sensor of Sentio-e²; determined from drift over 1 minute while heated in an oven

be eliminated by using a temperature compensation technique based on crystal temperature measurement.

Nonetheless, measuring the crystal temperature adds additional costs to the synchronization algorithm. Sampling the temperature sensor periodically during the inactive period requires the processor to wake-up and read the current temperature value. Furthermore additional active time is mandatory to process the taken values and calculate the expected drift according to equation 2.31. This has to be accomplished once the last temperature reading has been sampled and before the radio wake-up should occur, allowing for an adjustment of the scheduled wake-up time. These necessary activities will increase the microcontroller duty-cycle and therefore lower the energy-efficiency. On the other hand, temperature drift can also be handled by applying longer guard-times, while not compensating for changing temperature. While this consumes no sampling related energy, the radio module has to be in reception state for a longer time, thus spending more energy. As temperature drift has to be dealt with, the question remains which of the methods is the more energy-efficient one.

Estimating additional energy costs by adding expected power consumption levels and periods for each of the methods can provide the answer. This leads to

$$P_{tmpComp} = \frac{T_{meas} \cdot N_{meas} \cdot P_{meas}}{T_{frame}} + T_{process} \cdot P_{process} \tag{2.32}$$

for a temperature compensating approach. In this case, T_{meas} is the time period for taking a sample from the temperature sensor, N_{meas} are the number of temperature samples in one synchronization frame, P_{meas} the consumption level during a sample, T_{frame} the time of a synchronization frame, while $T_{process}$ and $P_{process}$ are the time and power consumption of processing the temperature drift respectively.

Likewise, the consumption for providing an additional guard-period can be summed up as

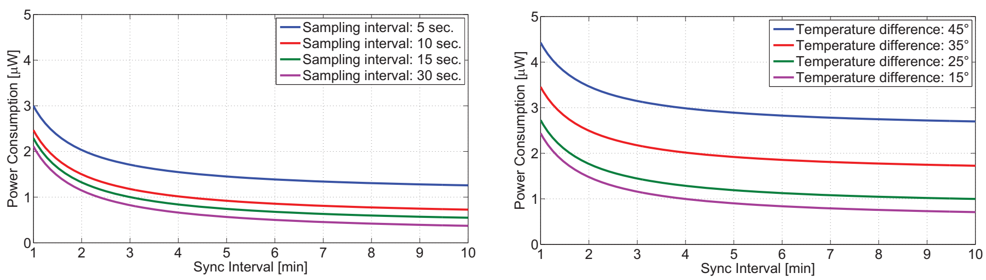
$$P_{guard} = \Delta(\theta_{dif,max}) \cdot T_{frame} \cdot P_{RX} \, , \tag{2.33}$$

with P_{RX} being the power consumption level in reception state, T_{frame} the synchronization period as above, and $\Delta(\theta_{dif,max})$ the drift rate, determined by the maximum temperature difference from the turnover temperature.

An estimation was conducted with parameters, measured on Sentio-e[2] and listed in Table 2.8. The results obtained are shown in Figure 2.15, covering both solutions. In these figures, the power consumption of synchronization reception is additionally included, providing a complete energy cost with regards

Parameter	Value	Description
T_{meas}	950 µs	Time for measurement of a temperature sample
T_{frame}	60 s to 600 s	Time of the synchronization interval
$T_{process}$	150 µs	Time for processing the temperature drift
T_{sync}	2 ms	Time for receiving the sync packet
P_{meas}	2 mA	Consumption during temperature sampling
$P_{process}$	2 mA	Consumption during processing
P_{RX}	20 mA	Consumption when radio in reception state

Table 2.8: Parameter overview for estimations in Figure 2.15, obtained from measurements on Sentio-e²



(a) (b)

Figure 2.15: Comparison of power consumption needed to deal with temperature drift – (a) Temperature compensation with different temperature sampling rates; (b) Extending reception guard-time for different maximum deviations from the turnover temperature

to using the temperature-robust synchronization algorithm. The calculations show, that temperature compensation is the more energy-efficient solution in almost all cases. Providing enlarged guard-periods is only recommended, if during the whole deployment time no large temperature variations are expected.

Moreover, integrated Real-Time Clock (RTC) circuits become an essential part in many products and offer ever increasing functionality. If additional system cost is bearable, RTC circuits with integrated temperature compensation [43] can provide further improvements. While these solutions will have little influence on energy efficiency compared to software temperature compensation, their usage can have tremendous impact on robustness and ease-of-use.

3 Extending Lifetime by Changing Energy Supply

While the previous chapter was mainly focused on node-level energy consumption in Wireless Sensor Networks, this chapter will address the energy source and energy supply as a factor, influencing the system lifetime. The focus in this case lies on Energy Harvesting (EH) – the supply from ambient energy sources – and, in particular solar energy harvesting is addressed as an alternative or supporting energy source. Nevertheless, many of the issues presented and addressed for solar energy harvesting can also be applied to some extent to EH in general.

It will be reasoned as to why Energy Harvesting is a desirable alternative to traditional energy sources, such as rechargeable or non-rechargeable batteries. Furthermore, challenges arising due to the use of EH will be addressed and possible solutions presented. These challenges can be divided into two levels, namely the *deployment-level* and the *planning-level*. The deployment-level challenges are challenges that are connected to the usage of EH-systems, covering for example harvesting efficiency, availability and energy buffering. As opposed to this, planning-level challenges are related to planning and dimensioning of these systems. These challenges include decisions on architecture, buffer capacity or converter size.

3.1 Solar Energy Harvesting Motivation

Although sensor nodes in Wireless Sensor Networks can typically be considered as low-power devices, and therefore consume rather small currents over time, any finite energy source will provide energy for only a limited amount of time. As mentioned previously, this contradicts with the desire for autonomous measurement systems to operate for as long as possible.

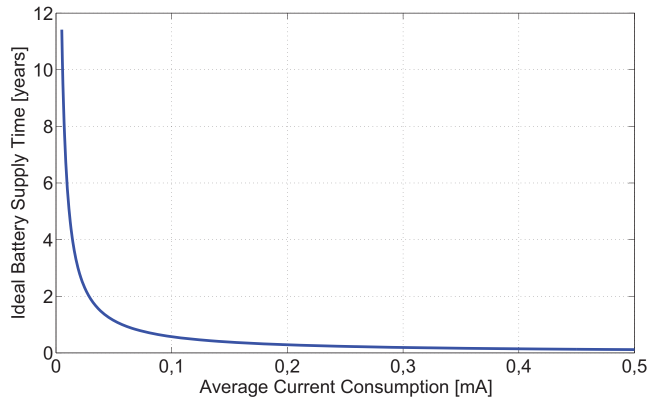


Figure 3.1: Simplified energy supply time from an AA-type battery in ideal situation

Figure 3.1 shows a simplified analysis of battery lifetime for varying load currents. Simplification is made in several ways, namely the nominal battery capacity (in this case 2000 mA h) is assumed to be the real capacity, that the environment will not degrade capacity, no self-discharge is included and no energy is spent for voltage conversion inefficiencies (here assumed from 1.5 V to 3 V). Hence, the illustrated case is a case presenting a very optimistic lifetime and a lower lifetime should be expected in reality.

Nevertheless, this qualitative description already shows, that the lifetime for sensor nodes, powered by batteries, can be very limited and decreases very dramatically with increasing consumption. One possible solution is to increase the number of batteries and thus accumulating capacity. However, this only partially solves the problem, as the available capacity will remain limited. In addition, this solution introduces some undesirable drawbacks. These include the linearly increasing size and cost of the system caused by the increasing number of batteries. Additionally, there is an increasing strain placed on the environment, especially if systems are deployed in locations where it is impossible to recover the system. Alternatively, rechargeable batteries can be implemented, thus reducing the number of battery exchanges. However, recharging still demands a considerable amount of manpower and is not usually a feasible option in the field and the usable capacity is reduced due to the greatly increased self-discharge rates which may be as high as 10 % to 20 % [44]. Moreover, rechargeable batteries display more rapid effect of degradation, due to the chemical process which occurs during the charge and discharge at the electrodes. This leads to a reduced component lifetime for these batteries, typically to a few years for commonly used technologies.

A desirable solution is then to use an energy storage device with limited capacity, that can be autonomously recharged by using ambient energy sources. As ambient sources, such as wind or sun, can be classified as inexhaustible energy sources – meaning that they have no capacity limit – it becomes possible to perpetually replenish a used storage device with limited capacity. From the load perspective, an unlimited energy supply has thus been attached. Nevertheless, for this to function, the conversion rate, consumption rate, storage capacity and storage lifetime have to be properties which are all matched. This means, that while the energy can be perpetually supplied from the ambient sources, this supply is limited to certain rates.

Table 3.1 provides an overview of ambient energy sources, commonly used in Energy Harvesting with their respective power densities. While conversion from solar and wind both provide sufficient power densities, solar energy conversion possesses some advantages over wind. Although both technologies can be considered as relatively mature conversion techniques, solar energy conversion is additionally readily scalable in size. Furthermore, solar energy is less restricted to placement and involves no mechanical parts, which are exposed to limited component lifetime and/or require maintenance. These are the main reasons for the popularity of solar energy harvesting in outdoor sensor networks.

The following sections will concentrate on presenting and addressing the challenges involved in using solar energy harvesting in environmental

Energy Source	Power Density	Condition
Solar	$100\,\mathrm{mW\,cm^{-2}}$	Direct sunlight, outdoors
Wind	$100\,\mathrm{mW\,cm^{-2}}$	$\approx 9\,\mathrm{m\,s^{-1}}$ wind speed, $10\,\mathrm{m}$ altitude
Ambient RF	$< 1\,\mathrm{\mu W\,cm^{-2}}$	Not close to emission source
Thermo-Electric	$60\,\mathrm{\mu W\,cm^{-2}}$	$\approx 5\,^{\circ}\mathrm{C}$ temperature difference

Table 3.1: Typical energy harvesting sources and their power levels available in outdoor environments; based on [45, 46]

Wireless Sensor Networks, in the previously described forms.

3.2 Harvesting Solar Energy

Ambient energy sources have characteristics which are different to those traditionally used for energy sources in WSNs. When using environmental energy to reliably supply sensor nodes, these characteristics have to be borne in mind and handled. While for example batteries are limited in their capacity and thus in their energy supply capability, the supply during the lifetime of the battery (i.e., the time from begin of operation to the battery's energy depletion) is constant and predictable. This is typically not the case in sources used for Energy Harvesting.

Classification of these sources can be conducted according to two parameters – controllability and predictability of energy supply [47]. Although there are controllable EH-sources, such as indoor light or an RF signal, sources used for Energy Harvesting in outdoor Environmental Monitoring are mostly uncontrollable. Thus, it is not possible to control how much energy can be harvested from the source at any given time, or whether it is indeed possible to harvest energy at all. Similarly, predictability can vary from source to source.

In solar energy harvesting, the energy source can be considered as uncontrollable, but predictable. While it is not possible to control how much energy can be harvested from the sun at a certain time (e.g., at nights or cloudy days lower levels have to be accepted), to some extent harvest-able energy levels can be predicted. Predictability will certainly be prone to some error, but general tendencies according to daily or seasonal behavior, as well as weather forecasting can be provided.

3.2.1 Energy Neutral Operation

Despite the variation of supply levels, that can be harvested from the sun, it is desirable to have an uninterrupted supply of the attached system (i.e., the sensor nodes). Therefore, the load power consumption has to match the power supply rate of the harvesting system. A metric, called the energy neutral operation [48], can be used to describe the relationship between consumption and supply, providing energy to the load at all times. Discussions concerning an energy neutral operation only make sense in connection with energy sources of infinite energy supply capability, while being bounded by certain supply rates. This for example means, that it is not possible for a primary battery to operate in an energy neutral manner, because its energy reservoir is finite.

The simplest form of an energy neutral operation is the direct connection of the EH-source to the load. In this case the supplied power has to exceed the consumed power at any time, such as

$$P_s(t) \geq P_c(t)\,, \tag{3.1}$$

with $P_s(t)$ being the supplied power and $P_c(t)$ the consumed power at time t respectively. However, in this scenario all the energy of times $P_s(t) > P_c(t)$ and $P_s(t) < P_c(t)$ is wasted. Moreover, in solar energy harvesting this solution does not provide an uninterrupted supply, since $P_s(t)$ regularly becomes zero (e.g., at night), while $P_c(t)$ in real systems is always greater zero.

As an uninterrupted operation is a desired system property, for solar energy harvesting systems some sort of energy buffer has to be implemented, so that

$$P_s(t) + P(W_b, t) \geq P_c(t)\,. \tag{3.2}$$

In this case $P(W_b, t)$ described the power that can be supplied by the storage device at time t, depending on the energy W_b which is simultaneously stored in the buffer. According to [48], over the whole system lifetime an energy neutral operation requires the following condition:

$$\begin{aligned} B_0 + \eta \int_0^T [P_s(t) - P_c(t)]^+ \, dt - \int_0^T [P_c(t) - P_s(t)]^+ \, dt \\ - \int_0^T P_{leak}(t) \, dt \geq 0 \quad \forall T \in [0, \infty) \end{aligned} \tag{3.3}$$

with B_0 being the initial energy stored in the buffer, P_{leak} the losses of the buffer, η the charging efficiencies, and $[x]^+$ the rectifying function

$$[x]^+ = \begin{cases} x & \text{if } x \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{3.4}$$

This does not include any capacity limitations of the used storage device, which in real systems is an important characteristic, leading to a waste of harvested energy. To account for this, we can further substitute $P_s(t)$ in equation 3.3 with

$$P'_s(t) = \begin{cases} P_s(t) & \text{if } B(t) < 100\,\% \\ 0 & \text{otherwise} \end{cases} \tag{3.5}$$

where $B(t)$ is the charge level of the buffer at time t.

3.2.2 Solar Energy Harvesting Architectures

The desire for an uninterrupted operation over the whole system lifetime, and therefore an energy neutral operation, means that it becomes necessary for there to be an intermediate circuit between the energy converter and the sensor node, as shown in Figure 3.2. In the simplest case, this intermediate circuit merely consists of the previously mentioned energy buffer and related charging circuitry. However, further functions are usually either necessary or desired, and the implementation of the intermediate circuit can differ tremendously.

For solar energy harvesting, however, certain basic modules, contained in the intermediate circuit, can be defined. A typical structure is provided in Figure 3.3, which contains input regulation, storage and output regulation. In this case the main purpose of the input regulation is in matching the output of the solar panel to the requirements of the energy buffer. This typically means that the voltage levels will be adjusted, the energy flow will be rectified to protect the solar panel and, additionally, the power level extraction will be optimized. The energy buffer, as addressed previously, stores energy in

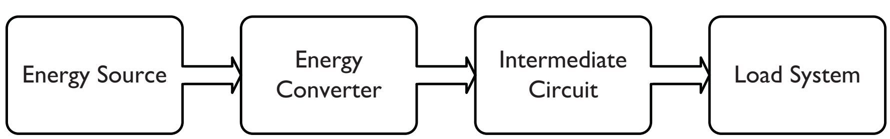


Figure 3.2: Basic architecture of a system powered by energy harvesting from an ambient energy source

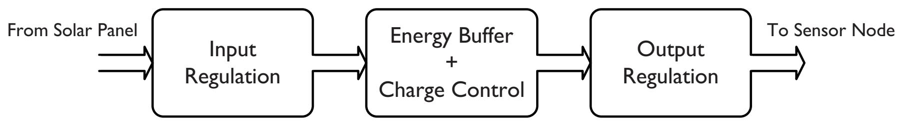


Figure 3.3: Typical structure of the intermediate circuit for solar energy harvesting systems

times of ambient energy excess and which supplies the load system from this energy in times of insufficient ambient levels. Typical storage devices used (e.g., rechargeable batteries and supercapacitors) require additional charge control so as to protect the buffer from damage. The purpose of the output regulation is to adjust the energy buffer output to the input requirements of the attached sensor node. If more than one energy buffer is used, this might also include some selection circuitry which will automatically supply the load from the correct source.

While this basic structure is usually the same in all systems, implementation of the individual stages can differ, depending on the focus and application of the systems respectively. The choices for the storage devices usually consist of rechargeable batteries or supercapacitors (also known as ultracapacitors or DLCs). Examples for these systems are [50, 51] for lithium-based recharge-

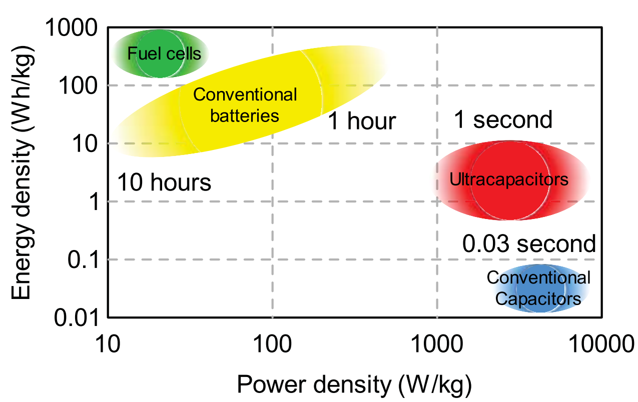


Figure 3.4: Classification of energy density and power density of energy storage devices [49]

able batteries, [52, 53, 54] for nickel-based rechargeable batteries and [55] for supercapacitor-based systems. Additionally, some systems attempt to accommodate the advantages of both storage types by implementing hybrid storage solutions, such as in [56, 57]. Figure 3.4 depicts the relationship between these storage devices.

Although rechargeable batteries provide larger energy densities and have smaller self-discharge rates, leading to energy storage capabilities for longer periods in comparison to supercapacitors, they have a very limited amount of charge/discharge cycles and thus have a limited lifetime. However, in an apparent contradiction, supercapacitors provide long lifetimes due to charge/discharge mechanisms based on charge separation instead of chemical reactions. Furthermore, supercapacitors have higher power densities than batteries, which means that they can be charged and discharged at high rates. Nevertheless, supercapacitors suffer from relatively high discharge rates and low energy densities, which makes them usable only as short-term buffers. This can introduce problems during longer periods of limited solar radiation. While Lithium-Ion (Li-Ion) batteries provide some beneficial advantages, such as very low self-discharge, high energy density and nominal voltage levels that typically do not require adaptation to sensor node levels, one major problem associated with this type is its rather complex charging behavior. It is usual to have a dedicated charging circuit when implementing Li-Ion batteries to allow for their proper use, which introduces extra costs to the system. Table 3.2 provides an overview of the characteristics of the different rechargeable battery types. It is not mentioned in the table, but an important parameter for autonomous systems, is that all types apart from the Li-Ion require regular maintenance of the order of months in order to achieve the mentioned parameters.

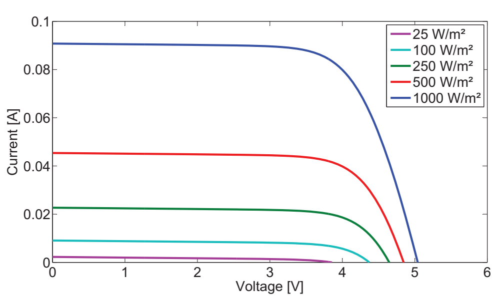
Despite the individual charging mechanisms and protection circuits, input regulators are often used to increase harvesting efficiencies by forcing the solar panel to operate at the Maximum Power Point (MPP). Solar panels behave like a voltage-dependent current source, where the current is a function of the irradiance level. This leads to a typical current-voltage characteristic (often referred to as a solar panel's IV-curve) as depicted in Figure 3.5a. For good quality solar cells (i.e., small serial resistance and large shunt resistance) the current remains almost constant until a certain voltage is reached, then breaks down rapidly until reaching zero. When charting the power levels

Type	Voltage [V]	Energy Density [W h kg^{-1}]	Self-discharge [%/month]	Cycles [#]	Toxicity
Lead-Acid	2	30-50	5	200-300	high
NiCd	1.2	40-80	20	1500	high
NiMH	1.2	60-120	30	500	low
Li-Ion	3.6	100-150	< 10	1000	low

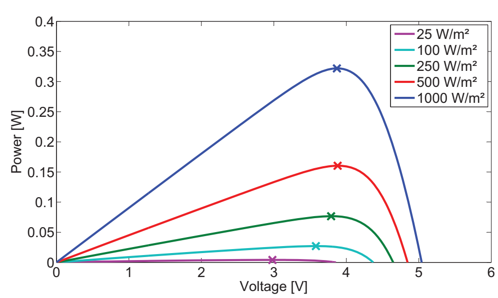
Table 3.2: Overview of main characteristics for different, typically used battery types; based on [44, 58]

for each of the respective voltage levels, as shown in Figure 3.5b, a clear voltage of maximum power is obvious (the MPP). Therefore any solar panel has three distinct parameters, the open-circuit voltage V_{oc}, the short-circuit current I_{sc} and the Maximum Power Point (V_{mpp}, I_{mpp}). As solar cells are rather costly and might increase the system size, the energy extraction should occur as efficiently as possible. Estimating the MPP voltage and forcing the solar panel to operate at this voltage is referred to as Maximum Power Point Tracking (MPPT).

Nevertheless, performing MPPT also comprises some system costs. On the one hand additional circuitry is required, which will increase the size and monetary costs, on the other these required functionalities will consume



(a)



(b)

Figure 3.5: Solar panel characteristics of a 450 mW panel with 9 cells in series at different irradiance levels – (a) I-V characteristic and (b) P-V characteristic

energy themselves. The latter cost, in particular, depends on the accuracy, and therefore the method, of the MPPT. Accurate methods, such as the *hill-climbing* method [59, 60], continuously measure the power output of the solar panel, while slowly adjusting the terminal voltage. While this allows it to operate at, or very close to, the Maximum Power Point of the solar panel, this method is very costly, because measurements and adjustments are continuously performed. Furthermore, depending on the step-size, the method can be rather slow to react to abrupt changes (e.g., due to abruptly changing irradiance conditions). Thus, a compromise has to be made for the perturbation step-size, as a small step-size leads to long reaction times, while a large step-size leads to a strong influence from the oscillation around the actual MPP.

Because solar panels, typically used to supply sensor nodes, are of limited area and power output, the overhead costs of the Maximum Power Point Tracking solution have a considerable impact on the system efficiency. Therefore hill-climbing methods (also similar methods, such as Perturb-and-Observe (PnO)) are too costly to implement. Simpler, less accurate methods are usually used, such as *fractional open-circuit voltage* or *fractional short-circuit current* [61]. These two methods are based on the approximately linear relationship between the MPP and the open-circuit voltage V_{oc} and the short-circuit current I_{sc} respectively. This leads to

$$V_{mpp} \approx k_V \cdot V_{oc} \tag{3.6}$$

E [W m^{-2}]	V_{mpp} [V]	I_{mpp} [mA]	V_{oc} [V]	I_{sc} [mA]	k_V	k_I
1000	3.87	83.18	5.04	90.77	0.767	0.916
500	3.88	41.34	4.84	45.39	0.810	0.910
250	3.79	20.22	4.64	22.69	0.816	0.891
100	3.58	7.57	4.37	9.08	0.819	0.833
25	2.98	1.39	3.85	2.27	0.774	0.612

Table 3.3: Overview of measured characterization parameters of a 450 mW silicon solar panel

and

$$I_{mpp} \approx k_I \cdot I_{sc} \,, \tag{3.7}$$

where V_{mpp} and I_{mpp} are the voltage and current at the Maximum Power Point, while k_V and k_I are the proportionality factors, approximated as being constant for varying environmental conditions. However, it has to be borne in mind that this is only an approximation, as shown in Table 3.3. In this case it can be seen that the factors k_V and k_I change with the irradiance level. The achieved accuracy is still relatively good as can be seen through the effect on the MPP estimation, shown in Figure 3.6. Estimation did occur by averaging k_V and k_I respectively. As accuracy at lower irradiance levels is not of significant importance (because the output is very low anyway), accuracy at higher irradiance levels could further be improved by removing low-irradiance numbers from the averaging process. In addition, these results show that for the particular solar panel under test, the fractional open-circuit voltage method provides better estimations in most cases. Nonetheless, the mentioned factors are solar panel dependent, meaning that each individual panel has to be tested in order to achieve the optimal outcome. An analysis of panels from the same type can provide an idea of variations within the same manufacturer. If this is not too great then, this could allow for some generalizations. Furthermore, although this method is usually less costly

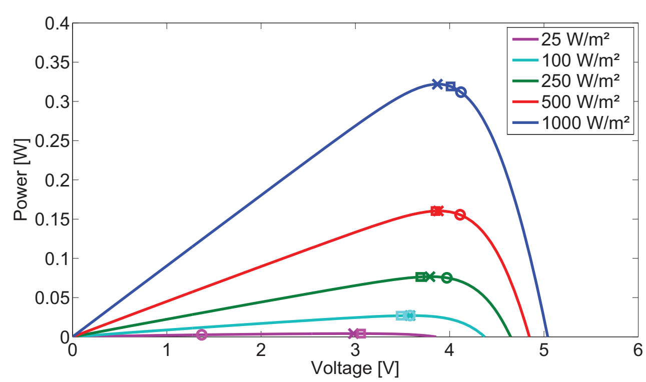


Figure 3.6: Accuracy of MPPT based on fractional open-circuit voltage and fractional short-circuit current; real MPP marked by cross, estimated from V_{oc} marked by squares, estimated from I_{sc} marked by circles

than the hill-climbing methods, there are still necessary costs associated with measurement and adjustment. To be able to react to changing irradiance conditions, sampling and adjustment rates should be quite high.

As opposed to the input regulation, the output regulation in most systems is rather straight-forward. While systems based on Li-Ion batteries often require no output regulation, since the battery voltage is rather high, most other storage types require a boost for their voltage levels in order to fulfill load demands. In the majority of cases this involves the implementation of a standard step-up regulator which boosts the battery voltage according to

$$V_{out} = V_{bat} \cdot \frac{I_{bat}}{I_{out}} \cdot \eta \,, \tag{3.8}$$

where V_{out}, V_{bat}, I_{out} and I_{bat} are the voltages and currents of the battery and the regulator output respectively, and η is the conversion efficiency. While consumption of these regulators can be relatively low, the converter efficiency is an important parameter for the selection process.

3.2.3 Harvesting Solar Energy at Low Irradiance Levels

While many solar energy harvesting systems for Wireless Sensor Networks have been proposed in the literature, the majority of these systems originate from locations where yearly solar radiation is high. Typically, with reference to the design considerations of the systems, several hours of strong sunlight are expected each day [50, 62, 57]. This might be the case in targeted locations, but does not hold true in general. Figure 3.7 provides an overview of the solar radiation during 2008 in Sundsvall, Sweden ($62°24'N, 17°19'E$). In locations such as these, strongly unequal irradiance distribution can be observed. While this might lead, in general, to acceptable solar radiation over the year, the solar irradiance during certain periods in the year can be immensely low. Furthermore, even at locations where solar radiation is usually high, limited irradiance conditions can occur due to shading by objects (e.g., trees and buildings). In situations involving frequently limited irradiance levels, different design considerations have to be applied than those in the previously mentioned systems.

When targeting long system lifetime, a battery-based system is not suitable, because of this previously mentioned limitations with respect to their lifetimes. Hence, Double-Layer Capacitors are the remaining choice for energy buffering. However, because DLCs have only short-term storage capabilities, they do not have the possibility to bridge periods of low irradiance, balancing unequal distributions. In the targeted environment, irradiance conditions can be low over longer periods of time, thus a DLC-based system has to be able to harvest energy even during these periods. As energy income and load are fixed, the energy consumption of the harvesting system itself is the determining parameter of the lifetime under low irradiance conditions. The harvesting architecture should allow for a minimal own energy consumption by the system, while allowing for a proper and safe operation. Two architecture possibilities, capable of providing this functionality, have been investigated and their basic architectural overview is given in Figure 3.8. These architectures have been chosen, because of their simplicity, their limit in the number of components, and their consumption of energy to allow harvesting.

Because both architectures are based on a DLC energy buffer of the same type, their output regulation is also identical. Due to the typically low nominal voltage of supercapacitors (i.e., in this case 2.5 V) and the ideally linear

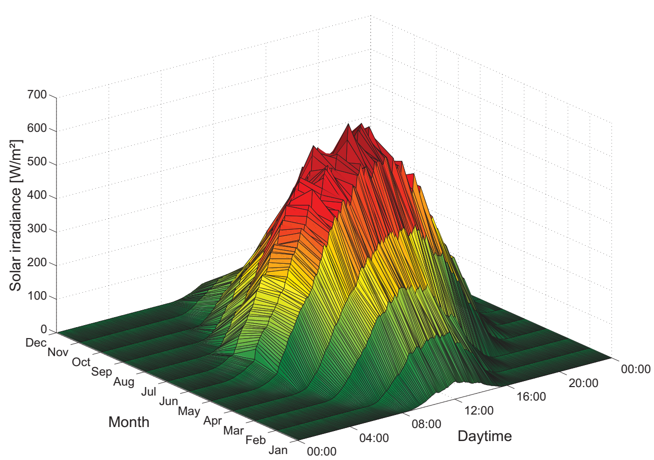


Figure 3.7: Annual solar irradiance distribution in Sundsvall, Sweden 2008; daily data averaged for each month; daytime resolution 5 min

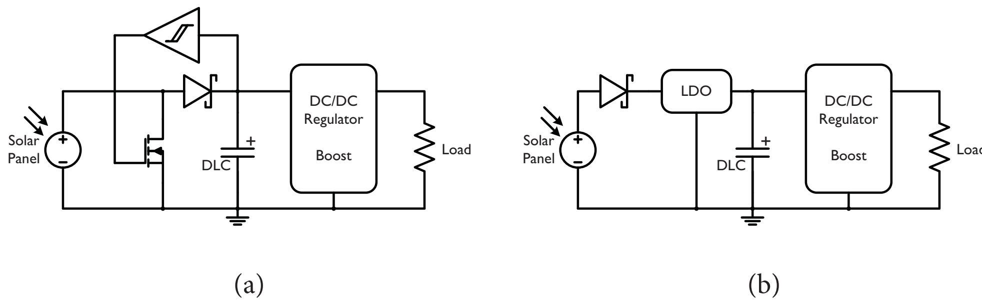


Figure 3.8: DLC-based solar energy harvesting architecture implementations – (a) directly coupled architecture and (b) architecture with LDO input regulation

relationship between the terminal voltage and charge state

$$V_{DLC} = \frac{1}{C} \int I_{DLC} \, dt \,, \tag{3.9}$$

conversion of the terminal voltage to meet the load requirements is necessary. Therefore, both architectures include a DC-DC regulator which includes boost topology. This regulator has to be able to take relatively low input voltages to make use of as much of the stored energy in the DLC as is possible, while also providing satisfactory efficiencies over the whole conversion spectrum. To avoid the implementation of a DC-DC regulator, two alternatives are possible. Several Double-Layer Capacitors can be stacked in series to add their individual voltages to a higher terminal voltage or a DLC with higher nominal voltage can be used. The former case has the disadvantage that charge and discharge rates for individual devices can vary and they do not balance their voltages automatically. This means that the series configuration of two DLCs can be charged to the double nominal voltage V_{nom}, with one device operating at a higher voltage than V_{nom} and the other one at a lower voltage. Due to the internal structure of the Double-Layer Capacitors, the operation at voltages over a specified V_{nom} is not recommended and can lead to a shortened lifetime or even destruction [63]. Therefore in the serial configuration, as mentioned previously, an external balancing circuit would be required, leading to additional energy losses. In addition, using a single

DLC with a higher nominal voltage V_{nom} comes with disadvantages, as either the cost is disproportionally high or one has to live with a higher internal resistance. High internal resistance is not desired, as it changes the charge and discharge behavior from its ideal linear behavior.

For these reasons, the implementation of a DC-DC regulator was regarded as the best possible choice, further allowing for the adjustment of the output to meet different load systems. For the implementation, a Texas Instruments TPS61070 [64] has been chosen.

The difference in the two architectures lies in the input regulation stage. While Figure 3.8a shows a directly-coupled architecture (i.e., an architecture where a solar panel and storage device are coupled directly, without any intermediate stage), Figure 3.8b illustrates an architecture, where an LDO regulator is used for the input regulation. Both systems do not implement Maximum Power Point Tracking, because of its energy overhead cost and the operation is often at low irradiance conditions. As presented at an earlier stage, MPPT is rather costly from the perspective of energy overhead, because frequent measurements of currents and/or voltages have to be obtained, and an adjustable input regulator is required. The additional energy spent however, means that the operation becomes more efficient, in cases where more energy surplus is gained than the invested additional energy costs. As Figure 3.5b shows, lower irradiance conditions result in a flatter P-V characteristic for solar panels. Hence, operating at the MPP provides less gain in harvested power. Additionally, at lower irradiance levels MPPT is typically less accurate.

This leaves the main function of the input regulation to provide protection mechanisms. Therefore, in the second architecture, the output voltage of the LDO should be chosen to match the nominal voltage of the DLC V_{nom}. This allows the Double-Layer Capacitor to be charged to its maximal value, while not risking any damage due to over-charging. Furthermore, when fully charged, the charge level will not decrease as long as a sufficient irradiance level is available. The condition for this is, that the solar irradiance level E is sufficiently high to allow

$$\begin{aligned} I_{solar}(E, V_{DLC}) \geq & I_{LDO}(V_{solar}) + I_{DLC}(V_{DLC}) \\ & + I_{DC-DC}(V_{load}, I_{load}, V_{DLC}) \,. \end{aligned} \tag{3.10}$$

In this case, I_{solar} is the output current of the solar panel, I_{LDO} the current consumption of the LDO regulator, I_{DLC} the leakage current of the Double-Layer Capacitor and I_{DC-DC} the necessary input current for the DC-DC regulator, each with their respective dependencies.

The architecture shown in Figure 3.8a completely avoids the use of any dedicated input regulator. This, however, makes some dedicated protection mechanisms necessary. While both systems require a rectifying-diode to protect the solar panel from reverse currents, this directly-coupled architecture also requires an additional over-voltage protection for the DLC. As mentioned previously, charging the Double-Layer Capacitor to a higher value than its nominal voltage can lead to lifetime reductions or damage. Due to the LDO with matching output voltage, this is not possible in the second architecture, presented in Figure 3.8b. In the directly-coupled architecture, when not limited, charging would occur until the open-circuit voltage V_{oc} of the solar panel is achieved. This has to be prevented without adding too much extra cost. The typical analog component used for this purpose is a Zener-diode. However, because Zener-diodes have non-ideal behavior, extreme losses occur close to its cut-off voltage. This is shown in a behavioral comparison between real and ideal Zener-diode in Figure 3.9.

As these losses are not acceptable in the proposed architecture, ideal Zener-

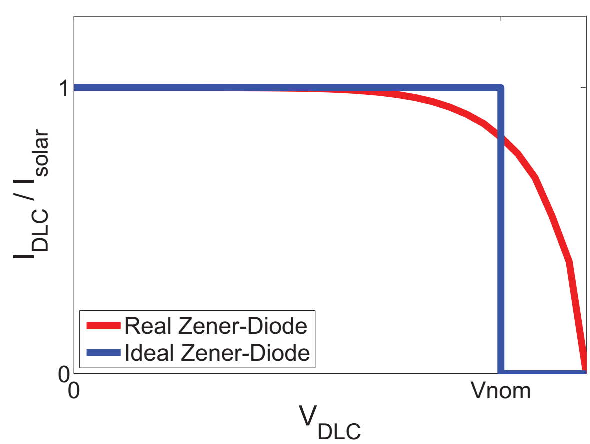


Figure 3.9: Behavioral comparison of ideal and real zener-diode characteristics

diode behavior is replicated by a combination of a MOSFET and a hysteresis comparator (see Figure 3.8a). The comparator checks the actual voltage of the DLC against its reference voltage, which should be set to match the nominal voltage of the DLC. Once the two parameters match, the comparator will drive the MOSFET, which disconnects the power source by shortening the solar panel. Thus leading to the ideal Zener-diode behavior, as in

$$\frac{I_{charge}}{I_{solar}} = \begin{cases} 1 & \text{if } V_{DLC} < V_{nom} \\ 0 & \text{otherwise} \end{cases} \tag{3.11}$$

Until the reference voltage is achieved charging occurs without limitation, then charging is abruptly stopped. The only losses included in the replication are the operating energy consumption of the hysteresis comparator. When choosing a low-power device, the current draw of the comparator can be limited to a few µA.

To investigate the architectures' ability to provide uninterrupted power supply even during low irradiance conditions, both architectures have been implemented and deployed for experimental purposes during winter 2009/2010 (Period: November 2009 to January 2010). At the deployment site in Sundsvall,

Module	Directly-Coupled	LDO-based
Solar Panel	450 mW, 94 × 61 mm²	
DLC	$C = 10$ F, $V_{nom} = 2.5$ V $C = 22$ F, $V_{nom} = 2.5$ V	
Reverse Current Protection	Shottky Diode (MBR0520L)	
Over-Voltage Protection	Comparator (MAX9017) MOSFET (MGSF2N02EL)	—
Input Regulator	—	LDO (TPS71525)
Output Regulator	Boost (TPS61070)	
Load System	Sentio-e²	

Table 3.4: Module configuration for implementation of the two presented architectures

Sweden ($62^\circ 24' N, 17^\circ 19' E$), this covers the darkest period of the year. Thus, when the system operates correctly during this worst-case period, we can conclude that operation during the whole year is possible. Implementation of the two architectures occurred according to the module configuration listed in Table 3.4. During the deployment period, Sentio-e^2 nodes were used to sample and communicate the environment temperature, humidity and the voltage level of the DLCs at 5 min intervals. Additionally, one node measured the solar radiation to correlate the local values with data from the reference sensors, situated on a weather station at the deployment location. The current consumption of the duty-cycled load is about 20 µA at about 3 V.

Characteristic system behavior for both architectures is given in Figure 3.10. The graphs represent one weeks worth of data for the deployed systems. Although both systems behave in a quite similar manner under the same environmental conditions, there are some behavioral differences related to their architectures. In general it can be seen, that both systems replenish their energy buffer during rather short periods of low solar irradiance. However, while the maximum capacitor voltage is limited by the output voltage of the LDO in Figure 3.10b, the limitation in Figure 3.10a is provided by the internal voltage reference of the hysteresis comparator. This leads to slightly varying triggering voltages, in this example of 2.45 V for the 10 F DLC and 2.55 V for the 22 F DLC. Additionally, due to the hysteresis of the comparator, a second charging condition is introduced to the directly-coupled architecture. In the LDO-based architecture charging is purely limited by environmental irradiance conditions. However, the hysteresis comparator only allows charging, when the lower hysteresis bound is crossed. This leads to the bi-daily charge cycle of the 22 F Double-Layer Capacitor, as illustrated in Figure 3.10a. While the hysteresis setting of the comparator prevents oscillation and thereby any strain on the devices, it also can lead to an undesired system behavior. This can, for example, be observed at Day 7 in Figure 3.10a, where the DLC voltage crossed the lower hysteresis bound immediately after the sun had set. Hence, the DLC discharges for another day until it is again recharged. Despite the programmability of the hysteresis band, finding a good configuration for individual conditions and locations is a difficult task. Although the LDO-based architecture avoids these situations, it is also associated with some disadvantages. Because of the connected energy buffer at the LDO output and the intermittent energy source at the input, the LDO regulator has to

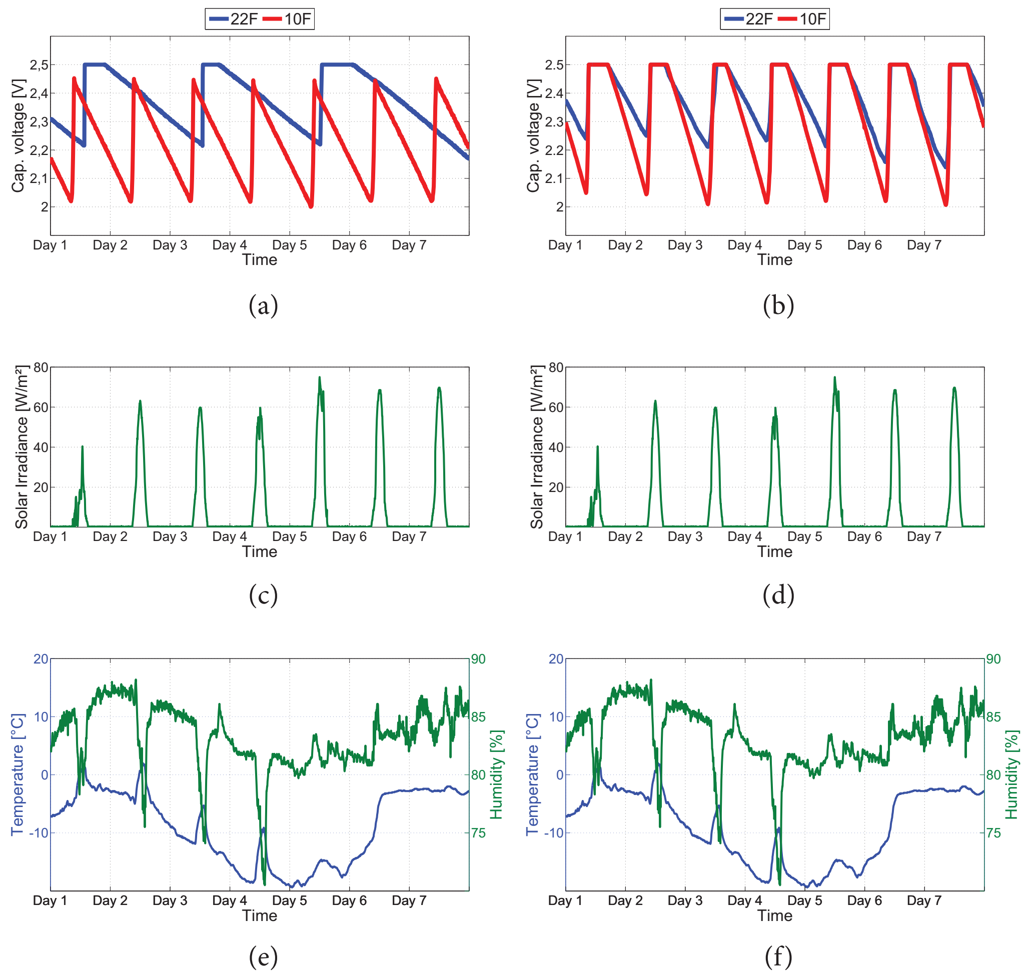


Figure 3.10: System behavior of two solar energy harvesting architectures during one week of deployment – (a),(b) capacitor voltage levels for directly-coupled architecture (left column) and LDO-based architecture (right column); (c),(d) irradiance conditions during the week; (e),(f) temperature and humidity during the week

allow for higher output voltages than the input voltage. For most devices, this is actually not the case, and therefore the choices for this type of regulator are limited. This leads, in particular, to a more limited choice of low-power components, ultimately leading to higher system consumption and thus faster discharge rates (as visible in Figure 3.10b). Furthermore, this architecture is less flexible to changes, such as adding Maximum Power Point Tracking or increased input voltages due to larger solar panels.

3.3 Planning Solar Energy Systems

While, in general, because of their inexhaustible energy source, solar energy harvesting systems allow for a perpetual energy supply, EH systems are limited by the supply rate, as mentioned in section 3.2. Thus uninterrupted long-term supply is only possible when supply rate conditions and consumption conditions are met. Planning of the solar energy harvesting system becomes an essential part in the design process. Figure 3.11 provides an overview of the constraint factors involved in solar energy harvesting systems. The goal of system planning is to determine certain required constraints under otherwise given constraints. A typical exemplary case would be the determination of the necessary harvester dimensions, while the environmental and application constraints are fixed.

This component dimensioning is necessary for harvesting systems, regardless of the particular architecture chosen. Taking the architectures presented in the previous section as an example, straight-forward parameters include DLC capacity and solar panel size. The choice of these parameters depends mainly on the power consumption of the system load and the environmental

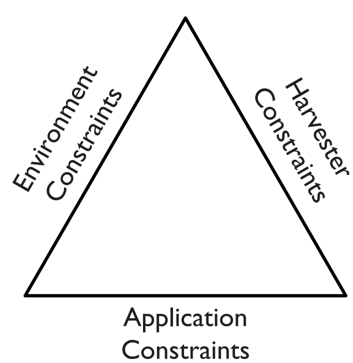


Figure 3.11: Overview of constraint sources, influencing each other, in energy harvesting systems

conditions at the intended location (i.e., especially the irradiance levels). While on the one hand, under-dimensioning leads to unreliable operation, over-dimensioning is not desirable as it leads to an extended system in relation to both cost and size. Even though a systematic method of dimensioning system components is certainly desired, the complexity of the system and the challenge of simplifying location-based influences makes a supporting tool necessary. Using an architectural model, simulating the energy level available at every instance of time, can provide the necessary information for supporting system parameter optimization. The possibility and performance of this type of dimensioning method will be demonstrated in the following section , using the directly-coupled architecture (as in Figure 3.8a) as a case example.

3.3.1 Modeling of the Solar Energy Harvesting Architecture

Figure 3.12 illustrates the energy interactions in the system. Because of the direct coupling between the solar panel and DLC, the voltage of the solar panel will follow the voltage of the energy buffer (with a small offset due to the reverse current protection diode). However, at the same time the current at this voltage depends on the irradiance level. Thus, the operating point of the solar panel at any point in time is given by the irradiance and the

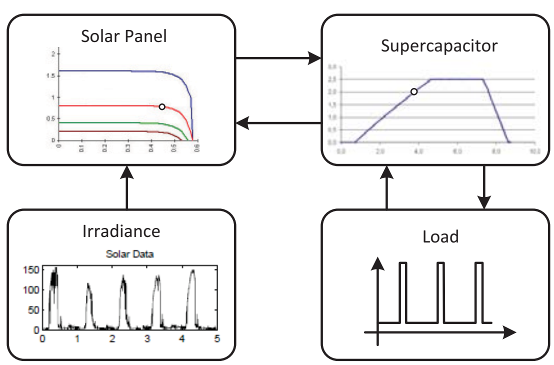


Figure 3.12: Overview of energy-related interactions between modules of the solar energy harvesting system, based on a directly-coupled architecture

DLC charge level. In return, this operating point defines the charge-rate of the DLC, changing its voltage level over time, leading again to a change of the operating point of the solar panel. Additionally, the DLC is discharged continuously by the load system at a rate defined by the load's duty cycle. This influences the charging of the DLC, leading to a resulting charge/discharge rate.

To model this dynamic system behavior, accurate modeling of the involved modules is important. Nonetheless, any unnecessary complexity of the model should be avoided in order to increase the development time and make the model flexible to changes. Therefore, a hybrid modeling approach was chosen, in which the main modules with complex behavior are modeled electrically, while the other modules are simplified using logical behavior. Typical examples for electrically modeled components are the Double-Layer Capacitor and solar panel, while protection functions represent the logically implemented functions.

Implementation has been conducted in Matlab/Simulink, requiring analytical representation of the above mentioned functions. Therefore, electrical models have been implemented according to Kirchhoff's circuit laws. Figure 3.13 illustrates a typically used equivalent circuit of solar panels, based on a single-diode model. From Kirchhoff's laws we can represent output voltage

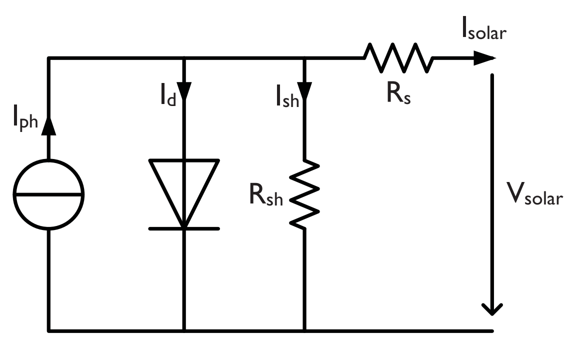


Figure 3.13: Equivalent circuit of a solar panel according to the single-diode model

and current as

$$V_{solar} = V_D - I_{solar} \cdot R_s \tag{3.12}$$

$$I_{solar} = I_{ph} - I_D - I_{sh} \tag{3.13}$$

with notations as in Figure 3.13. Using (3.12) in (3.13) and replacing I_D with Shockeley's diode equation, leads to a more detailed description of resulting output current

$$I_{solar} = I_{ph} - I_0 \cdot \left(e^{\frac{q(V_{solar}+I_{solar}\cdot R_s)}{nkT}} - 1 \right) - \frac{V_{solar} + I_{solar} \cdot R_s}{R_{sh}}, \tag{3.14}$$

where I_{ph} can be deduced as being proportional to the irradiance level, such as

$$I_{ph} = k_{irr} \cdot E\,. \tag{3.15}$$

Substituting I_{ph} in (3.14) with (3.15), the solar panel model is implementable with standard operations. This leaves the key modeling parameters as being the serial resistance R_s, the shunt resistance R_{sh}, the irradiance level E, as well as the proportionality factor k_{irr}. With the exception of the irradiance, these parameters are solar panel properties and can be determined by various methods, e.g. by analyzing I-V curve characteristics [65].

Similarly, the DLC can be implemented according to voltage and current

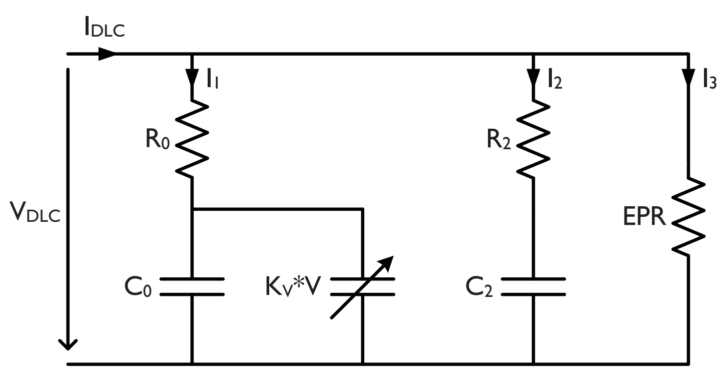


Figure 3.14: Equivalent circuit of a Supercapacitor according to the two-branch model

analysis of the equivalent circuit, shown in Figure 3.14. The circuit shows a two-branch model, which has been introduced in [66]. The two branches represent the long-term and short-term behaviors of the DLC respectively, while leakage is modeled with an additional Equivalent Parallel Resistance (EPR). Following the currents and voltages as described in Figure 3.14, we can describe the model by means of

$$I_{DLC} = I_1 + I_2 + I_3 \tag{3.16}$$

$$V_{DLC} = I_1 \cdot R_0 + \int \frac{I_1}{C_0 + K_V \cdot V_{DLC}} \, dt \tag{3.17}$$

$$V_{DLC} = I_2 \cdot R_2 + \frac{1}{C_2} \int I_2 \, dt \tag{3.18}$$

$$V_{DLC} = I_3 \cdot EPR \tag{3.19}$$

In this case, R_0, C_0, K_V, R_2, C_2 and EPR are constants, which are determinable according to the description presented in [66]. The determination of these parameters, however, has to be repeated for every type and size of DLC to be used in simulations, as simple deduction is not feasible.

Implementing the two main modules and filling connections between them by adding logical representations of protection mechanisms, it is possible to produce a system model of the energy harvesting power supply. At a high abstraction level, this system model can be described as depicted in Figure 3.15. While architectural parameters are defined inside the system model, such as solar panel type or DLC capacity, external conditions are provided as inputs to the model. The final output of the model is the voltage level of the Double-Layer Capacitor at any simulation point, e.g. allowing for there to be an estimate for the amount of time when there is insufficient energy under given conditions.

3.3.2 Simulating Available Energy Levels

Before the model was used for simulations, an evaluation has been conducted against a real-world deployment of the same architecture, as described in section 3.2.3. The results of the measured energy level and estimated energy level from the model for one week of operation are shown in Figure 3.16. In the deployment two different DLC capacities have been implemented,

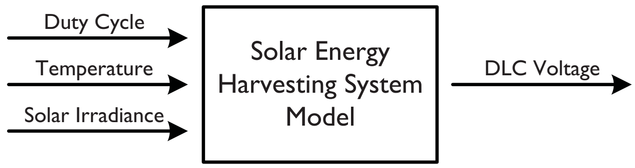
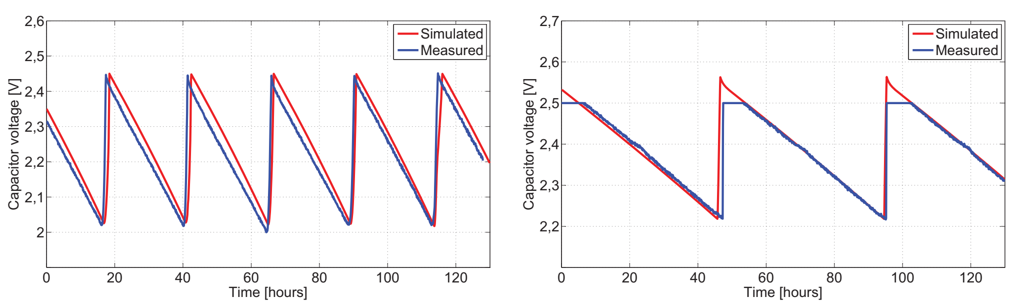


Figure 3.15: Overview of the solar energy harvesting system model on high abstraction level

leading to results from a 10 F version, shown in Figure 3.16a and results from a 22 F version in Figure 3.16b. While for both cases the evaluation shows that the simulations and measurements generally conform well, smaller error sources can be detected. This is visible especially in Figure 3.16a, where some time-shifts can be observed. It was shown that, this is the result of slightly varying hysteresis bounds. While the model can be adjusted to average hysteresis bound levels, variations in the trigger level related to environmental conditions are very difficult to predict. In Figure 3.16b a difference at the sawtooth tip is obvious. However, this does not result from a modeling error of the EH system, but from a measurement error, due to a limited measurement range of up to 2.5 V.

After the model has been evaluated, it can be used for its purpose to sup-



(a) (b)

Figure 3.16: Evaluation of energy levels estimated by the system model compared to deployment results – configured with (a) a 10 F DLC and (b) a 22 F DLC

Parameter	Optimization Goal
Solar panel size	Minimization of cost and size
DLC capacity	Minimization of cost and size
Load consumption	Maximization of duty cycle (sampling rate)
Solar irradiance	Minimization of environmental dependency
Load availability	Maximization of system uptime
Hysteresis band	Minimization of component stress

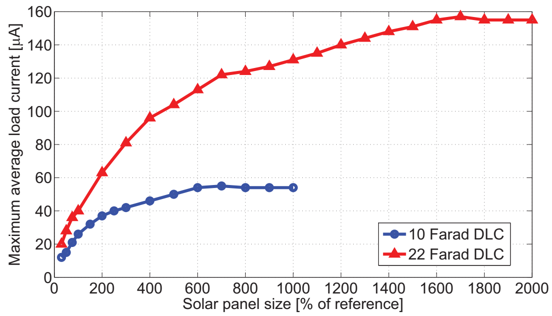
Table 3.5: Dimensioning parameters of the example solar energy harvesting architecture with their respective optimization goals

port the optimization process of parameter dimensioning. Parameters of interest to be optimized for the example architecture are listed in Table 3.5. While typically not all the optimization goals can be met simultaneously, the simulator enables the discovery of trade-offs that meet the given application requirements. A typical simulation outcome for a given system architecture can appear as shown in Figure 3.17, where the simulations have been conducted for the example architecture. These results have been obtained using two simplifications:

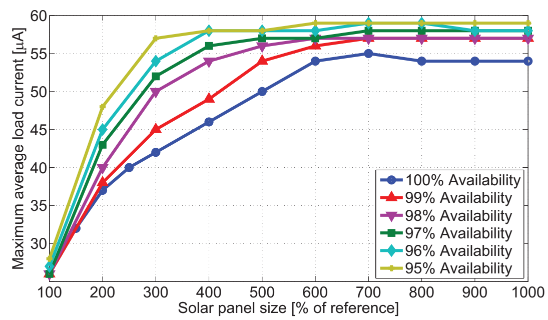
1. Solar panel size is perfectly scalable, meaning that an increase of solar panel size leads to an increase by the same factor in irradiance-to-current conversion at the same voltage.

2. At times of insufficient energy (i.e., voltage level of the DLC is lower than minimal boost converter input) the load will consume no energy.

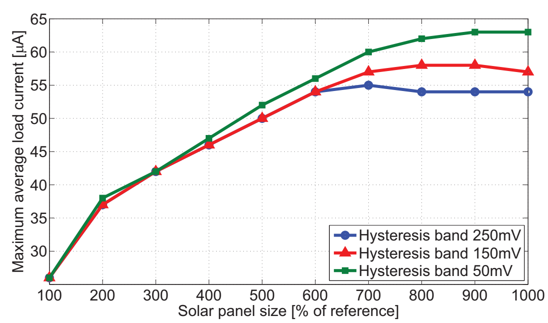
Furthermore, all the simulations are based on the irradiance data of December 2009 in Sundsvall, Sweden, representing the darkest month in the year. The results show that the maximum allowable load current (as average value) under varying system configurations. All sub-figures include variations of solar panel size as a typical parameter which is to be changed. Additionally, Figure 3.17a shows the impact of change in DLC capacity, Figure 3.17b the change of availability requirements and Figure 3.17c the variation of the hysteresis band setting of the over-voltage protection circuit. These types of



(a)



(b)



(c)

Figure 3.17: Simulation sweeps of different dimensioning parameters with their influence on maximum allowed load current consumption – (a) DLC sweep at 100 % availability, 350 mV hysteresis band; (b) Availability sweep at 10 F DLC capacity, 350 mV hysteresis band; (c) Hysteresis sweep at 100 % availability, 10 F DLC capacity

results can then be used to choose the appropriate configuration parameters in order to fulfill desired system operation.

In particular, these results can support decisions for system designers. Taking Figure 3.17b as an example, it can be extracted that it is not reasonable to increase the solar panel size by more than a factor of three, when lifetime is the constraint with an availability factor of 95 %. At the same time, trade-off decisions can be made, such as determining, that a 45 µA average load current constraint can be reached with a 96 % availability and double solar panel size, or with a 99 % availability, combined with a triple-sized solar panel.

4 Conclusions

4.1 Thesis Summary

In this thesis, mechanisms for extending the lifetime of Wireless Sensor Networks for Environmental Monitoring have been presented, allowing them to operate as autonomous measurement systems for long periods of time. After defining the application background and scope in chapter 1, an extension of system lifetime has been addressed on two levels. On the first level, the energy consumption of the system has been addressed and methods to reduce the energy consumption have been identified. This includes the consumption of resources provided, as well as the energy efficient usage of these resources. Existing means to reduce average energy consumption, such as module duty-cycling, have been presented. It was then demonstrated that the node synchronization could be used as an optimization tool.

On the second level, the energy supply has been targeted to allow for a long-term operation. Limitations of batteries as traditional power sources have been disclosed and alternative battery-less energy supply architectures have been presented. Architectures, optimized for low irradiance conditions (i.e., optimized for low energy overhead) have been demonstrated in real-world conditions. Moreover, the dimensioning of energy harvesting systems has been addressed with the assistance of energy-level simulations. The capability involved using these simulations to support an optimal system configuration has been presented, using previously mentioned system examples.

4.2 Discussion of Contributions

4.2.1 Synchronized Duty Cycling

Duty cycling is a very efficient approach to conserving energy for modules that have bimodal energy consumption states. While for many node-internal modules duty cycling can be used in a straight-forward manner, communication shows some limitations, as it has external interactions. Nevertheless, particularly in relation to RF units, it is desirable for there to be duty cycling, because it shows a typical bimodal behavior with energy consumption differences of magnitudes between active and inactive states.

Strict scheduling of communication enable active times of the module to be minimized, eliminating energy waste through unnecessary operations, such as idle listening, over-hearing or retransmissions. To implement schedules, the same notion of time between communication partners is necessary, requiring time synchronization due to error-prone clock sources.

As energy savings are the main reason for implementing a synchronization, the energy cost of the synchronization algorithm itself is of significant concern. By detecting a single synchronization message per communication frame and measuring the interval between two consecutive messages, the synchronization receiver can compare the measured and ideal period. Hence the node is capable of compensating for clock offset and rate with the assistance of a single message per frame.

Nevertheless, this does not allow for high-precision time synchronization, as is the case for other such protocols. Although it is possible for guard times to be reduced to values of the order of 100 μs unrelated to the sampling period, at the same time a significant reduction is produced with regards to necessary message exchanges. An additional temperature compensation is possible to deal with clock drift, which occurs in the measurement period. In comparison to providing extended guard times, the compensation mechanism allows for an energy consumption reduction by approximately 50 %.

4.2.2 Battery-less Solar Energy Harvesting Architectures

Targeting near-perpetual lifetimes in low-power WSNs, battery-based power supply units are often the bottleneck and thus limiting the system lifetime. Even when replenished by ambient energy sources, battery lifetime is strictly limited by the chemical reaction occurring at the electrodes.

Using battery-less energy harvesting systems, implementing e.g. Double-Layer Capacitors as the energy buffer, enables lifetimes far longer than those for battery-based systems. However, due to lower energy densities and higher self-discharge rates compared to batteries, the energy consumption overhead from the energy management in DLC-based systems has a comparably high influence on storage discharge.

Two architectures have been presented, targeting the reduction in energy consumption overhead so as to make solar energy harvesting use feasible in locations with challenging irradiance conditions. This has been accomplished by strapping down systems to their minimal configuration necessary to provide for reliable usage. While these systems cannot compete in energy harvested under strong irradiance conditions, it has been experimentally shown that it allows an uninterrupted supply of low-power sensor nodes, even under low irradiance conditions.

4.2.3 Simulation-based Dimensioning

Using ambient energy sources for WSNs, there is a significant relationship between power supply and the deployment location. While discharge conditions are typically constant, charge conditions depend on environmental parameters (i.e., mainly irradiance level) and the internal parameters of the system. Hence, the available energy levels also depend on these parameters. Due to variations in environmental parameters and the complexity of the charging conditions, it is not obvious how it is possible to obtain a relationship between the different system constraints.

Simulating available energy levels for any point in time, enables it to be determined whether there are sufficient energy levels for system operation at all times under defined component configuration. The results of these simulations can be used to make decisions with regards to component dimensioning in order to meet certain application and environment constraints. Furthermore, it enables it to be possible to support trade-off decisions, such as reducing system availability or sampling rates in order to save cost.

Model implementation of a exemplary architecture has been demonstrated and was evaluated against experimental deployment results. It has been shown, that based on environmental conditions, the input and simulation of energy levels, the relationship between external and internal system constraints can be estimated which thus allows for the prediction of system operation and therefore system dimensioning.

4.3 Overall Conclusions

System lifetime is one of the main limiting parameters for the autonomous operation of measurement systems. Although Wireless Sensor Networks offer tremendous opportunities in relation to enhancing measurement systems so as to make them distributed and autonomous, most existing systems have a very restricted lifetime. These lifetime restrictions are to a great extent given by node level lifetime restrictions due to their energy management, including types of energy sources and their usage.

Contributions presented in this work show, that a combination of efficient resource use and inexhaustible energy supply can support the extension of a system lifetime.The main focus has been in relation to a reduction of energy overhead, which in turn increases the energy efficiency of the system. In addition, the supply of these reduced energy demands should originate from durable energy sources. Experimental results have been presented, showing that such sources, despite their limitations, can be used to provide an uninterrupted, near-perpetual power supply for given application scenarios.

4.4 Future Work

Future work in WSN energy management should include further investigation into node platforms, the balancing of unequal energy distributions and long-term behavioral studies of systems in real-world deployments. For node platforms, it might be of particular interest to investigate hybrid architectures as in these, high performance data processing is outsourced to the sensor, while the system and communication control is handled centrally. Additionally, the problem of unequally distributed energy availability should be addressed. This includes spatial, as well as temporal distribution of energy availability. Adaptive sampling algorithms have been previously presented in the literature although it does appear that a further study of system improvement capabilities and the integration into network structures is required. Finally, long-term studies of systems at the deployment stage might provide data, relating environmental conditions to system behavior. Energy-efficient self-monitoring mechanisms are necessary in order to allow these studies without unnecessarily influencing the system lifetime.

Bibliography

[1] M. Weiser, "The computer for the 21st century," *Scientific American*, vol. 265, pp. 66–75, January 1991.

[2] G. D. Abowd and E. D. Mynatt, "Charting past, present, and future research in ubiquitous computing," *ACM Trans. Comput.-Hum. Interact.*, vol. 7, no. 1, pp. 29–58, 2000.

[3] B. Warneke, M. Last, B. Liebowitz, and K. S. J. Pister, "Smart dust: Communicating with a cubic-millimeter computer," *Computer*, vol. 34, no. 1, pp. 44–51, 2001.

[4] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Comput. Netw.*, vol. 52, no. 12, pp. 2292–2330, 2008.

[5] J. Artiola, I. Pepper, and M. Brusseau, *Environmental Monitoring and Characterization*. Elsevier Academic Press, 2004.

[6] J. Porter, P. Arzberger, H.-W. Braun, P. Bryant, S. Gage, T. Hansen, P. Hanson, C.-C. Lin, F.-P. Lin, T. Kratz, W. Michener, S. Shapiro, and T. Williams, "Wireless Sensor Networks for Ecology," *BioScience*, vol. 55, no. 7, pp. 561–572, 2005.

[7] G. Barrenetxea, F. Ingelrest, G. Schaefer, M. Vetterli, O. Couach, and M. Parlange, "Sensorscope: Out-of-the-box environmental monitoring," *Proceedings of the 7th international conference on Information processing in sensor networks*, pp. 332–343, 2008.

[8] T. Wark, W. Hu, P. Corke, J. Hodge, A. Keto, B. Mackey, G. Foley, P. Sikka, and M. Brunig, "Springbrook: Challenges in developing a long-term, rainforest wireless sensor network," *2008 International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, pp. 599–604, 2008.

[9] F. Ingelrest, G. Barrenetxea, G. Schaefer, M. Vetterli, O. Couach, and M. Parlange, "SensorScope: Application-Specific Sensor Network for Environmental Monitoring," *ACM Transactions on Sensor Networks*, vol. 6, pp. 1–32, Feb. 2010.

[10] D. M. Doolin, "Wireless sensors for wildfire monitoring," in *Proceedings of SPIE*, vol. 5765, pp. 477–484, SPIE, 2005.

[11] G. Werner-Allen, K. Lorincz, M. Ruiz, O. Marcillo, J. Johnson, J. Lees, and M. Welsh, "Deploying a wireless sensor network on an active volcano," *IEEE Internet Computing*, vol. 10, pp. 18–25, Mar. 2006.

[12] T. Le Dinh, W. Hu, P. Sikka, P. Corke, L. Overs, and S. Brosnan, "Design and deployment of a remote robust sensor network: Experiences from an outdoor water quality monitoring network," in *32nd IEEE Conference on Local Computer Networks, 2007. LCN 2007*, pp. 799–806, 2007.

[13] K. Martinez, R. Ong, and J. Hart, "Glacsweb: a sensor network for hostile environments," *2004 First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks, 2004. IEEE SECON 2004.*, pp. 81–87, 2004.

[14] L. Selavo, A. Wood, Q. Cao, T. Sookoor, H. Liu, A. Srinivasan, Y. Wu, W. Kang, J. Stankovic, D. Young, and Others, "Luster: wireless sensor network for environmental research," in *Proceedings of the 5th international conference on Embedded networked sensor systems*, pp. 103–116, 2007.

[15] R. Szewczyk, A. Mainwaring, J. Polastre, J. Anderson, and D. Culler, "An analysis of a large scale habitat monitoring application," in *Proceedings of the 2nd international conference on Embedded networked sensor systems*, pp. 214–226, 2004.

[16] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, "Wireless sensor networks for habitat monitoring," *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pp. 88–97, 2002.

[17] K. Martinez, P. Padhy, A. Elsaify, G. Zou, A. Riddoch, and JK, "Deploying a sensor network in an extreme environment," *Sensor Networks, Ubiquitous and Trustworthy Computing, June*, 2006.

[18] K. Martinez, J. K. Hart, and R. Ong, "Deploying a wireless sensor network in iceland," in *GSN '09: Proceedings of the 3rd International Conference on GeoSensor Networks*, pp. 131–137, 2009.

[19] G. Werner-Allen, J. Johnson, M. Ruiz, J. Lees, and M. Welsh, "Monitoring volcanic eruptions with a wireless sensor network," in *Proceeedings of the Second European Workshop on Wireless Sensor Networks*, pp. 108–120, 2005.

[20] H. Liu, L. Selavo, and J. Stankovic, "SeeDTV: Deployment-time validation for wireless sensor networks," in *Proceedings of the 4th workshop on Embedded networked sensors*, pp. 23–27, 2007.

[21] G. Tolle, D. Gay, W. Hong, J. Polastre, R. Szewczyk, D. Culler, N. Turner, K. Tu, S. Burgess, T. Dawson, and P. Buonadonna, "A macroscope in the redwoods," in *Proceedings of the 3rd international conference on Embedded networked sensor systems - SenSys '05*, p. 51, ACM Press, 2005.

[22] I. Dietrich and F. Dressler, "On the lifetime of wireless sensor networks," *ACM Transactions on Sensor Networks*, vol. 5, pp. 1–39, Feb. 2009.

[23] J. Hill and D. Culler, "Mica: A wireless platform for deeply embedded networks," *IEEE Micro*, vol. 22, no. 6, pp. 12–24, 2002.

[24] P. Sikka, P. Corke, L. Overs, P. Valencia, and T. Wark, "Fleck - A platform for real-world outdoor sensor networks," in *2007 3rd International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, pp. 709–714, 2007.

[25] J. Polastre, R. Szewczyk, and D. Culler, "Telos: enabling ultra-low power wireless research," in *IPSN 2005. Fourth International Symposium on Information Processing in Sensor Networks, 2005.*, pp. 364–369, 2004.

[26] H. Dubois-Ferriere, R. Meier, L. Fabre, and P. Metrailler, "TinyNode: a comprehensive platform for wireless sensor network applications," in

2006 5th International Conference on Information Processing in Sensor Networks, pp. 358–365, 2006.

[27] P. Dutta, J. Taneja, J. Jeong, X. Jiang, and D, “A building block approach to sensornet systems,” in *Proceedings of the 6th ACM conference on Embedded network sensor systems*, pp. 267–280, 2008.

[28] C. Lynch and F. O’Reilly, “PIC-based sensor operating system,” in *Irish Signals and Systems Conference*, pp. 95–100, 2004.

[29] I. Hakala, M. Tikkakoski, and I. Kivelä, “Wireless Sensor Network in Environmental Monitoring - Case Foxhouse,” in *2008 Second International Conference on Sensor Technologies and Applications (sensorcomm 2008)*, pp. 202–208, IEEE, 2008.

[30] “Sentio series of sensor nodes – website.” `http://www.miun.se/stc/Sentio`. (accessed 08-10-2010).

[31] E. Serpedin and Q. M. Chaudhari, *Synchronization in Wireless Sensor Networks: Parameter Estimation, Peformance Benchmarks, and Protocols*. New York, NY, USA: Cambridge University Press, 2009.

[32] B. Sundararaman, U. Buy, and A. D. Kshemkalyani, “Clock synchronization for wireless sensor networks: A survey,” *Ad Hoc Networks*, vol. 3, pp. 281–323, 2005.

[33] Y. R. Faizulkhakov, “Time synchronization methods for wireless sensor networks: A survey,” *Programming and Computer Software*, vol. 33, no. 4, pp. 214–226, 2007.

[34] S. Ganeriwal, R. Kumar, and M. B. Srivastava, “Timing-sync protocol for sensor networks,” in *SenSys ’03: Proceedings of the 1st international conference on Embedded networked sensor systems*, pp. 138–149, ACM, 2003.

[35] J. Elson, L. Girod, and D. Estrin, “Fine-grained network time synchronization using reference broadcasts,” *SIGOPS Oper. Syst. Rev.*, vol. 36, no. SI, pp. 147–163, 2002.

[36] M. Maróti, B. Kusy, G. Simon, and A. Lédeczi, "The flooding time synchronization protocol," in *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*, pp. 39–49, ACM, 2004.

[37] J. van Greunen and J. Rabaey, "Lightweight time synchronization for sensor networks," in *WSNA '03: Proceedings of the 2nd ACM international conference on Wireless sensor networks and applications*, pp. 11–19, ACM, 2003.

[38] K.-L. Noh and E. Serpedin, "Pairwise broadcast clock synchronization for wireless sensor networks," in *World of Wireless, Mobile and Multimedia Networks, 2007. WoWMoM 2007. IEEE International Symposium on a*, pp. 1 –6, jun. 2007.

[39] W. Su and I. F. Akyildiz, "Time-diffusion synchronization protocol for wireless sensor networks," *IEEE/ACM Trans. Netw.*, vol. 13, no. 2, pp. 384–397, 2005.

[40] Tech Tools, *Digiview DV1-100*.

[41] Epson-Toyocom, *kHz Range Crystal Unit FC-135*.

[42] Texas Instruments, *Low-power Digital Temperature Sensor TMP102 (SBOS397A)*, 2007.

[43] Intersil, *Low-power RTC ISL12022M (FN6668.7)*, 2010.

[44] D. Linden and T. Reddy, *Handbook of Batteries*. McGraw-Hill Professional, 2001.

[45] J. Paradiso and T. Starner, "Energy Scavenging for Mobile and Wireless Electronics," *IEEE Pervasive Computing*, vol. 4, pp. 18–27, Jan. 2005.

[46] "U.s. energy information administration – website." `http://www.eia.doe.gov`. (accessed 02-08-2010).

[47] S. Sudevalayam and P. Kulkarni, "Energy Harvesting Sensor Nodes: Survey and Implications," 2008. Tech. Report.

[48] A. Kansal, J. Hsu, S. Zahedi, and M. B. Srivastava, "Power management in energy harvesting sensor networks," *ACM Trans. Embed. Comput. Syst.*, vol. 6, no. 4, p. 32, 2007.

[49] "Source of figure 3.4." `http://commons.wikimedia.org/wiki/File:Supercapacitors_chart.svg`. (accessed 23-09-2010).

[50] C. Park and P. Chou, "AmbiMax: Autonomous Energy Harvesting Platform for Multi-Supply Wireless Sensor Nodes," in *3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks*, pp. 168–177, IEEE, 2006.

[51] D. Krüger, C. Buschmann, and S. Fischer, "Solar powered sensor network design and experimentation," in *ISWCS'09: Proceedings of the 6th international conference on Symposium on Wireless Communication Systems*, pp. 11–15, IEEE Press, 2009.

[52] V. Raghunathan, A. Kansal, J. Hsu, J. Friedman, and M. Srivastava, "Design considerations for solar energy harvesting wireless embedded systems," in *IPSN '05: Proceedings of the 4th international symposium on Information processing in sensor networks*, pp. 457–462, IEEE Press, 2005.

[53] P. Corke, P. Valencia, P. Sikka, T. Wark, and L. Overs, "Long-duration solar-powered wireless sensor networks," in *EmNets '07: Proceedings of the 4th workshop on Embedded networked sensors*, pp. 33–37, ACM, 2007.

[54] C. Alippi and C. Galperti, "An adaptive system for optimal solar energy harvesting in wireless sensor network nodes," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 55, pp. 1742 –1750, jul. 2008.

[55] F. Simjee and P. H. Chou, "Everlast: long-life, supercapacitor-operated wireless sensor node," in *ISLPED '06: Proceedings of the 2006 international symposium on Low power electronics and design*, pp. 197–202, ACM, 2006.

[56] X. Jiang, J. Polastre, and D. Culler, "Perpetual environmentally powered sensor networks," in *IPSN '05: Proceedings of the 4th international*

symposium on Information processing in sensor networks, pp. 463 – 468, IEEE Press, 2005.

[57] J. Alberola, J. Pelegri, R. Lajara, and J. Perez, "Solar inexhaustible power source for wireless sensor node," in *Instrumentation and Measurement Technology Conference Proceedings, 2008. IMTC 2008. IEEE*, pp. 657 –662, 2008.

[58] I. Buchmann, "Characteristics of commonly used rechargeable batteries." `http://www.batteryuniversity.com/partone-3.htm`. (accessed 24-09-2010).

[59] W. Teulings, J. Marpinard, A. Capel, and D. O'Sullivan, "A new maximum power point tracking system," in *Proceedings of IEEE Power Electronics Specialist Conference - PESC '93*, pp. 833–838, IEEE, 1993.

[60] O. Wasynezuk, "Dynamic Behavior of a Class of Photovoltaic Power Systems," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-102, pp. 3031–3037, Sept. 1983.

[61] M. Masoum, H. Dehbonei, and E. Fuchs, "Theoretical and experimental analyses of photovoltaic systems with voltageand current-based maximum power-point tracking," *IEEE Transactions on Energy Conversion*, vol. 17, pp. 514–522, Dec. 2002.

[62] V. Kyriatzis, N. S. Samaras, P. Stavroulakis, H. Takruri-Rizk, and S. Tzortzios, "Enviromote: A New Solar-Harvesting Platform Prototype for Wireless Sensor Networks / Work-in-Progress Report," in *2007 IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 1–5, IEEE, 2007.

[63] Cooper Bussmann, *PowerStor Application Guidelines (PS-5006)*, 2007.

[64] Texas Instruments, *TPS6107x datasheet (SLVS510C)*, 2009.

[65] A. Aberle, S. Wenham, and M. Green, "A new method for accurate measurements of the lumped series resistance of solar cells," in *Conference Record of the Twenty Third IEEE Photovoltaic Specialists Conference*, pp. 133–139, 1993.

[66] R. Faranda, M. Gallina, and D. T. Son, "A new simplified model of double-layer capacitors," in *International Conference on Clean Electrical Power*, pp. 706–710, 2007.