Thesis for the degree of Doctor of Technology
Sundsvall 2014

Energy Efficient Wireless Sensor Node Architecture for Data and Computation Intensive Applications

Khurram Shahzad

Supervisor: Professor Bengt Oelmann

Faculty of Science, Technology and Media,
Mid Sweden University, SE-851 70 Sundsvall, Sweden

ISSN 1652-893X
Mid Sweden University Doctoral Thesis 192

ISBN 978-91-87557-64-4

Mittuniversitetet
MID SWEDEN UNIVERSITY

Akademisk avhandling som med tillstånd av Mittuniversitetet i Sundsvall framläggs till offentlig granskning för avläggande av teknologie doktors examen i elektronik onsdagen den 11 Juni 2014, klockan 10:15 i sal M102, Mittuniversitetet Sundsvall. Seminariet kommer att hållas på engelska.

Energy Efficient Wireless Sensor Node Architecture for Data and Computation Intensive Applications

Khurram Shahzad

Faculty of Science, Technology and Media
Mid Sweden University, SE-851 70 Sundsvall
Sweden

Telephone: +46 (0)60 148755

Printed by Kopieringen Mittuniversitetet, Sundsvall, Sweden, 2014

Abstract

Wireless Sensor Networks (WSNs), in addition to enabling monitoring solutions for numerous new applications areas, have gained huge popularity as a cost-effective, dynamically scalable, easy to deploy and maintainable alternatives to conventional infrastructure-based monitoring solutions.

A WSN consists of spatially distributed autonomous wireless sensor nodes that measure desired physical phenomena and operate in a collaborative manner to relay the acquired information wirelessly to a central location. A wireless sensor node, integrating the required resources to enable infrastructure-less distributed monitoring, is constrained by its size, cost and energy. In order to address these constraints, a typical wireless sensor node is designed based on low-power and low-cost modules that in turn provide limited communication and processing performances. Data and computation intensive wireless monitoring applications, on the other hand, not only demand higher communication bandwidth and computational performance but also require practically feasible operational lifetimes so as to reduce the maintenance cost associated with the replacement of batteries. In relation to the communication and processing requirements of such applications and the constraints associated with a typical wireless sensor node, this thesis explores energy efficient wireless sensor node architecture that enables realization of data and computation intensive applications.

Architectures enabling raw data transmission and in-sensor processing with various technological alternatives are explored. The potential architectural alternatives are evaluated both analytically and quantitatively with regards to different design parameters, in particular, the performance and the energy consumption. For quantitative evaluation purposes, the experiments are conducted on vibration and image-based industrial condition monitoring applications that are not only data and computation intensive but also are of practical importance.

Regarding the choice of an appropriate wireless technology in an architecture enabling raw data transmission, standard based communication technologies including infrared, mobile broadband, WiMax, LAN, Bluetooth, and ZigBee are investigated. With regards to in-sensor processing, different architectures comprising of sequential processors and FPGAs are realized to evaluate different design parameters, especially the performance and energy efficiency. Afterwards, the architectures enabling raw data transmission only and those involving in-sensor processing are evaluated so as to find an energy efficient solution. The results of this investigation show that in-sensor processing architecture, comprising of an FPGA for computation purposes, is more energy efficient when compared with other alternatives in relation to the data and computation intensive applications.

Based on the results obtained and the experiences learned in the architectural evaluation study, an FPGA-based high-performance wireless sensor

platform, the SENTIOF, is designed and developed. In addition to performance, the SETNIOF is designed to enable dynamic optimization of energy consumption. This includes enabling integrated modules to be completely switched-off and providing a fast configuration support to the FPGA.

In order to validate the results of the evaluation studies, and to assess the performance and energy consumption of real implementations, both the vibration and image-based industrial monitoring applications are realized using the SENTIOF. In terms of computational performance for both of these applications, the real-time processing goals are achieved. For example, in the case of vibration-based monitoring, real-time processing performance for tri-axes (horizontal, vertical and axial) vibration data are achieved for sampling rates of more than 100 kHz.

With regards to energy consumption, based on the measured power consumption that also includes the power consumed during the FPGA's configuration process, the operational lifetimes are estimated using a single cell battery (similar to an AA battery in terms of shape and size) with a typical capacity of 2600 mA. In the case of vibration-based condition monitoring, an operational lifetime of more than two years can be achieved for duty-cycle interval of 10 minutes or more. The achievable operational lifetime of image-based monitoring is more than 3 years for a duty-cycle interval of 5 minutes or more.

Acknowledgements

First and foremost, I thank Almighty Allah for giving me the strength and courage to complete this work.

I would like to thank Prof. Bengt Oelmann for giving me the opportunity to pursue my PhD studies at Mid Sweden University and providing guidance during this research work.

Peng Cheng is acknowledged for his assistance in the development of the hardware and software.

Mattias O′Nills, Claes Mattsson, Carolina Blomberg, Fanny Burman are gratefully acknowledged for providing administrative assistance during the study period.

Najeem Lawal is acknowledged for providing his assistance in debugging VHDL codes.

Mikael Bylund is acknowledged for his help in "context aware" translation of letters etc. and in learning Swedish culture and language. It was because of him that I have been fluently speaking Swedish, "Jag prata inte svenska!"

I would like to thank Jawad Saleem, Abdul Majid, Abdul Waheed Malik, Muhammad Imran, Mazhar Hussain, Muhammad Amir Yousaf, Naeem Ahmed, Khursheed, Muhammad Nazar ul Islam, and Shakeel Ashraf for their help both at the school and at off-school. Muhammad Imran is also acknowledged for providing valuable reviews on this thesis.

I would also like to thank all other colleagues at the department of Electronics Design who have directly or indirectly contributed to this work.

Higher Education Commission (HEC) of Pakistan, and Sensible Things that Communicate (STC), a research program of Mid Sweden University, are gratefully acknowledged for their financial support during this work.

I would like to thank my family and friends for their constant inspirations, motivations and well wishes that helped me in achieving this stage. In this regards, I would like to mention the guidance and financial help of my uncle Muneer Ahmed Abbasi which played an important role in paving a way to pursue my higher education studies.

Finally, I would like to thank my wife Sidra for her unconditional and untiring support during all the good and bad days. This acknowledgement would be incomplete without mentioning my daughter Eshaal, who's almost three years of journey in this world, has brought so much happiness in my life.

Sundsvall, April 2014

Khurram Shahzad

Abbreviations and Acronyms

ADC	.....	Analog to Digital Converter
API	.....	Application Programming Interface
ASIC	.....	Application Specific Integrated Circuit
BLE	.....	Bluetooth Low Energy
BRAM	.....	Block RAM
CLB	.....	Configurable Logic Block
CNC	.....	Computer Numerically Controller
CPLD	.....	Complex Programmable Logic Device
CPU	.....	Central Processing Unit
DCM	.....	Digital Clock Manager
DMA	.....	Direct Memory Access
DRAM	.....	Dynamic RAM
DSN	.....	Distributed Sensor Network
DSP	.....	Digital Signal Processing/Processor
FFT	.....	Fast Fourier Transform
FIR	.....	Finite Inpulse Response
FPGA	.....	Field Programmble Gate Array
GPP	.....	General Purpose Processor
GPS	.....	Global Positioning System
GPU	.....	Graphic Processing Unit
HDL	.....	Hardware Description Language
IOB	.....	Input/Output Block
IP	.....	Intellectual Property
LUT	.....	Look Up Table
MAC	.....	Multiply and Accumulate
MOS	.....	Metal Oxide Semiconductor
PLD	.....	Programmble Logic Device
PLL	.....	Phase Lock Loop
QoS	.....	Quality of Service
RAM	.....	Random Access Memory
RISC	.....	Reduced Instruction Set Computer
RTC	.....	Real Time Counter
RTL	.....	Register Transfer Level
SNR	.....	Signal-to-Noise Ratio
SoC	.....	System-on-Chip
SPI	.....	Serial Peripheral Interface
SRAM	.....	Static RAM
		Very High Speed Integrated Circuit (VHSIC)
VHDL	.....	Hardware Description Language
WSN	.....	Wireless Sensor Network

Table of Contents

List of Figures

List of Tables

List of Papers

This thesis is mainly based on the following eight papers, herein referred to by their Roman numerals. The papers are also appended at the end of the thesis.

Paper I **Feasibility Study of On-Rotor Vibration Monitoring using Accelerometers**
Khurram Shahzad, Peng Cheng, and Bengt Oelmann
In draft.

Paper II **Architecture Exploration for a High-performance and Low-power Wireless Vibration Analyzer**
Khurram Shahzad, Peng Cheng, and Bengt Oelmann
IEEE Journal on Sensors, vol.13, no.2, pp.670-682, Feb. 2013

Paper III **A comparative study of in-sensor processing vs. raw data transmission using ZigBee, BLE and Wi-Fi for data intensive monitoring applications**
Khurram Shahzad and Bengt Oelmann
Submitted in the ACM conference SenSys 2014

Paper IV **Investigating Energy Consumption of an SRAM-based FPGA for Duty-Cycle Applications**
Khurram Shahzad and Bengt Oelmann
Conference on Advances in Parallel Computing, ParCo2013, vol. 25, pp. *548-559,* 10-13 Sept. 2013.

Paper V **SENTIOF An FPGA-based High-Performance Wireless Embedded Platform**
Khurram Shahzad, Peng Cheng, and Bengt Oelmann
Federated Conference on Computer Science and Information Systems (FedCSIS) 2013, pp.901-906, 8-11 Sept. 2013

Paper VI **Quantitative Evaluation of an FPGA based Wireless Vibration Monitoring System**
Khurram Shahzad and Bengt Oelmann
Accepted for publication in International Conference on Circuits and System, 2014.

Paper VII **An FPGA-Based High-Performance Wireless Vibration Analyzer**
Khurram Shahzad and Bengt Oelmann
IEEE Conference NORCHIP 2013, pp. 1-5, 11-12 Nov. 2013

Paper VIII **Energy Efficient FPGA based Wireless Vision Sensor Node: SENTIOF-CAM**
Muhammad Imran, Khurram Shahzad, Naeem Ahmad, Mattias O'Nils, Najeem Lawal and Bengt Oelmann.
Submitted to IEEE Transactions on Circuits and Systems for Video Technology, 2013.

1

Introduction & Problem Formulation

The transition from the 20th to 21st century observed the emergence of low-cost, low-power, and miniature size electronics, enabling attractive solutions for numerous new application areas to be created as well as facilitating several existing ones to improved. One such example is the development of Wireless Sensor Network (WSN), providing a low-cost alternative to both manual monitoring solutions and traditional infrastructure-based monitoring solutions, in which both the power and the data are required to be transported over a physical media, such as cables. In contrast to an infrastructure-based monitoring network, a WSN is comprised of spatially distributed sensor nodes that, in addition to sensing the environment, are capable of communicating wirelessly to transport the acquired data to a desired destination. In addition, the wireless communication in a WSN also provides the means to establish a self-organizing wireless monitoring network. A WSN finds its applications in numerous fields spanning from home automation to industrial monitoring, and to battlefield tracking etc. [1]-[7]. Some of the notable applications include environmental monitoring [8]-[10], fire detection in forests [11]-[13], structural health monitoring for buildings and bridges [14]-[15], health care monitoring [16]-[17], industrial condition monitoring [18]-[19], battlefield monitoring [20]-[22], precision agriculture [23] and logistics monitoring [24].

In comparison to infrastructure-based monitoring networks, the advantages associated with WSNs are low-cost, ability to self-organize, scalability and ease of deployment [25]-[28].

- As the sensor nodes in a WSN communicate wirelessly and, the means of energy (for example, batteries) are integrated within them, the cost associated with the development and maintenance of communication and power related infrastructure is low as compared to infrastructure-based networks. In addition, typically, sensor nodes have a simple design that leads to the low-cost development and thus, enables the realizing of cost-effective WSNs.

- The ability of wireless sensor nodes to self-organize and establish a network plays an important role in achieving highly scalable monitoring networks. In contrast to infrastructure-based networks, the low-complexity and the associated cost of adding new sensor nodes in a WSN provides the means to achieve a highly scalable solution. The scalability enables applications in which sensor nodes may be added or removed without halting the operation of the network which, in fact, in certain cases this characteristic might be essential in achieving the desired goals. For example, in monitoring and tracking applications consisting of non-stationary sensor nodes, some nodes may collaborate in one network for a specific time or event and then leave that network and join another nearby WSN.
- In relation to physical deployment, wireless sensor nodes provide high-flexibility as compared to infrastructure-based monitoring solutions. For example, in addition to regular deployment in which sensor nodes are placed according to pre-planned fixed locations, wireless sensor nodes can also be deployed in a random manner. Such flexibility could be highly desirable in certain applications, for example, in order to detect and monitor fire in a forest; such nodes can be dropped from an airplane. In addition, infrastructure-less operability of wireless sensor networks enables the sensor nodes to be deployed in hard-to-reach as well as non-stationary locations.

In a wireless monitoring application that may monitor the desired parameters at a few locations over a small geographical area or at hundred of locations over a large area, it is a wireless sensor node that plays a central role in realizing monitoring solutions based on WSNs. A typical wireless sensor node is a compact sized hardware unit that acquires desired data from its environment and communicates wirelessly to other nodes in a network so as to relay that data or extracted information to a central station. The architecture and data flow of a typical wireless sensor node [29] is shown in **Figure 1.1**. Depending upon what and how many parameters are to be monitored in an application, it may consist of one or more transducers that measure physical phenomena and produce equivalent electrical outputs, that is, in the form of electrical current or voltage. The output signal from a transducer, which is typically low in amplitude, is amplified with the assistance of signal conditioning circuits so as to ensure that it matches the requirements of the digitization circuits. Following on from the amplification

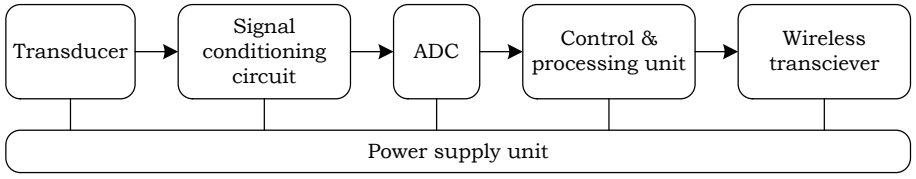

Figure 1.1 Architecture of a typical wireless sensor node

process, a signal is then digitized using an analog-to-digital converter (ADC). In the case of a transducer that produces digital output, along with basic sensing system, both the amplification and the digitization circuits are embedded into the transducer. The digitized signal is then fed to a control and processing unit, typically a micro-controller, which with little or no processing at all, relays the acquired data to its destination using a wireless transceiver. With regards to modules integrated in a wireless sensor node, the control and processing unit plays an important role, as it not only coordinates all the activities within a sensor node but also manages its association with other nodes in a WSN. In addition to these modules, which enable sensing, processing/control and wireless communication capabilities, a sensor node incorporates an energy source such as a battery or some mechanism for energy scavenging [30]-[32], through which all of the integrated modules are powered. In some applications, a sensor node may also be equipped with actuators that enable mobility in sensor nodes [2],[33]-[34].

A WSN, comprising of distributed sensor nodes that require no physical media to enable communication and energy transfer, when compared with wired-counterparts offers the advantages of reduced infrastructure related costs, improved scalability, and ease-of-deployment [25]-[28]. The infrastructure-less and distributed nature of the operation, however, imposes certain constraints that restrict their widespread adoption in (envisioned) application domains listed in [2]-[3]. The constraints imposed on the basic building block of a WSN, i.e. a wireless sensor node are depicted in **Figure 1.2** and are discussed in the following.

In comparison to a wired-network in which an individual sensing system, deployed on-field, may only comprise of desired sensors, a wireless sensor node is required to integrate all of the required resources, i.e. sensors, communication transceiver, processing unit and energy source(s). A wireless sensor node comprising of all of the above mentioned resources is then constrained by the physical size and

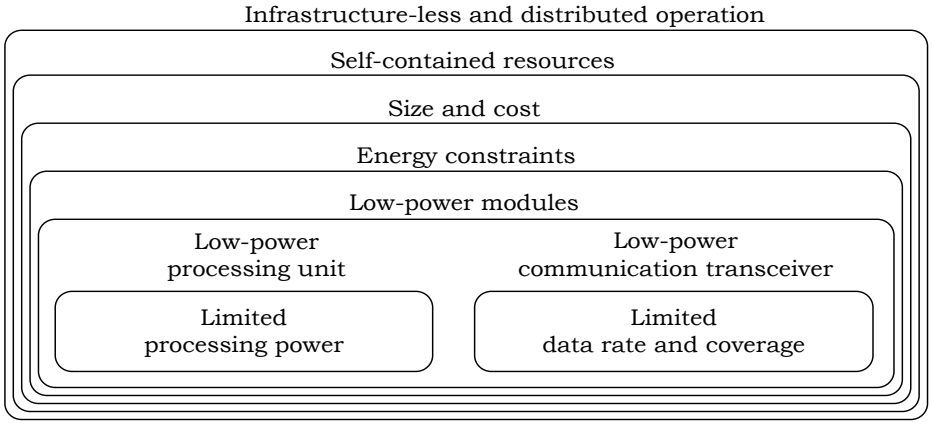

Figure 1.2 The relationship between desired characteristic and resulting constraints of a typical wireless sensor node

the cost. In order to achieve cost-effective high-spatial resolution monitoring through densely deployed sensor nodes, as is one of main objective of WSNs [35]-[37], both the size and the cost of a sensor node is desired to be as low as possible. The small size and low-cost, in turn, set constraints on the size, type and capacity of associated energy source i.e. a battery or an, alternate, energy scavenging unit that can be integrated in a wireless sensor node. Given the limited amount of energy available in typical small size batteries and the low power-to-area (or volume) ratio of alternate energy sources [38]-[41], a small size and low-cost wireless sensor node is restricted to operate on a limited energy budget.

In order to achieve a long operation lifetime with a limited energy budget, a wireless sensor node is typically designed using modules that offer ultra low-power characteristics and enable efficient power management [34],[42]-[43]. A wireless sensor node built on such low-power modules typically lacks high performance capabilities. For example, low-power micro-controllers that are typically used in wireless sensor nodes provide limited computational performance in terms of operating frequency and on-chip memory [44]. Similarly, the low-power and low-cost wireless transceivers provide limited communication data rates. Thus, a WSN, consisting of typical wireless sensor nodes that are constrained by energy, processing, and communication resources [44]-[45], is better suited to applications that acquire data at a low-sample rate and operate in intermittent pattern [27],[46]-[48]. In such applications, a small amount of data, acquired periodically, or on the basis of an event at a sensor node is wirelessly communicated.

As low-sample rate applications require small amounts of data to be communicated, and which can easily be accomplished with the performance provided by low-power processing and communication units, the use of such low-power resources enable energy consumption to be minimized. Using such low-power resources for data intensive monitoring applications, however, not only restrict in achieving the desired monitoring goals but also leads to higher energy consumption. For example, low-power and low-cost transceivers that are used in typical wireless sensor nodes lack the required bandwidth to communicate the large amount of data across the network. Even with intermittent monitoring, a large size memory is required to store the raw data so as to transmit it using a low-data rate wireless transceiver. In that case, the long transmission time causing high energy consumption leads to a short operational lifetime. On the other hand, in an endeavour to suppress the raw data by computing the useful information at each sensor node, the processing performance and amount of memory in low-power micro-controllers that are typically used in wireless sensor nodes is often insufficient to realize applications with the large amount of data that is required to be processed using complex processing algorithms. Improving the communication bandwidth and/or processing capabilities by integrating powerful resources at the sensor node level typically leads to higher power consumption and thus, is required to be assessed in an energy consumption perspective.

With regards to the constraints of a wireless sensor node, as depicted in **Figure 1.2**, this thesis explores an energy efficient architecture that can fulfil the processing and communication requirements of data and computation intensive monitoring applications while enabling a practically feasible operational lifetime.

1.1 Data and Computation Intensive Monitoring Applications

In principle, any monitoring application that requires the monitoring of parameters for which required transducers are available can potentially be realized with a WSN. As a huge collection of transducers [49] do exist that can be incorporated in a wireless sensor node and therefore, the potential applications that could be benefit with the adoption of WSNs is limited only by one's imagination. In this section, an overview of potential application areas for WSN followed by an application classification and design space relating to data and computation intensive wireless monitoring applications is provided.

1.1.1 Overview of applications domains of WSN

The potential applications of WSNs are limitless. Nevertheless, the applicability of WSNs can be briefly summarized in terms of existing real world application fields.

1.1.1.1 Military applications

In fact, the concept of (distributed) sensor network (DSN) can be traced back to the late 1970's, when researchers working on a military project funded by DARPA, identified components associated with a DSN for monitoring applications [50]. The practical utilization was demonstrated by employing a custom-built prototype system, mainly relying on acoustic sensors, for tracking aircrafts flying at low altitude. Despite the successful demonstration, the technology at that time was not considered ready to expand the concept of distributed sensor network both in terms of a large scale and across different fields. However, with the recent advances in technology, numerous military related applications, which in a broader spectrum are those defining the monitoring and tracking of enemy lines as well as the surveillance of one's own territory, are being realized with WSNs. Some of the examples in this domain include WSN-based counter sniper system [51], surveillance [20], and tracking vehicles and troops' movement [21]-[22].

1.1.1.2 Environmental monitoring

With the emergence of low-power, low-cost, and miniature size electronics, the prospective benefits associated with DSNs have attracted both the research and

businesses communities to extend the concept to numerous fields, including environmental monitoring [8]-[11]. In relation to traditional environmental monitoring that relies on highly accurate but on a small number of expensive sensing systems, a WSN enables cost-effective deployment on a large scale [52]. Besides monitoring basic environmental parameters such as temperature, pollution, humidity, etc. to inform inhabitant about their surroundings, other applications in this domain include monitoring of green house gases [53], habitat monitoring [37],[54], forest fire detection [11]-[13], agriculture [23], etc.

1.1.1.3 Home automation

In relation to enabling smart homes, in which appliances are connected to a network so as to manage them intelligently, WSNs are considered an attractive choice [55].

1.1.1.4 Remote health monitoring

In relation to personal health care applications [55], continuous monitoring of patients using WSN [16]-[17], [56]-[58] could be used to detect emergency conditions early on and thus provide the necessary treatment. In addition, such solutions can also be enabled to perform additional activities such as reminding patients regarding the taking of medicines in a timely manner, managing their home appliances, etc. In comparison to keeping a patient in a hospital, a WSN could potentially facilitate the patients in performing their regular activities at home or work while being continuously examined.

1.1.1.5 Business related applications

WSNs can be incorporated in different businesses so as to increase the production, in addition to improving the management and delivery system. Example applications in this regards include monitoring soil and crop related parameters in agriculture business [23][59], monitoring and tracking of livestock in farming [60]-[62], keeping track of both the quality and geographical parameters of logistics [24], industrial monitoring and control, managing inventory related tasks [63] etc.

1.1.1.6 Industrial process management and control

There are number of industrial applications [64]-[67] that can benefit from the use of WSN. Examples include real-time monitoring of operating machinery for possible defects and/or performance degradation, monitoring different processes and their states so as to better control them, enabling factory automation, process control, monitoring of liquid and gas lines for possible leakages, monitoring of

contaminated areas, structural condition monitoring of buildings, mines, bridges etc.

1.1.2 Classification

In order to assess the requirements and challenges associated with designing and deploying WSNs for a diverse nature of applications as mentioned above, it is important to firstly classify these applications in relation to some relevance so that a design space can be deduced for a given set of applications. In [1] and [68], on the basis in which a sensor node interacts with a data sink, the authors classified WSN applications into monitoring, tracking and event detection. Such an abstract level classification does not provide the means to deduce an optimized design space for set of applications being addressed in this thesis. Therefore, the potential applications of WSNs are classified in relation to their (sensed) data size and computational complexity so as to explore an energy efficient architecture for data and computation intensive applications. For a diverse range of monitoring applications, as discussed in section **1.1.1**, a quantitative classification based on data size and computational complexity is a challenging task. Instead, we opt to use a relative scale representing low, medium, high and intensive data size and computational complexity.

1.1.2.1 Application with small data size and low computational complexity

A large number of monitoring applications require static sensor nodes, each of which monitor only few parameters at a relatively low-frequency. In addition, the acquired data, ranging from a couple of bytes to tens of bytes are often transmitted without requiring any complex processing besides simple aggregation (addition/subtraction). Examples of such applications include general environmental monitoring, monitoring of crop & soil in agriculture, monitoring of green house gases, monitoring of different industrial processes, etc.

1.1.2.2 Applications with medium to high data size and computational complexity

Applications such as habitat and logistics monitoring, in which sensor nodes may be carried from one place to other, require a sensor node to continuously discover its surrounding nodes so as to relay the acquired data. In such an application, the size and computational complexity of data may often be low; however, the frequency of acquisition and transmission could be irregular varying the size of the data to be handled at a given time. Apart from acquiring the desired data at a low-sample rate, providing geographical related support, for example,

through Global Position System (GPS) could increase the data size and complexity to process it [10].

In applications such as tracking vehicles and troops, surveillance, structural health monitoring, automation and control etc. the factors that determine the data size and computational complexity include the types and number of sensors used, rate at which the data from each of the sensor is acquired, and level of the information that needs to be extracted from the raw data. Depending upon the objectives and requirements of a given application scenario, the amount of data and complexity involved in processing that data may be solved easily with a typical low-power and low-cost wireless sensor node. In other cases, these may require high-performance processing resources and/or communication transceivers to realize such applications. However, the applications that certainly involve a large amount of acquired data and require intensive processing, as explained in the following section, are dealt within this thesis.

1.1.2.3 Applications with intensive data and computational complexity

Monitoring applications which either acquire scalar data at a high-sample rate (e.g. more than few kHz) or when the data associated with each sample is large (e.g. an image frame), are best candidates for data intensive applications in relation to limited data rate transceivers [69] that are typically used in wireless sensor nodes. In addition, the applications requiring complex processing so as to extract useful information from the large amount of acquired data fall in computational intensive category. Examples of such applications include vibration-based structural and machinery health monitoring, image-based process monitoring, video-based surveillance, etc.

In order to perform quantitative analysis based on actual implementations of data and computation intensive applications, case studies based on applications of highly practical importance, i.e. industrial condition monitoring are conducted in this thesis. In industrial monitoring applications, the most widely accepted methods for analyzing the operating condition of machinery are based on vibration and oil analysis [70]-[73]. In relation to WSNs, vibration-based monitoring, in particular, multi-axes and high-frequency monitoring generate large amounts of data and require intensive signal processing to analyze that data [74]-[79].

In relation to industrial condition monitoring, oil analysis enables the machinery's wear and tear to be assessed by accounting debris/residual particles that detach from the machinery and circulate in the oil [80]-[83]. With camera based wireless sensor nodes, by capturing images of oil passing through a small window such as of glass, debris/residual as well as other foreign particles present in the oil can easily be detected [84]-[86]. In a similar manner to that of the high-frequency vibration-based condition monitoring, image-based oil analysis also

generates a large amount of raw data and involves intensive processing for the analysis of the data [87].

1.1.3 Design space for data and computation intensive applications

The design aspects of a typical WSN as listed in [88] are equally important in developing WSNs for data and computation intensive monitoring applications. However, the scope and emphasis of certain aspects that are of high importance to data and computation intensive applications, especially with regards to industrial condition monitoring, are discussed in the following.

1.1.3.1 Energy

As shown in **Figure 1.2**, the infrastructure less and distributed nature of operation in a wireless sensor node, together with its small size and low-cost sets constraints on the type of energy sources and their capacity that can be integrated in a sensor node. Given the low area/volume to energy ratio, high-cost, and dependency on an appropriate environment for the energy scavenging methods [38]-[41], wireless sensor nodes are typically required to rely on the limited amount energy provided through integrated batteries.

From the energy consumption's perspective of a wireless sensor node, especially with regards to applications mentioned in section **1.1.2.3** the responsible modules are sensor(s), processor and communication transceiver. As the specific requirements of an application may dictate the choice of the sensors to be used thus, leaving not many options to optimize the energy consumption associated with them. However, the energy consumption associated with processing and communication is required to be taken in to consideration. At a wireless sensor node architecture level, this requires choosing appropriate processing algorithms, integrating low-power but energy efficient modules, applying energy conserving techniques, operating nodes in duty-cycle manners, etc [34],[89].

1.1.3.2 Communication modality

Data intensive applications such as those mentioned in section **1.1.2.3** generate a large amount of data, typically in hundreds of kbps to tens of Mbps. The choice of wireless communication technology, such as radio or optical, to communicate such a large amount of data hugely impacts upon both the reliable transfer of data to a destination and the associated energy consumption [90]. The different interferences in the industrial environment can also affect the performance of wireless communication [91]-[92] and thus, are required to be taken in to consideration. In addition, the size of the network, coverage area, and any additional costs associated with obtaining wireless communication services, for

example, licence for the frequency spectrum or other paid services are required to be accounted for.

1.1.3.3 Quality of services (QoS)

In addition to application dependent QoS requirements such as robustness, temper/eavesdropping-resistance and unobtrusiveness, data and computation intensive applications generally tend to have real-time constraints, which could add to existing constraints regarding processing performance and/or communication throughput.

1.1.3.4 Operational lifetime

Mainly derived from the available amount of energy and its consumption for a given application, the operational lifetime highly influence the maintenance cost. With regards to industrial monitoring applications, sensor nodes may be deployed on to continuously operating machinery as well as non-stationary and hard to access locations and would, thus, require the suspending of an on-going activity to replace batteries for sensor nodes. In order to reduce such a high maintenance cost, an operational lifetime of at least several years could be highly desirable.

1.1.3.5 Cost and size

Infrastructure-less nature of WSNs generally enables a reduction in the cost associated with installing the infrastructure and its maintenance. The cost of the individual sensor nodes and the associated maintenance, for example, replacing batteries is, however, required to be taken into consideration. Therefore, for a widespread use of wireless sensor technology, it is important that not only the cost of the individual sensor nodes is reasonable for the given applications but practically viable operational lifetime could also be achieved.

In relation to industrial condition monitoring in which sensor nodes might be deployed on to the space constrained locations of the machinery, the small size and the low weight is desirable to enable effective monitoring of the desired locations without causing mechanical imbalances.

1.2 Problem statement

A WSN, as an infrastructure-less network that requires no physical connectivity to enable communication and energy supply amongst its nodes, promises to enable cost-effective monitoring solutions both for existing wire-based monitoring applications and for those in which wire-based monitoring had not been feasible previously. However, untethered energy supply, in conjunction with low-cost and small size led to the development of wireless sensor nodes with restricted

energy, processing and communication resources, thus limiting WSNs to low-sample rate intermittent monitoring applications.

In order to analyze the resources of such wireless sensor nodes in relation to realizing data and computation intensive monitoring applications, firstly, some of the popular wireless sensor nodes are briefly described. Following on from that, the problem description is continued.

1.2.1 An overview of existing wireless sensor nodes

Starting from WeC, the first prototype sensor node which was developed at UC Berkeley in 1998, a large number of wireless sensor nodes have been developed as a result of academic research and commercial activity. Some of the popular and relevant in this regard, as summarized in **Table 1.1**, are discussed in the following.

1.2.1.1 Mica motes

The WeC integrated an 8-bit micro-controller AT90LS8535, from Atmel, that was operated at 4 MHz and had 512 B of internal Static Random Access Memory (SRAM). In order to perform wireless communication, the WeC was integrated with a radio transceiver, TR1000, operating at 915 MHz frequency band that provided a transmission rate of 10 kbps. Based on the development of the WeC, a number of sensor nodes with varying design issues related to communication, micro-controller, memory, size, etc. were developed at UC Berkely and a commercial venture Crossbow. These sensor nodes such as Rene, Mica, Mica2, Mica2Dot, MicaZ and Telos integrated 8/16-bits low-power micro-controllers with SRAM of up to 10 kB, and radio transceivers with a maximum data rate of 250 kbps. These detailed specifications for each of these can be found in **Table 1.1**. The typical power consumption of these nodes is less than 100 mW [95].

1.2.1.2 Eyes

Based on the 8-bit micro-controller, MSP430 from Texas Instruments, Eyes nodes [69] were developed by Infineon as part of a European Union funded project to enable energy efficient sensor networks. The architecture of the EYES is quite similar to the Mica motes; however, the latest versions of EYES motes, EYESIFXv1 and EYESIFXv2, were equipped with radio transceivers from Infineon that support data transmission rates of up to 64 kbps.

1.2.1.3 Medusa-MK 2

The MK-2 sensor node [96], developed at UCLA, integrates two micro-controllers from Atmel. One 8-bit micro-controller for baseband processing for the

Table 1.1 Technical specifications of some of the existing wireless sensor nodes

Sensor node	Processing unit, type *, name, (clock freq. MHz)	Wireless transceiver: type, name, (data rate)	Memory SRAM/Flash	Operating system
WeC [95]	µC: AT90LS8535 (4)	Radio – TR1000 (10 kbps)	512B / 32kB	Tiny OS
Renee [95]	µC: ATmega163 (4)	Radio – TR1000 (10 kbps)	1kB / 32kB	Tiny OS
Mica [93]	µC: ATmega103 (4)	Radio – TR1000 (40 kbps)	4kB /512kB	Tiny OS
Mica2Dot /Mica2 [94]	µC: ATmega128(7/4)	Radio – CC1000 (40 kbps)	4kB /512kB	Tiny OS
MicaZ [94]	µC: ATmega128(4)	Radio – CC2420 (250 kbps)	4kB /512kB	Tiny OS
Telos/ TelosB [95]	µC: TI's MSP430 (4)	Radio – CC2420 (250 kbps)	10kB / 1MB	Tiny OS
Eyes/Eyes IFX v1&2 [69]	µC: TI's MSP430 (5)	Radio – TR1001/ TDA5250 (115.2/64 kbps)	10kB / 8 MB	PeerOS/ Tiny OS
Medusa MK-2 [96]	µC: Atmel's Atmega128L & AT91FR4081(8/40)	Radio – TR1000 (10 kbps)	4kB / 4kB 136kB / 1MB	Palos
BTnode [97]	µC: ATmega128L (7/4)	Radio – CC1000 (40 kbps) & ZV4002 Bluetooth (1 Mbps)	(64 + 180 kB) / 128 kB	TinyOS
SENTIO-em [98]	µC: EFM32 (32)	Radio- XBee (250 kbps) & TI's CC1101 (500 kbps)	16/128 kB	N.A

IMote[99]	µC: ARM7TDMI (12)	Radio – Blue tooth	64kB / 512kB	TinyOS, Linux
Imote2 [48][100]	µC/P: Intel's PXA271(13-416)	Radio – CC2420 (250 kbps)	256kB / 32kB +32MB SDRAM	µOS
µAMPS[101]	µC/P: StrongARM SA-1110 (59-206)	Radio – Bluetooth (1 Mbps)	4MB / 4MB	TinyOS
25mm cube [102]	µC:+FPGA: Atmega 128L (-) + Spartan IIE XC2S300E-7FG256	Radio – nRF2401 (1 Mbps)	4+20 kB 128+524	N.A
mPlatform [103]	µC+CPLD+ µP : TI's MSP430 (8) + Xilinx CoolRunner CPLD + OKI ML67Q5003	µRadio – CC2420 (250 kbps)	10kB + . + 32kB /48kB + . + 512kB	N.A
Modular Architecture [104]	µC: + FPGA: ADuC812 (12) + Spartan IIIE XC3S200	Radio – Bluetooth	256 B + 30 kB /8 kB	N.A
Cookies Platform [105]	µC: + FPGA: ADuC831 (-) + Spartan IIIE XC3S200	Radio – Bluetooth (732.2 kbps)	2 kB + 30 kB /62 kB	N.A

µC: micro-controller; µP: micro-processor; CPLD: complex programmable logic device; FPGA: field programmable gate array

.radio and management activates, and the other one based on 32-bit ARM core to be used as an accelerator. It used the same radio transceiver as of the Mica mote

1.2.1.4 SENTIO-em

The SENTIO-em [98], a domain specific sensor node targeting low-power environmental monitoring applications, was developed at Mid Sweden University, Sweden. Built on Energy Micro's micro-controller, EFM 32, the node can be operated at a clock frequency of 32 MHz. The node has 16 kB of RAM and 128 kB of flash, in addition to a support for mounting an external SD card. For wireless communication, the node is enabled to integrate a commercially available XBee module, in addition to custom built communication layer integrating TI's CC1101 that offers data rates of up to 600 kbps.

1.2.1.5 Intel's motes

Imote [99], the first mote from Intel, was based on TC200P, a system-on-chip (SoC) module from Zeevo in which an ARM core and Bluetooth modules were integrated on a single chip. The Imote2 [100] was based on a high performance processor from Intel that can be operated from 13 to 416 MHz. In addition, it was integrated with SRAM, flash and SDRAM. Unlike Imote, the IEEE 802.15.4 complaint radio transceiver was used for wireless communication.

1.2.1.6 25mm cube

This sensor node [102] based on a modular structure was developed by Tyndall National Institute and University College Cork, Ireland. It consists of four 25mm square layers namely, communication, processing, sensors and power supply. The communication layer comprises of an 8-bit micro-controller and a radio transceiver from Nordic Semiconductor that supports data rates of up to 1 Mbps. The processing layer is comprised on a Spartan II FPGA from Xilinx. Unlike the long communications range of Mica2 mote, its communication is limited to 10 metres [102].

1.2.1.7 mPlatform

It is also a modular platform that supports different layers comprising of different combinations of micro-controllers and a CPLD [103]. The 8-bit micro-controller MSP430 from TI is used as a main controller. In addition to an CPLD from Xilinx, it also supports an ARM micro-controller that can be operated at clock frequencies of up to 60 MHz. For communication purposes, it uses IEEE 802.15.4 complaint radio transceiver supporting data transmition of up to 250 kbps.

1.2.1.8 Cookies

Starting with an FPGA-based modular architecture [104] comprising of processing, communication, sensing and power supply layer, researchers at Universidad Politecnica de Madrid (UPM) built several variants later called Cookies, HireCookies etc. In this first version [104], the processing layer was comprised of an 8-bit micro-controller from Analog Devices and a Spartan IIE FPGA from Xilinx. For communication purposes, a Bluetooth module OEMSPA13i from ConnectBlue was used. In the next upgraded version [105], in addition to the integration of ZigBee technology for communication, adoption of a different micro-controller from the same Analog Devices was reported. In the latest design, called HireCookie [106]-[106], the processing layer is comprised of Atiny 2313V micro-controller and Spartan-6 LX150 FPGA from Xilinx. In addition to the processing layers comprising of SRAM-based FPGA and micro-controller from Analog Devices, the development of processing layer comprising of flash-based FPGA from Actel's Igloo family of FPGAs and a TI's MSP430 is also reported in [106].

1.2.2 Problem statement (cont'd.)

From the technical specifications, as listed in **Table 1.1**, we observe that the wireless communication in the sensor nodes is typically enabled with the IEEE 802.15.4 or the IEEE 802.15.1 (Bluetooth) compatible transceivers that offer communication data rates of 250 kpbs and 1 Mbps, respectively [108]. It should be noted that these transceivers operate in a license free frequency band where several other devices, in addition to deployed sensor nodes, are in competition for the radio spectrum. Given the fact that the wireless communication is prone to external interferences and there is a significant overhead involved in secure and reliable communication, the actual throughput is less than the above mentioned data rates [109]-[111]. In low-sample rate applications, a throughput of tens of kbps is often sufficient. However, for data intensive applications, such as high-sample rate vibration and image-based industrial condition monitoring, the data rates achieved by these transceivers are unable to attain the desired communication gaols between a pair of wireless nodes, even in ideal conditions i.e. no external interferences. For example, multi-axes (e.g. tri-axes) vibration data acquired at a sampling rate of 50 kHz and sample resolution of 16-bits requires a communication throughput of 2.4 Mbps. In a similar manner, camera based nodes acquiring image data for transmission across the network also requires high-bandwidth communication support.

In order to overcome the limited communication bandwidth of wireless sensor nodes with regards to data intensive applications, an alternative, as motivated by in-network processing [112], is to process the data locally in the sensor node. By processing the data within the sensor node, which is termed as in-

sensor processing in this thesis, the small amount of extracted meaningful information could then be communicated through low-throughput radio transceivers. In relation to typical sensor nodes such as those discussed above and several others [69] that are based on 8/16-bit micro-controllers having a small amount of on-chip RAM and flash memories and offering limited processing performance, this may be feasible for applications involving relatively low-computation complexity. However, for computation intensive applications, for example, vibration monitoring that involves intensive signal processing algorithms such as digital filters, Fourier and wavelet transforms, correlation functions etc. [74]-[79], both the memory and processing capabilities become a bottleneck. In computation resourceful (micro-processor-based) nodes such as μAMPS and iMote2, the support for higher operating frequencies could definitely be counted as an advantage when compared to the micro-controller based nodes. However, the sequential processing and high-power consumption pose a challenge in achieving high processing performance with a sustainable lifetime. On the other hand, FPGA-based wireless sensor nodes that enable achieving high computational performance by parallelizing different tasks are constrained by power consumption, design complexity (both hardware and software) and the cost associated with FPGA-based designs. In particular, the high-static power and long-configuration time associated with SRAM FPGAs [113], which are widely used in computation intensive applications [114]-[117] because of their high-performance characteristics, pose a challenge in achieving long operational lifetimes.

With reference to the above mentioned limitations of communication and processing technologies, and the limited energy budget of typical wireless sensor nodes, this thesis explores an energy efficient architecture in order to efficiently realize data and computation intensive wireless monitoring applications.

1.3 Main contributions

In this thesis, an energy efficient wireless sensor node architecture is explored in relation to data and computation intensive monitoring applications. The research aspects addressed in the thesis and their relationship with the papers included in the thesis is depicted in **Figure 1.3**. The main scientific contributions are listed in the following.

1. *Analytical and quantitative evaluation of different architectural alternatives with regards to vibration and image-based data and computation intensive monitoring applications*

   Firstly, on the basis of energy consumption and cost, several standard-based wireless communication technologies are evaluated for raw-data transmission. Secondly, with respect to energy consumption and computational performance, in-sensor processing architectures comprising

of micro-controller and FPGA based processing units are realized and evaluated. Lastly, the most suitable choices among raw data transmission and in-sensor processing architectures are compared so as to obtain energy efficient solution. The results of this study could provide the guidelines to other designers and researchers working in this field.

2. *A feasibility study of an SRAM-based FPGA in relation to duty-cycle applications*

   With respect to duty-cycle applications that enable to switch-off the power to integrated modules in a wireless sensor node, the suitability of an SRAM-based FPGA is analyzed experimentally. In addition, a comparison is made with a flash-based FPGA.

3. *Design and development of high-performance wireless sensor node*

   Designed and developed an FPGA-based wireless sensor node that not only provides the means to achieve high-performance for computation intensive monitoring applications but also enables the dynamic optimization of the energy consumption.

4. *Quantitative evaluation of computation intensive processing using FPGA*

   A study is conducted to determine a boundary for which the FPGA-based wireless sensor node, when compared to the micro-controller based counterpart, achieves both the real-time processing performance and energy efficiency in relation to the computation intensive vibration mentoring application.

5. *Realization of data and computation intensive applications*

   The vibration and image-based industrial condition monitoring applications are realized in an energy efficient manner using the high-performance wireless platform.

1.4 Thesis outline

The remainder of the thesis is organized as follows. In chapter **2**, the applications used as the case study for the architectural evaluation are discussed in detail in relation to their practical usefulness, types, existing solution, computational algorithms and their complexity etc. In chapter **3**, different architectural alternatives for data and computation intensive monitoring applications are explored. The feasibility study of SRAM-FPGA for the duty-cycle application is presented in chapter **4**. The design and development of a high-performance wireless sensor node, the SENTIOF, is presented in chapter **5**. Chapter **6** is devoted to the implementation of vibration-based condition monitoring using SENTIOF and the associated results and discussion. In chapter **7**, the development

of image-based monitoring and the associated results and discussion is carried out. Chapter 8 presents the concluding remarks and future work.

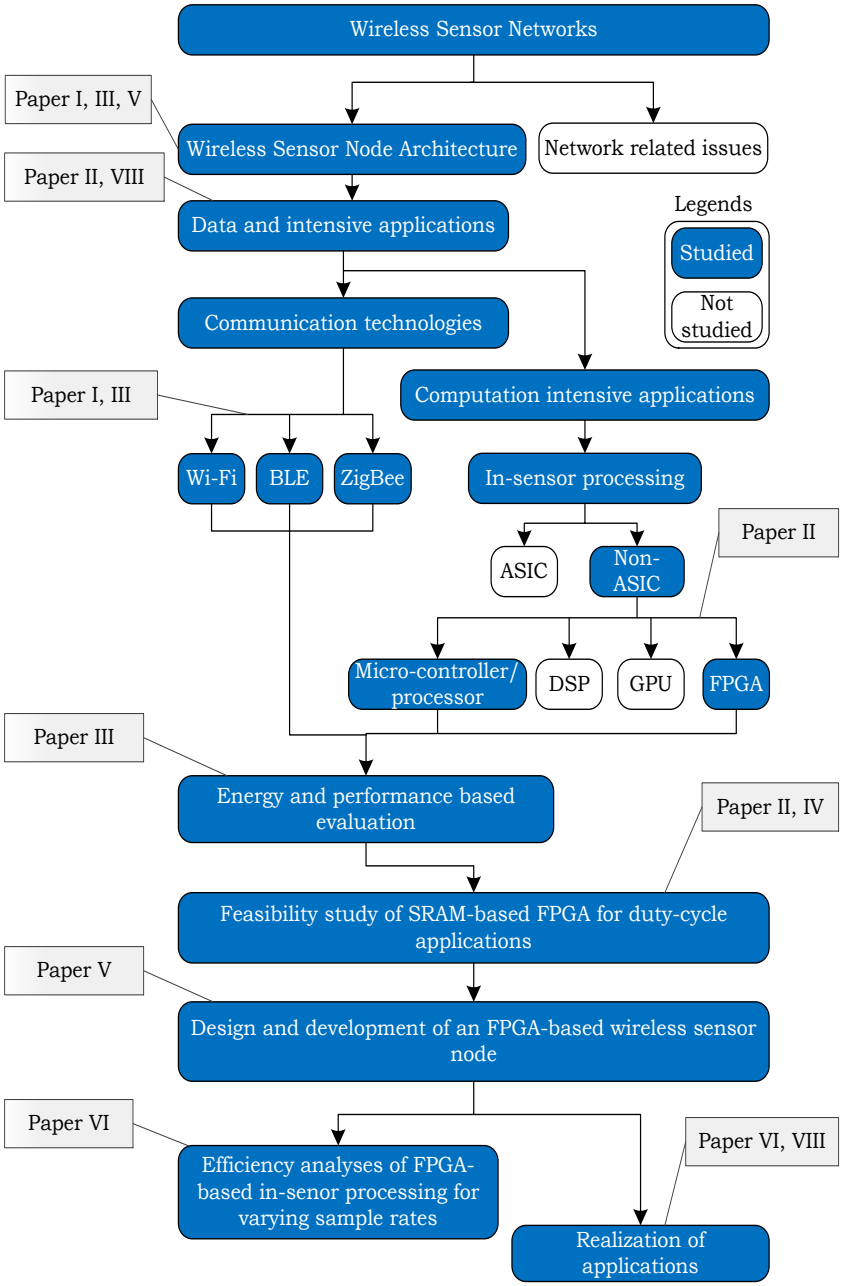

Figure 1.3 A graphical depiction of the research contributions of the thesis

2

Data and Computation Intensive Condition Monitoring Applications

In the present competitive market era, on-time production is important for many industries. The high costs involved in any production delays due to industrial equipment failure are therefore undesirable. Several maintenance methodologies are in practice to ensure the smooth functionality of the machinery in the process industry. These approaches can be broadly categorized as breakdown, preventive, predictive and proactive maintenance [70],[118]. In the breakdown approach, maintenance work is only initiated when a machine can no longer operate and because it is possible for a failure to occur at any time, therefore a longer maintenance time is required to detect and rectify the problem. Preventive maintenance overcomes the shortcomings of the breakdown approach by introducing schedule based repair work. The maintenance work is carried out at defined time intervals based on the operating hours of the machinery. However, this does not allow for the efficient utilization of the machinery and often results in unnecessary maintenance shutdowns. With regards to continuous process industries both the breakdown and preventive methods are not well suited. Predictive maintenance which involves condition based monitoring [119]-[120], on the other hand, offers the minimum downtime, as in this case the operating conditions of the machinery are regularly monitored and maintenance work is only performed when a problem has been identified in a machine. Proactive maintenance, which can be considered as being an extension of condition-based maintenance, aims to rectify the root causes of the problem so as to avoid future failures. Therefore, the key advantage associated with condition monitoring method lies not only in well planned maintenance work but also enables root causes of failures to be rectified. In relation to industrial equipment condition monitoring, both the vibration and image-based methods are widely accepted for detecting growing faults in defects. Due to their practical usefulness and importance as well as involving large amounts of data and computation intensive analysis, the study related to the architecture evaluation and exploration is mainly conducted in relation to these two applications.

2.1 Vibration-based condition monitoring

The importance of condition monitoring lies in the early detection of any underlying problems in order to minimize the downtime, maintenance costs and to maximize the lifetime of the machinery. There are several techniques in existence but vibration-based condition monitoring is the most widely accepted method for determining the growing defects and the performance degradation of rotating machinery [73]. Vibration data is either obtained by means of displacement, velocity or acceleration. The choice as to which of these three phenomena to be measured is dependent on the speed and type of machinery under observation and the suitability of the technology. As the amplitude of displacement vibration at higher frequencies is lower and therefore, signal-to-noise ratio (SNR) for displacement transducer is lower at higher frequencies as compared to that of at lower frequencies, therefore displacement vibration measurement is considered to be effective for machines operating at low speed. Similarly, velocity and acceleration data are considered as being effective for moderate and high frequency vibration analysis, respectively. As a mathematical relationship exists between these three, conversion from one to another can easily be performed by using a computer. For example, acceleration can be computed by differentiating the displacement twice, however, because of the noise amplification nature of the digital differentiation process, the general practice is to perform an integration process on the acceleration data in order to obtain the velocity and displacement equivalents [121]. Irrespective of whether it is the displacement, velocity or acceleration that is captured on a machine under observation, the acquired vibration data is typically analyzed in relation to that of the normal operating condition of machinery for possible faults.

2.1.1 Commonly diagnosed problems

In general, all rotating machines produce vibration when they are operated. These vibrations are typically a function of machine dynamics and the installation parameters such as alignment, balance, etc. Therefore, by measuring the vibration and relating it to the dynamics, alignment, and balance parameters, the health of a machine and its associated parts can be determined. Machinery faults that are commonly diagnosed using vibration data include unbalance, misalignment, structural looseness, broken rotor bars, bearing looseness and rubbing, belt misalignment and wear, gear misalignment and wear, loose rotor bars and an uneven air gap [122]. It has been observed that most of the problems in rotating machinery are due to imbalance, misalignment and looseness, which are best detected by means of vibration analysis.

2.1.2 Vibration-based condition monitoring methodology

In order to assess the condition of a machine on the basis of vibration, the first step is to acquire the vibration data using appropriate sensors measuring displacement, velocity and/or acceleration. The vibration data is then required to be analyzed in relation to vibration levels (amplitude, frequency, phase etc.) corresponding to the normal operating condition of the machinery or its parts. For time-domain vibration data, the analysis can either be conducted in time, frequency and time-frequency domain [123]-[123], as shown in **Figure 2.1**.

2.1.2.1 Time-domain analysis

Features that are commonly extracted from time-domain vibration data include peak values, mean, RMS, variance, kurtosis, shape of the envelope of the vibration signal etc. Time-domain analysis, though useful in diagnosing certain fault conditions [125], is highly susceptible to instantaneous disturbances in the vibration signals and therefore, is typically supplemented with other analysis.

2.1.2.2 Frequency-domain analysis

The frequency-domain analysis enables the energy present in desired frequency components of the vibration signal to be measured, which in turn can easily be related to defect frequency components. The Fast Fourier Transform (FFT) is a commonly used method to obtain frequency domain information from time domain vibration signals. Other algorithms used for frequency domain analysis include computing correlation of spectrum, averaging, power spectrum etc.

2.1.2.3 Time-Frequency domain analysis

The time-frequency domain analysis, typically conducted using wavelet transforms or spectrograms, provides the means to analyze non-stationary vibration signals in relation to both the time and frequency.

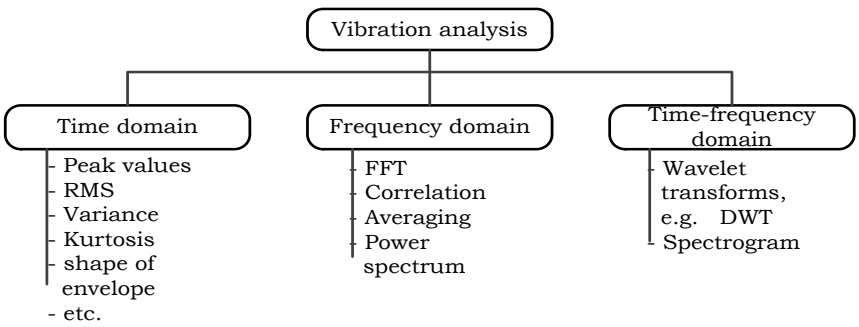

Figure 2.1 Vibration based feature extraction methods

2.1.3 Complexity analysis

In order to suppress the large amount of raw data acquired for a high-sample rate vibration monitoring application, processing data within a sensor node and transmitting only the resultant information could be an appropriate alternative. However, as summarized in **Table 2.1**, most of the vibration analysis functions involve intensive processing.

Although, the statistical functions such as those for calculating peak value, RMS, variance, kurtosis etc. are relatively simple. When these are combined with other functions, and in particular for large data sets, these may add a good share to the complexity of vibration analysis. The commonly used functions such as FFT, correlations, filtering, etc. require significant processing resources. For example, for a data set comprising of N samples, the FFT requires NLogN complex multiply and addition operations. It should be noted that one complex multiplication requires four real multiplication and two addition operations. In terms of memory requirements, except for input data, and intermediate and output results, the twiddle factors are also required to be buffered. It is worth mentioning that the output of an FFT function is in complex format, so it requires one buffer for real and another for the imaginary part. The power spectrum, which enables an analysis of the energy present in the vibration spectrum and is most frequently used analysis method, requires more operations than that of an FFT. The spectrogram, which provides both the time and frequency related information, does so by computing a large number of Fourier transforms of overlapped data sets and accumulating the output of each transform. The complexity of the spectrogram is much higher than that of the power spectrum, and requires much larger memory for the output.

2.1.4 WSN and vibration-based condition monitoring

With the developments in the technology, practices and methods, vibration-based condition monitoring evolved from pure mechanical devices, capable of only depicting complex time-domain vibration signals, to portable digital analyzers and automated digital computer based experts systems with advance analysis capabilities. The continuous monitoring of assets using expert systems generally assists in avoiding unexpected failures, however, the high cost associated with initial setup and maintenance of these systems often restricts their utilization to highly critical machinery in high-end industry. Portable analyzers are typically used for irregular monitoring under the supervision of skilled analysts. On the other hand wireless sensor networks are capable of providing low-cost continuous monitoring solutions on a large scale, and therefore, are considered as enabling technologies for the next generation industrial condition monitoring [129].

Along with other application areas, examples of WSNs prototyping, test and deployment in relation to industrial condition monitoring can also be found in the literature. Many of these studies are related to analysing the performance of induction motors on the basis of slowly changing parameters such as temperature and/or current [130]-[132], thus, requiring relatively low-sampling rates. Nevertheless, there are few studies, as discussed below, in which vibration-based condition monitoring is addressed.

In one of the earliest study in relation to vibration-based condition monitoring [133], the potential of WSN in condition based monitoring is demonstrated using a single hop wireless sensor network. The acquired data was transmitted wirelessly to a remote station where processing such as FFT and kurtosis was conducted using a general purpose computer through LabView software. In [134], a WSN based condition monitoring in end-milling was reported. In the experimental work, an MSP430 based sensor node with IEEE 802.15.4 compliant radio transceiver was used. The 12-bit two channel vibration data was sampled at 1 kHz. B. Lu et.el in [135] demonstrated the potential of WSN-based condition monitoring and energy evaluation for an electric machine. In their design analysis, the authors assume that in an industrial based condition monitoring, the

Table 2.1 Computational complexity of vibration analysis functions

Function	**Complexity Add-Sub / Mult-Div**	**Memory**
Peak value	(N-1) / -	Mainly for input data
RMS	(N-1) / (N+1)	Mainly for input data
Variance	(2N+2) / (N+2)	Mainly for input data
Kurtosis	(3N-1) / (4N+3)	Mainly for input data
Shape detection (using Hilbert transform) [126]	1 MAC per sample per coefficient	N. A
FFT	NlogN complex operations	Input data, Output, and twiddle factors
Correlation [127]	Same as that of FFT	Similar to that of the FFT
Averaging	N / N	Mainly for input data
Power spectrum	Same as of the FFT + N additions and 2N multiplications	Similar to that of the FFT
DWT [128]	O(N)	N. A
Spectrogram	N x (window size – overlap) x NLogN	
FIR filter	Number of coefficients x N	For, intermediate and final results, and for coefficients

power is generally available from main outlets and therefore, there is no issue regarding the energy consumption of the network. In [136], a micro-controller and FPGA based sensor node is proposed for processing ground vibration signals typically dominant in frequencies less than 10 Hz. No performance or energy consumption measure is reported. However, the authors acknowledge that the power consumption of the solution is a problem for long operating requirements. In relation to high-sample rate vibration-based condition monitoring for pumps, a complex architecture is proposed in [137]. In this architecture, an FPGA is used for data acquisition, a DSP for data processing, and a ZigBee module (consisting of a micro-controller and radio transceiver) for data communication. Some results regarding diagnostic are reported however, no information relating to performance and power consumption of the proposed system is given. Sandra et.el in [138] presented an FPGA based wireless sensor node for monitoring vibration parameters at a low-sample rate, i.e. 1500 Hz, for computer numerically controller (CNC) machines. In the [139], Liqun et. el. analyzed the system requirement for industrial condition monitoring involving low-sample rate vibration data. The emphasis of their work, however, has been on proposing some modification for the media access (MAC) layer of the IEEE. 802.15.4 protocol, so as to enhance its performance in relation to heterogeneous signals, energy consumption and real time.

In the case of high-sample vibration monitoring and analysis, high-performance processing and communication architectures are required. In infrastructure based monitoring, the targets architectures include general purpose computers [140]-[143], DSPs [144]-[146] and FPGAs [147]-[148]. However, integrating such resources in a wireless sensor node causes several challenges to be faced such as high-power consumption, cost, size, design complexity etc. In this thesis, these challenges are analyzed in order to design a wireless sensor node that offers high-performance processing and communication capabilities while keeping other aspects such as energy consumption, size, cost etc. at a practically viable level.

2.2 Image-based condition monitoring

The mechanical equipment in industry typically degrades over time and with careful inspection, possible measures can be taken so as to avoid any unexpected failure and even to improve the performance of the machinery. In the above, we analyzed that the vibration-based condition monitoring is commonly used for this purpose, espically for rotating machinery. Additionally, another condition monitoring technique commonly used in the industry is to analyze the oil in the machinery for any signs of possible degradation [80]-[83]. The oil analyses are typically based on monitoring the viscosity of the oil, color of the oil, and presence of debris wearing from the machinery. In order to perform oil-based analysis, apart from viscosity data that is often acquired with the associated sensors, the color,

presence of water wapors and debris etc. is typically visually inspected by the technicians. With regards to employing wireless sensor networks, oil-based condition monitoring can be performed in a more automated and cost-effective way. In particular, the visual inspection can be replaced by wireless sensor networks equipped with image-monitoring capabilities.

2.2.1 Image-based condition monitoring methodology

In relation to analyzing the color of the oil, presence of water wapors and debris etc. using a camera-based sensor node, the first step is to acquire the image data for particular locations in the machinery. Following on from this, the data is required to be processed so as to extract meaningful information that enables the condition of the machinery to be assessed. The typical processing functions required for this purpose are same as those used in the machine vision applications [149].

The tasks involved in a typical vision processing algorithm are shown in **Figure 2.2**. An image, acquired using a video or an image sensor and represented in two-dimensional data, is processed through several stages so as to extract meaningful information. During the pre-processing stage, an image is enhanced by normalizing the intensity of pixels and removing noise [150], before further processing. This is typically achieved by filtering [151], and is sometimes supplemented by other methods such as background subtraction etc [173]. During the segmentation stage, as its name suggests, an image is segmented/partitioned in to multiple smaller sections on the basis of certain similarities such as intensity, texture, color, etc. Common methods for the image segmentation include thresholding (fixed and adaptive), edge-detection, fuzzy sets etc. [152]-[153]. Segmenting an image, for example by employing intensity thresholding, desired features or objects are separated from the background image. However, for further processing, these are required to be identified and by using functions such as labelling, feature extraction, and classification. The process of identifying and labelling each individual object falls under labelling category. During the morphological stage the information enabling the description and representation of shapes is extracted.

2.2.2 Complexity analysis

In a similar manner to that of the high-sample rate vibration monitoring, the image-based monitoring generates a large amount of data that is required to be transmitted across a wireless senor network for further processing and analysis at a

Figure 2.2 Tasks and their data flow in a typical vision processing algorithm

central station. In order to reduce the amount of data to be transmitted wirelessly, in-sensor processing can be an appropriate choice. However, it would require the necessary computational resources in the sensor node to process the large image data through several intensive image and signal processing functions.

Depending upon the requirements of applications being realized, the actual processing algorithm and associated computational complexity could be different from one application to other. For example, as mentioned above, the segmentation can be performed using several methods including fixed and adaptive thresholding, edge detection, region based etc. Given a large number of image processing functions and their variations [154], it is a challenging task to characterize all of these in terms of their computation complexity. In relation to typical low-power wireless sensor node integrating micro-controllers as processing resources, the image-based wireless monitoring node requires higher-computational resources other than these micro-controllers in order to process the image data.

2.2.3 WSN and image-based condition monitoring

With regards to realizing image-based monitoring applications using wireless sensor networks, several studies covering the potential applications, research challenges and application specific implementations have been reported [155]-[161]. In comparison with typical low-sample WSN applications, the major challenge in image-based wireless monitoring applications is to transmit the large amount of data across the network in a reliable and efficient manner using a wireless sensor node with limited energy, communication and computation resources.

In order to enable a sensor node with the necessary resources for image acquisition and processing, there are several camera-based sensor nodes such as SensEye [162], MeshEye [163], CMUcam3 [164], and others [156]-[157],[165] that have been reported in the literature. Most of these nodes integrate high-performance sequential processors to enable image processing locally in the node. For communication purposes, low-power and low-data rate radio transceivers such as those IEEE 802.15.4 compliant are used generally.

In relation to realizing energy efficient and high-performance sensor nodes for data and computation intensive applications, our aim is to develop a hardware architecture that not only caters for the needs of a particular application rather provides the means to enable energy efficient realization for a wide variety of applications including image-based monitoring.

3

Architecture Exploration

Enabling reliable, robust and cost-effective WSNs for data and computation intensive applications requires energy efficient wireless sensor nodes that provide the means to realize processing and communication related tasks in an efficient manner. In order to design such a sensor node, the analytical and experimental evaluation of different architectural alternatives performed in relation to the vibration and image-based data and computation intensive applications are discussed in this chapter.

3.1 Architectural design alternatives

In order to explore an energy efficient hardware architecture for data and computation intensive wireless monitoring applications, the choice of raw data transmission or partitioning feature extraction algorithms to determine the correct amount of in-sensor processing and result transmission, in conjunction with different hardware technologies, is a promising beginning.

3.1.1 Raw data transmission

Wireless sensor networks are typically aimed at collecting raw data or computed information at a central station so as to take appropriate actions on the basis of analysis of that data or information. In relation to wireless vibration and image-based condition monitoring applications, it is assumed that collecting raw data that can be processed in general purpose computers provides an important means as to assess the condition of assets and take the necessary actions to avoid any unexpected failures. Therefore, the first architectural choice, just like many other WSN applications, is to transmit raw data across the network using single or multi-hop routing topology. However, unlike typical low-sample rate WSN applications, the amount of raw data is large for these applications. For example, in the case of vibration-based industrial condition monitoring, depending on the type of machinery and expected faults that may arise, the desired sampling frequency typically varies from a few kHz to more than a hundred kHz, and data resolution

from 12-bits to 16-bits [74]-[79],[137],[166]. In addition, with the requirements to measure vibration at multiple axes such as at horizontal, vertical and axial axes, the amount of raw data can grow up to several Mbps, which is much larger than the maximum transmission rate of many low-power wireless transceivers. In a similar manner to that of the vibration, a gray scale image of 640 x 480 resolution, depending upon the number of frames captured per second, would require a transmission rate of more up to tens of Mbps.

Finding an appropriate radio transceiver, which can fulfil the throughput requirements of high-data rate applications while consuming low energy, the different alternatives that are available in terms of standard-based wireless communication technologies are depicted in **Figure 3.1**. In relation to a low-energy, low-cost and compact size wireless sensor node, an analytical evaluation of these technologies enables a choice to be made regarding feasible alternatives for quantitative comparison. The practical infrared transceivers can be used to achieve data rates of up to several Mbps. These transceivers are cheap and easy to integrate in a sensor node. However, their communication range is limited. In addition, both the sender and the receiver are required to be in the line-of-sight in order to achieve the desired communication. Therefore, infrared communication can be better suited to specific operating conditions, such as underwater communication etc. [167]. Among radio communication alternatives, both the mobile broadband and the WiMax offer data rates well above the requirements of the above mentioned application scenarios. However, both the cost and the power consumption associated with integrating these technologies into each node of a sensor network pose a challenge in realizing a cost effective sensor network that can achieve a long operational lifetime. Therefore, these technologies can be better suited to interconnecting physically dispersed wireless networks [168]-[169].

On the other hand, the typical cost, power consumption, and coverage of the Wi-Fi, Bluetooth, in particular, Bluetooth Low Energy (BLE) and ZigBeee, apparently appears to be promising in realizing low-power and low-cost nodes

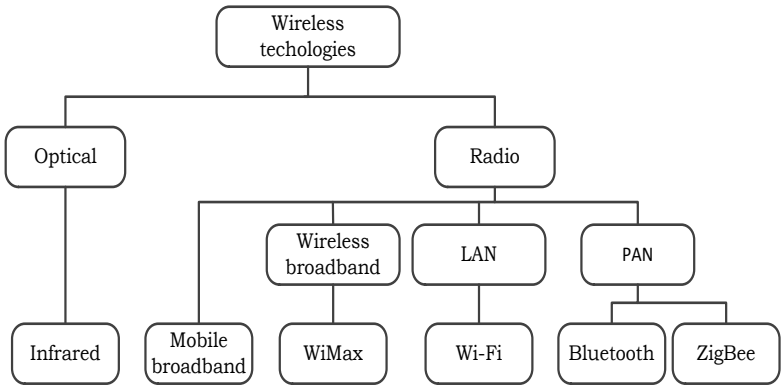

Figure 3.1 Standard based wireless communication technologies

communicating over a short range. A comparison of these three standards is given in **Table 3.1**. In terms of real-time continuous monitoring, among these three standards, it is only the Wi-Fi that offers the data rates required for the vibration and image-based applications as discussed above. However, given the actual throughput of typical Wi-Fi transceivers in real environments, in particular in industrial environments where several interferences are present, the real time data transmission could be challenging for data intensive applications [170]-[171]. Nevertheless, in a similar manner to typical WSN applications, many data intensive applications may be operated in a duty-cycle manner so as to conserve power, and thus, it is motivating to include Wi-Fi, ZigBee and Bluetooth for evaluating architectures. Furthermore, the low-power consumption of the wireless transceivers for these standards would enable to assess the energy efficiency of in-sensor processing architectures.

3.1.2 In-sensor processing

In this section, the in-sensor processing is analyzed so as to obtain the performance and energy consumption related results. The results are then evaluated with regards to different architectural alternative, including raw-data transmission.

3.1.2.1 In-sensor processing for vibration-based monitoring

In relation to vibration-based condition monitoring and analysis, the vibration data processing algorithm shown in **Figure 3.2** is realized on three in-sensor processing architectures. The processing algorithm comprises of a number of

Table 3.1 A comparison of different parameters for ZigBee, BLE and Wi-Fi

Standard	ZigBee	BLE	Wi-Fi
IEEE Specs	802.15.4	802.15.1	802.11 b/g/n
Frequency spectrum	868/915 MHz; 2.4 GHz	2.4 GHz	2.4 GHz; 5 GHz
Topology	Star, mesh, cluster tree	Star, point-to-point	Start, point-to-point
Network size	65536	Not defined	32
Data rate (Mbsps)	0.02 – 0.25	1	11/ 54/600
System resources	4 kB – 32 kB	-	1 MB+
Range (m)	< 100	< 50	< 100
Number of channels	1;10;16	40	11-14 (only three are orthogonal)
Security	128-AES	128-AES	SSID

signal processing functions so as to compute the meaningful information. At data acquisition stage high-resolution (i.e. 16-bits) vibration data for tri-axes (horizontal, vertical and axial axes) is acquired at a high sampling rate of 50 kHz. During the pre-processing stage, the raw vibration data is converted into acceleration. Following on from that, time domain acceleration data is filtered using a low-pass Finite Impulse Response (FIR). This enables high frequency content in the vibration data to be removed while relaxing the design requirements of the analog anti-aliasing filter. The realized filter has the following specifications: filter order of 130, cut-off frequency of 20 kHz, transition band of 20 kHz to 22 kHz, pass band ripple of 0.001 dB, and stop band attenuation of -92 dB. The above mentioned transition band was chosen in accordance with the specifications of the transducer used. In addition to the FIR filter, a hamming window because of its high performance and relatively low computational complexity characteristics was used to minimize the spectral leakage in following Fast Fourier Transform (FFT). Following on from this windowing, power spectrum is computed using FFT. As an FFT of an N-point real data produces duplicate information in the frequency spectrum thus, only $\mathbf{N/2 + 1}$ output points were used for further processing. In the analysis stage (also called spectrum processing in our articles), a commonly used vibration analysis method of comparing the currently computed frequency spectra with a predefined threshold level and a spectrum obtained during a normal operating condition of the machine is opted. For this comparison, three different amounts of resultant information are generated so as to analyze the performance and energy consumption of the architecture under evaluation. The resultant information to be transmitted varies from low (1 byte for each axes) to medium (N-bits per spectrum) and to full spectrum at high-resolution. These spectrum analyses leading to these three variations in resultant information are abbreviated as 1BPA, NbPA, and FSPA, respectively, for the later reference. In the final phase of wireless vibration monitoring and analysis, the results are transmitted through a wireless transceiver.

For the in-sensor processing, three architectures based on a sequential processor consisting of a micro-controller, a high-performance processing unit consisting of an FPGA and combination of both are realized. In addition, a simple architecture supporting only raw-data transmission was realized to relate the performance and energy consumption of in-sensor processing architectures. A graphical depiction of these architecture showing integrated modules and the data flow are given in **Figure 3.3**. In these architectures, the Sensing Subsystem is comprised of accelerometer sensors, anti-aliasing filters and ADCs. For realizing vibration data processing algorithm in a micro-controller based architecture, a

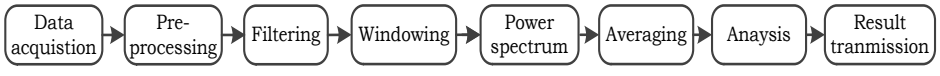

Figure 3.2 Vibration data processing algorithm and data flow

wireless embedded platform, SETNIO32 developed at Mid Sweden University is used. The SENTIO32 comprises a 32-bit micro-controller, AVR32UC3B and the IEEE 802.15.4 complaint radio transceiver, CC2530. The FPGA related tasks are verified on a commercially available evaluation board, SP605 that integrates Spartan 6 FPGA, XC6SLX45. This FPGA consists of large quantity of logic resources than required for the vibration data processing algorithm and the evaluation board does not facilitate measurement of power consumption. The power consumption was obtained for the XC6SLX16 FPGA that provides sufficient resources for this application using Xilinx's power analysis tool, XPower. In addition, the power consumption of micro-controller and radio transceiver was measured using SENTIO32. For other modules such as memory, the specification provided in the datasheet by their manufacturer was used to estimate the overall power consumption of each architecture. The detailed hardware specifications of these architectures, including the performance and power-consumption of each module, can be found in paper II that is included at the end of this thesis. Based on power consumption and execution obtained for each individual functional task for the corresponding hardware unit, all architectures were modelled to analyze the performance and energy consumption.

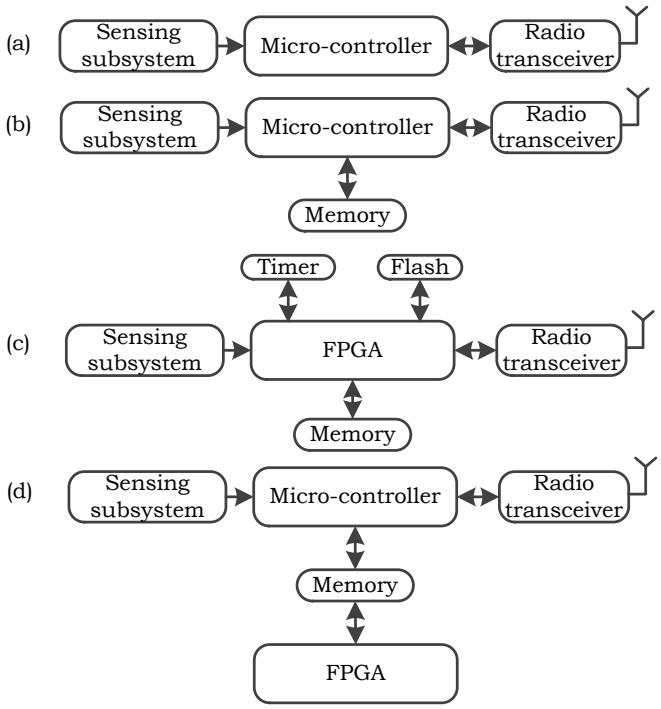

Figure 3.3 Architectures explored (a) architecture I for raw data transmission (b) architecture II: Micro-controller based architecture (c) architecture III: FPGA based architecture (d) architecture IV combines a micro-controller and an FPGA in a vibration analyzer

The performance and energy consumption of all of the architectures is analyzed in relation to a sampling frequency of 50 kHz. The choice of the sampling frequency was made as per the following two reasons. Firstly, to observe the bottleneck associated with high-sample rate vibration data monitoring. Secondly, as it satisfies the Nyquist's criteria to acquire vibration data from a maximum bandwidth MEMS accelerator available in the market.

In architecture I, the vibration data from three accelerometers representing horizontal, vertical and axial vibration, was acquired and transmitted to a nearby node using the IEEE 802.15.4 communication protocol. In architecture II, the acquired vibration data is processed in the micro-controller before transmitting the final results. Unlike architecture II in which vibration data is processed in a sequential manner, in architecture III the data processing for all three axes was performed in parallel. In architecture IV, the data acquisition and results transmission were performed by the micro-controller. However, once the required number of data samples were buffered in the micro-controller, the FPGA was activated to process that data. Unlike architectures I, II, and IV, in which the micro-controller was used to perform control specific operations including the power management, in architecture III, an external timer was assumed to power up the FPGA for duty-cycle operational states.

In relation to real-time performance for the vibration data processing at a sampling frequency of 50 kHz, it was observed that only architectures III and IV, in which an FPGA is used to processes vibration data, are able to deliver the desired performance. In comparison to the sequential processing of architecture II, the performance gains in architectures III and IV were achieved due to a higher clock frequency (i.e. 100 MHz as compared to 60 MHz of micro-controller) and hardware parallelism. In addition, as shown in **Figure 3.4** it was also observed that

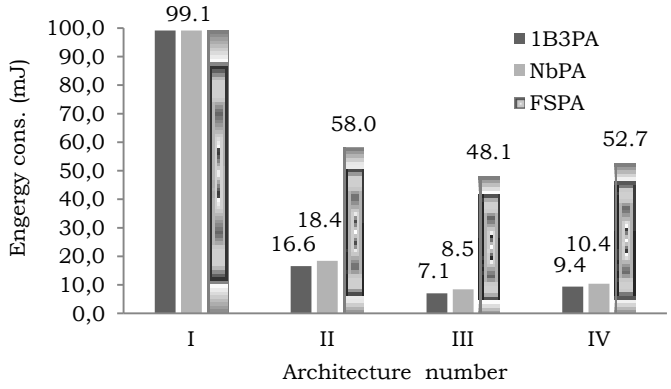

Figure 3.4 Absolute energy consumption of the architecture I, II, III and IV for vibration data of 4096 (4k) samples

both the architectures III and IV also consumed less energy as compared to architecture II. Based on a single 2100 mA battery [172], similar to that of a typical AA battery in size and shape, the operational lifetimes for these architectures were estimated. The achievable operational for continuous monitoring were 54 hours to 95 hours thus, enabling these architectures to be used effectively for scenarios in which machinery is required to be monitored continuously for several days. However, for intermittent monitoring such as after every five minute, the estimated operational lifetime was about 1.5 years for architectures III and IV, which can be considered quite reasonable for long term monitoring without frequent replacement of the batteries.

3.1.2.2 In-sensor processing for image-based monitoring

In order to evaluate the performance and energy consumption for in-sensor processing of image-based monitoring applications, it was observed that the results of an existing study [173], dealing with the implementation of a wireless sensor node for image-based particle characterization in fluid, could serve the purpose and thus, be used to avoid costs associated with design and development of these architectural alternatives. In this study, architectural evaluation is conducted for an image-based wireless sensor node aimed at analyzing the presence of foreign particles in the oil of a hydraulic machine. The evaluation process is conducted by varying processing and communication loads on the targeted architecture and observing the associated energy consumption so as to find a balanced amount of processing and communication load for the in-sensor processing and wireless transmission.

The different processing tasks on the basis of which the evaluation was conducted are shown in **Figure 3.5**. For an image of 640x400 pixels resolution acquired using Micron Imaging camera MT9V032, the minimum number of processing tasks performed on the sensor node include pre-processing and

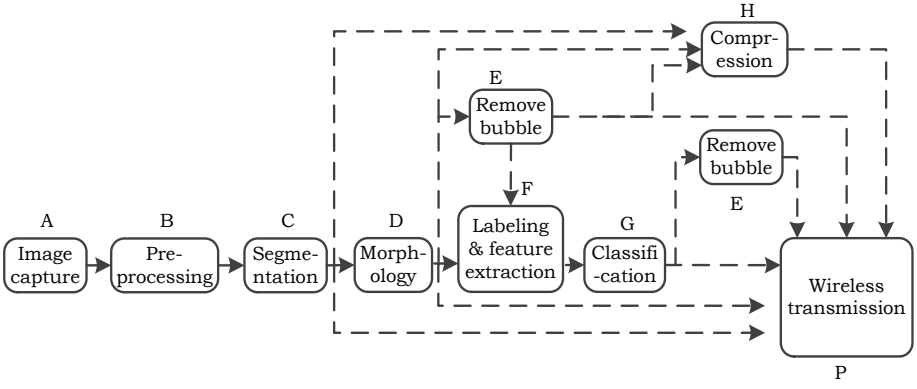

Figure 3.5 Image processing tasks and their data flow, as used for evaluation [173]

segmentation before transmitting the result using the radio transceiver. In the pre-processing, a previously acquired background image is loaded from the flash memory and then subtracted from the current image frame so as to highlight particles. Following on from pre-processing, based on fixed thresholding, segmentation is performed to distinguish mutually exclusive connected regions. Other alternatives (called "partition strategies") reported in the study include various combinations of transmitting both the compressed data after morphology and bubble removal operations and un-compressed data after labelling, classification and bubble removal operations. The morphological operation, realized through erosion followed by a dilation operation, is performed to remove small (represented by one or two pixel) objects. In the bubble remove process, air bubbles present in an image are first identified mainly based on their movement and then removed from the image. During the labelling stage, connection regions are assigned unique identifiers so as to extract features. After feature extraction, objects are classified according to area, intensity and/or position. In the case of compression, the bi-level compression scheme ITUT-G4 is reported for the evaluation study.

The target hardware architecture includes an ACTEL's flash-based FPGA, AGL600V5, for processing image data in the sensor node. The power consumption is estimated mainly based on this FPGA. However, the partition strategies for which the above mentioned FPGA does not offer the required resources, the (dynamic) power estimation is based on Xilinx Spartan 6 XC6SLX9. The power consumption associated with that micro-controller and radio transceiver is the same as that of SENTIO32.

In comparison to processing fewer tasks on the sensor node and transmitting resultant information using the IEEE 802.15.4 compatible radio transceiver, CC2520, processing all of the required tasks (ABCDEFG as shown in **Figure 3.5**) on an FPGA, resulted in a much higher performance, i.e. number of frames processed per second. In addition, the energy consumption estimates show that the processing of all the required tasks in wireless sensor node is also highly energy efficient. The minimum energy consumption, 7.19 mJ resulted in processing of all the required tasks in the sensor node among all of the alternative partition strategies.

3.2 Evaluation of in-sensor processing vs. raw data transmission architectures

In order to evaluate the performance and energy consumption associated with in-sensor processing of the above mentioned applications in comparison with that of raw-data transmission using ZigBee, Bluetooth low-energy and Wi-Fi, the typical data rates of the these communication, as shown in **Table 3.2** are assumed.

This is done to ensure that if the performance obtained with in-sensor processing is better, as compared to that of the raw data transmission using ideal channel conditions, then, it would certainly be in practical conditions that typically involve several interferences.

Regarding energy consumption, the parameters listed in the manufacturer's datasheet for the selected representative chipset of these communication technologies, as shown in **Table 3.2**, is used. In the energy estimation for a given amount of data transmission, it is assumed that two nodes, a wireless sensor node and a base station are equipped with compatible transceivers that enable maximum data rates. In addition to the wireless sensor node transmitting at the maximum allowable data rate and packet size, it is also assumed that the base station acknowledges the reception of the data using a minimum allowable packet size, and there is no packet loss between the two devices.

3.2.1 Performance based evaluation

In the case of continuous monitoring involving raw data transmission for the above mentioned vibration and image-based monitoring applications, the transmission data rate of 2.4 Mbps is required. It should be noted that for the image-based monitoring application, the data rate is highly dependent upon the number of frames captured per second. The above mentioned requirement

Table 3.2 Detailed parameters of ZigBee, Bluetooth and Wi-Fi modules used in the evaluation of in-sensor processing and raw data transmission

Standard		ZigBee	Bluetooth	Wi-Fi
Chipset		(CC2520)	(CC2540)	(CC3000)
Operating Voltage (V)		3.0	3.0	3.6
Current consumption	Deep sleep (μA)	0.03	0.4	0.5*
	Idle mode (mA)	1.6	N.A	N.A
	RX mode (mA)	18.5	15.8	92.0
	TX mode (mA)	25.8	21.0	190.0
Sleep to TX/RX time (ms)		0.5	0.5	60*
Connection time (ms)		15	400	4000
Data rate (Mbps)		0.25	1	54
Data packet size (bytes)		127	47	2346
Maximum payload size (bytes)		102	37	2312
Ack. packet size (bytes)		11	10**	14

*Shut-down ; **Data PDU without payload

corresponds to the assumption that only one frame, of 640x400 pixels, per second is sufficient for this application. By analyzing the typical data rates of the ZigBee, BLE, and Wi-Fi, as given in **Table 3.2**, it can be observed that only the Wi-Fi could potentially deliver the required transmission rate. Therefore, we analyze the power consumption of the Wi-Fi to that of in-sensor processing of the above mentioned application for continuous monitoring.

The maximum power consumption of any architectural alternative pertaining to vibration-based monitoring application is 127.3 mW. In the case of image-based monitoring application, the power consumption associated with processing all of the required tasks in the sensor node is about 360 mW, in which about 160 mW corresponds to the camera, and the remainder being divided among the FPGA, radio transceiver and the micro-controller used for controlling the radio transceiver. For the Wi-Fi transceiver, which we assume that most of the time is in transmit mode due to the large amount of data for transmission as compared to the receiving mode listening for acknowledgement, consumes more than 680 mW (190 mA x 3.6 V). It should also be noted that this power consumption does not include a controller required to acquire data from the camera and to control the Wi-Fi transceiver. With this comparison, we observe that none of three communication technologies are more efficient in realizing the data intensive monitoring applications in a continuous operational mode.

The lifetime of the above mentioned applications, when operated continuously, could last from few days to several weeks, depending upon the capacity a nominal sized battery. Therefore, in most of data and computation intensive applications it would be desirable to monitor the desired parameters at a regular interval, typically in a much higher frequency than a manual logging, so as to avoid replacing battery too often. This is particularly important to the wireless sensor node deployed in hard to reach places as well as non-stationary parts of industrial equipment. In such an intermittent monitoring scenario, in addition to Wi-Fi, both the ZigBee and BLE could be used to transmit raw data. However, in this case the most important parameter to evaluate in-sensor processing with raw data transmission is the energy consumption.

3.2.2 Energy consumption based evaluation

As opposed to continuous monitoring, in the case of monitoring at a specified regular interval, i.e. applying duty-cycling, a sensor node remains active only during the time of actual monitoring (i.e. data acquisition, processing and result transmission) and is otherwise switched to a sleep state so as to conserve power. This typically enables the power consumption to be reduced from hundreds of milli-watts in active state to tens of micro-watts in sleep state. Given the low-power consumption during sleep state and relatively long sleep interval, for example, for the above mentioned applications it could be from few minutes to several hours, we

assume that the overall energy consumption during sleep state in case of either raw data transmission or in-sensor processing is comparable. And therefore, it is the energy consumption during active state which should be analyzed for quantitative analysis.

In relation to a wireless sensor node that conserves energy by operating in duty-cycle manner, upon every wakeup it is required to establish a connection for a successful transmission. Therefore, it is important to include both the energy consumed during actual data transfer as well as that in establishing a connection for comparative analysis. From the parameters given in **Table 3.2**, it is evident that in terms of energy consumption per bit, the Wi-Fi results in the minimum consumption whereas, ZigBee consumes the maximum. However, energy consumption during connection establishment process and the amount of data to be transmitted plays an important role in the overall energy consumption. This can be observed in **Figure 3.6**, in which the energy consumption for a given data includes both the energy consumed in establishing the connection and the energy consumed in actual data transmission. From this figure, we observe that the ZigBee better suits when a small amount of data is to be transmitted, i.e. up to 500 bytes. The Wi-Fi, on the other hand, consumes the least amount of energy so as to transmit 800 kB or more data. For a wide range of data transmission i.e. 500 bytes to 800 kB, the BLE results in an energy efficient solution.

In order to evaluate the energy consumption associated with raw-data transmission to that of the in-sensor processing, we need to have an amount of data that is monitored during each active phase. In the case of vibration-based condition monitoring, we observe the one data set of 4096 samples for three axis vibration of 16-bits resolution for which, the minimum amount of data is 24 kB. However, in practical applications, often several data sets, typically four to six, are analyzed so as to assess the condition of machinery. Therefore, the data size could grow to

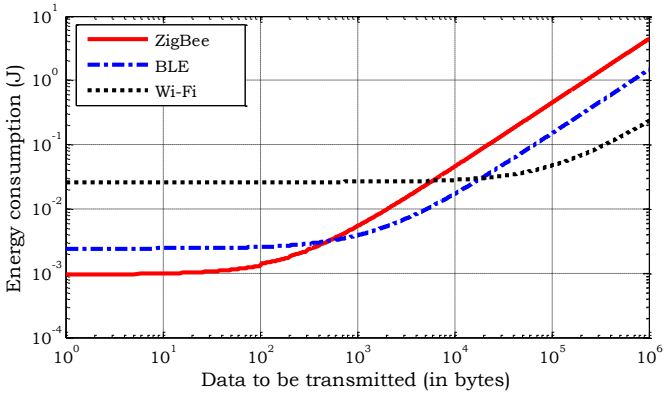

Figure 3.6 Energy consumption of the ZigBee, BLE and Wi-Fi in relation to different data loads

more than 100 kB. In the case of image-based particle characterization application, two to three frames are often sufficient to generate a background image and to analyze the particles, including the moving air bubbles. Therefore, with each frame of 256 kB, the amount of data to be transmitted is more than 750 kB.

For quantitative comparison, the energy consumption associated with in-sensor processing and raw data transmission using ZigBee, BLE or Wi-Fi is depicted in **Figure 3.7**. In this figure, the energy consumption of 9.4 mJ for the processing of a single vibration data set is extrapolated for up to six data sets. This enables a comparison to be made of in-sensor processing with raw data transmission for a data size of practical importance. In a similar manner, an energy consumption of 6.47 mJ [173], which corresponds to in-sensor processing of a single image frame, is used for the comparison. From this comparison, it can be observed that the in-sensor processing architectures for both the data and computation intensive vibration monitoring applications result in an energy efficient solution as compared to that for the raw data transmission.

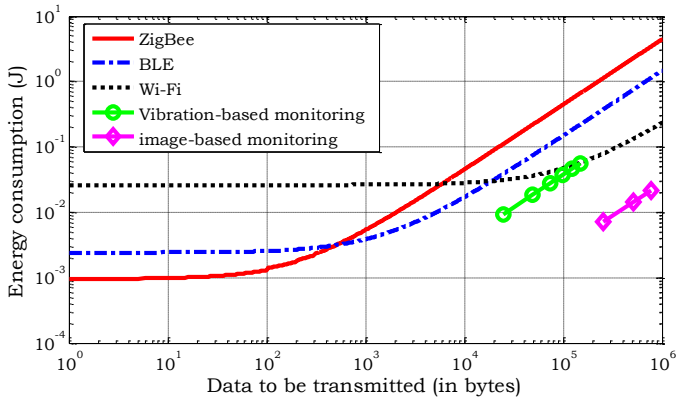

Figure 3.7 Energy consumption for the in-sensor processing in relation to that of raw data transmission using ZigBee, BLE and Wi-Fi

4

Feasibility Study of an SRAM-based FPGA for Duty-Cycle Applications

The architecture exploration and evaluation presented in chapter **3** suggest that by employing architectures that enable in-sensor processing, an energy efficient solution is achieved for data and computation intensive monitoring applications. Regarding the choice of the processing unit for the in-sensor processing, there are several alternatives, as shown in **Figure 4.1** that can be considered. Each of these has different pros and cons and therefore, should be analyzed in relation to energy efficient, cost-effective and generic wireless sensor node architecture.

4.1 Processing resources for in-sensor processing

Based on the design effort and associated cost, especially the non-recurring engineering (NRE) cost, the potential processing technologies for the in-sensor processing are divided into ASIC and non-ASIC categories, and are described in the following.

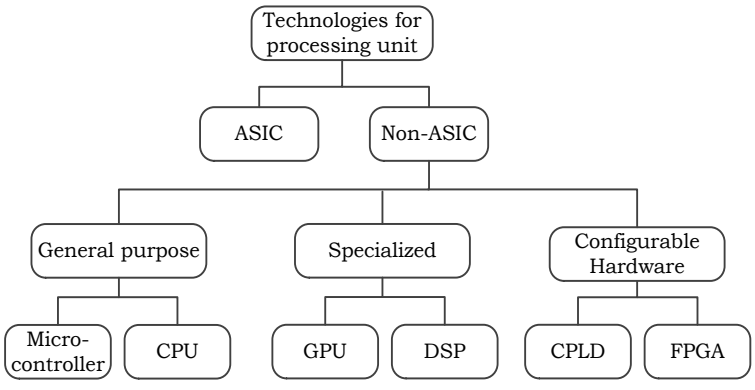

Figure 4.1 The different technological alternatives for s processing unit of a wireless sensor node

4.1.1 ASIC

In relation to other off-the shelf processing resources, an application specific integrated chip (ASIC) based processing unit can achieve the highest performance with minimum energy consumption. However, the design and development cost of an ASIC is highest in comparison to other alternatives, especially when considering a small to moderate volume production, which is most likely the case for an architecture targeting data and computation intensive applications. In addition, the time to market, despite the availability of several intellectual properties (IPs), is highest as it requires rigorous design, development and test efforts. In addition to high cost and long time to market, an ASIC solution, typically developed with design goals of a particular application, does not enable post development modifications so as to realize different applications. Therefore, in relation to a wireless sensor node that is aimed at a wide variety of data and computation intensive applications, as opposed to a particular application, ASIC does not appear to be a feasible solution.

4.1.2 Non-ASICs

Most low-end processors, such as micro-controllers that are typically used embedded applications are low-power and low-cost, however, their performance, typically measured in terms of operating frequency, is limited to a few MHz. In relation to an energy efficient wireless sensor node for data and computation intensive applications, the processing performance and the memory of such micro-controllers is not sufficient and therefore, can be excluded from the list of possible candidates for in-sensor processing.

General purpose processors, those used in computers, provide the means to design and develop applications cost effectively in high-level languages. Among these, the multi-core processors, which are commonly used in computers nowadays, enable multiple threads/processes to be executed in parallel to achieve high-performance. In addition, these also provide the means to adapt to different clock frequencies and voltage levels (a.k.a as dynamic voltage scaling, DVS) so as to optimize performance and power consumption. Despite the above mentioned power optimizing features, the power consumption for most of these processors varies from tens of watts to hundred of watts [175]. Integrating such a device in a wireless sensor node would require special current and heat related arrangements in order to use them at full capacity. Given the concerns regarding both the size and power consumption of a wireless sensor node, a general purpose processor based accelerator might not be an appropriate choice for a generic wireless sensor node architecture.

Both the graphic processing units (GPUs) and digital signal processors (DSPs) are programmable processors that are designed for particular application

domains. For example, the GPUs, as their name suggests, are designed to enable efficient processing of graphics related operations in computers. In a similar manner, the DSPs are optimized to design signal processing applications more efficiently. Therefore, incorporating either a GPU or DSP in a wireless sensor node targeted to a wide application area involving different algorithms to be processed, does not seem a promising option.

The configurable hardware such as complex programmable logic devices (CPLDs) and reconfigurable programmable gate arrays (FPGAs) are equipped with basic hardware logic gates that can be configured so as to realize the desired functionality in hardware. In addition to basic logic gates, most of these devices, in particular FPGAs, include memory, specialized hardware units such as multiply-and-accumulate (MAC) and phase-lock-loops (PLLs) that are often desirable in realizing complex and computation intensive algorithms to achieve desired performance metrics. In comparison to general purpose processors, GPUs and DSPs, the FPGAs enable each process of an algorithm to be realized to a hardware structure, i.e. sequential, pipelined and/or parallel hardware that results in high-performance and least energy consumption [176]. Comparing the design complexity in terms of the traditional method of specifying a design in register transfer level (RTL) levels, FPGAs can be considered highly complex in comparison with general purpose processor, GPUs and DSPs. The ASIC is however an exception, as it requires the highest design effort and takes the longest time to market. However, with the availability of wide range of IPs and design tools enabling abstraction level synthesis, the design time and cost of FPGA-based design has greatly improved. In addition to better performance and power consumption as compared to CPUs, GPUs and DSPs, and less design efforts than that of the ASICs, an FPGA-based accelerator, providing the means to be reconfigured to adapt to different applications and their requirements is the most promising solution in relation to a high-performance and low-energy wireless sensor node.

In relation to integrating an FPGA in an embedded system, it is the static random access memory (SRAM) based FPGAs that are commonly used [178]-[185]. This is mainly because these FPGAs not only offer embedded resources such as block RAM and Multiply-and-Accumulate (MAC) units in addition to basic logic cells, but they are typically built on a relatively more advance semiconductor technology as compared to their re-configurable counterparts such as flash-based FPGAs and therefore, provide a better performance [186]. However, their relatively high static power consumption, as compared to the flash-based FPGAs, is often considered as a limiting their potential advantages in battery-powered embedded systems, in particular in wireless sensor nodes. In addition, dynamic power management and duty-cycling techniques [187]-[188] that can be applied to conserve static power when an FPGA is idle are typically limited by the energy consumption associated with the resulting re-configuration process. It is the re-

configuration process and the associated energy consumption of SRAM-based FPGAs that most researchers use as supporting augments for excluding these FPGAs for wireless sensor nodes, in particular for those that are expected to operate in a duty-cycle manner. In the remainder of the chapter, the feasibility of an SRAM-based FPGA is analyzed in relation to duty-cycle applications.

In a duty-cycle mode of operation, a wireless sensor node periodically performs the required task(s) for a short period of time, τ, in relation to the total time period, T. During time period (T-τ), which is called inactive duration in this thesis, all integrated modules can be switched to appropriate low-power mode. In relation to an wireless sensor node that integrates an FPGA, the effective energy conservation that can be achieved by means of dynamic power management and duty-cycling depends on the energy consumption during sleep, power-off, sleep-to-active, and power-off-to-active states and, is typically dictated by a number of factors related to an FPGA in use and are discussed in the next section.

4.2 FPGAs for duty-cycle applications

In an FPGA-based wireless sensor node that is targeted to achieve high-performance, re-configurability, low-cost, compact size, and low-power consumption, the choice of an FPGA plays a crucial role. The factors that influence the choice include the size of an FPGA in terms of logic resources, the underlying technology, availability of low-power states and, support for easy and fast development of hardware and software.

In terms of logic resources, modern FPGAs typically consist of logic cells/elements (i.e. look-up tables, flip-flops, etc.), memory buffers, and embedded logic units such as multipliers, circuitry for synthesizing different clock frequencies, etc. As the number of logic resources in an FPGA has an impact on cost, physical size and power consumption, it is important to choose an FPGA with the right amount of resources so that, for a given set of applications, an integrated FPGA results in both an optimal logic resource utilization and power consumption. For an FPGA-based generic wireless sensor node targeting computation intensive applications, the actual type and amount of logic resources required to synthesize are expected to vary with underlying applications. However, to provide an indication to the readers, a list of major resources and their utilization regarding potential applications for this architecture as found in published literature [178]-[179] and [182]-[185] is complied in **Table 4.1**.

In addition to logic resources, the underlying technology of an FPGA also affects the performance, power-consumption, re-configurability, cost etc. and therefore, requires equal consideration in the selection of an FPGA. In relation to technology, modern FPGAs can be classified into three major categories, anti-fuse, flash, and SRAM-based FPGAs, and different pros and cons are associated with

each category. Unlike anti-fuse FPGAs, both the flash and the SRAM-based FPGAs can be configured multiple times and thus, can be integrated in a generic embedded system in order to realize different applications. Therefore, we restrict our discussion to only these two categories. The major difference in these two categorizes is the manner in which the configuration data is stored in the device. In

flash-based FPGAs, the configuration data is stored in flash memory cells and is retained even when power to the FPGA is turned-off. On the other hand, the configuration data in SRAM-based FPGAs is stored in SRAM cells and is lost when the power is turned-off. Therefore, a SRAM-based FPGA is required to be re-configured each time the power is turned-on. In relation to dynamic power management and duty-cycling in which the power supply to an FPGA is turned-off so as to conserve energy, a flash-based FPGA is likely to result in a shorter power-off-to-active transition time and the associated energy consumption. However, typical SRAM-based FPGAs not only offer additional embedded resources such as block RAM, and MAC units, but are built on relatively more advanced semiconductor process technology as compared to their contemporary flash-based counterparts, and thus provide a better performance [186].

In order to choose a reconfigurable, low-cost, and low-power FPGA that consists of ample amount of resources as described in **Table 4.1**, there are a number of FPGAs from different vendors that can be considered. However, at present, both in terms of volume and revenue, the Xilinx and the Actel are the leading manufacturers of the SRAM and flash-based FPGAs, respectively and therefore, provided the motivation to study the FPGAs from these two manufacturers. The Spartan-6 is a low-power and low-cost family of Xilinx's SRAM-based FPGAs, whereas the IGLOO family can be associated with the same attributes among the Actel's flash-based FPGAs. Among the Spartan-6 FPGAs, it is the XC6SLX16 that has the desired amount of resources as given in **Table 4.1** and therefore, it is used to investigate the energy consumption in relation to duty-cycling applications. Among the Actel's IGLOO family, it is the AGL1000V2 that provides a similar amount of resources as compared to that of the XC6SLX16 and therefore, we find it interesting to investigate its energy consumption so as to compare the SRAM-based FPGA with the flash-based FPGA.

Table 4.1 Type of logic resources and their utilization in realizing certain monitoring applications on FPGAs

Resource Type	Four –Input Look-up Tables	Flip-Flops	Dedicated RAM (kb)	Multipliers (18x18 bits)
Number of resources used	7000 to 11000	2800 to 6900	24 to 300	4 to 35

4.3 Operating conditions and measurement procedure used for the feasibility study

In order to study the feasibility of the SRAM-based FPGA for duty-cycle applications, the experiments are conducted using real hardware. The hardware system comprises of the FPGA, a micro-controller and a flash memory. The FPGA is enabled to be completely powered-off, in addition to be switched to its built-in sleep state. As shown in **Figure 4.2**, in order to monitor the state of the FPGA and assert the necessary control signal, a micro-controller is used. The built-in sleep state of the FPGA, which is called the 'suspend state' in relation to this FPGA, is activated by de-asserting the SUSPEND input signal of the FPGA. When the FPGA is in the suspend state, the micro-controller de-asserts this signal to switch it back to the active state. The micro-controller monitors the AWAKE signal, which is asserted by the FPGA when it is ready to perform the desired operations. In order to switch the FPGA to the power-off state, the power-supplies to the FPGA that are VCCINT (1.2V), VCCAUX (3.3V) and VCCIO (3.3V) are turned-off. It was only the FPGA's core that was powered through the 1.2 V voltage regulator, therefore, in order to turn-off the 1.2 V supply to the FPGA, the regulator was disabled. On the other hand, the 3.3 V is supplied to the micro-controller in addition to FPGA, therefore, P-type power MOSFET transistors, as shown in **Figure 4.2**, are used to turn-off VCCAUX and VCCIO of the FPGA.

The typical current drawn by the P-type transistor is 1 μA, and can typically be turned-on and off in 1 μs and 3 μs, respectively. It should be noted that **Figure 4.2** is included to show a simplified implementation view related to the suspend and power-off state and therefore, does not include pull-up/pull-down

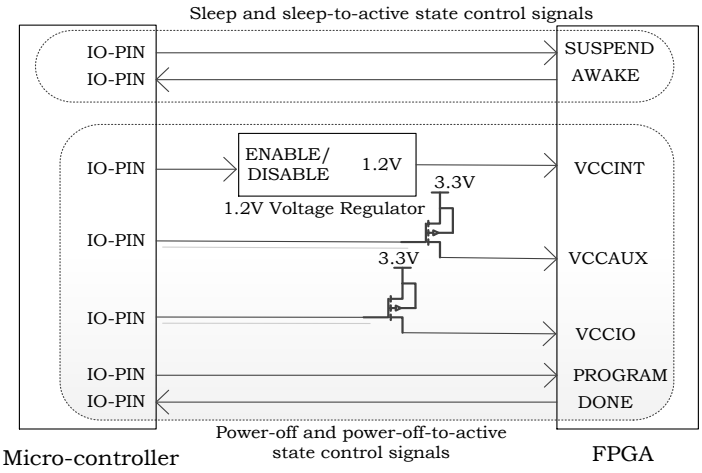

Figure 4.2 A simplified depiction of implementation details related to the suspend and power-off state

circuits etc. that are required to provide a stable operation. In order to switch the FPGA back to the active state from power-off state, the micro-controller enables the 1.2 V voltage regulator, turns-on the MOSFET and then asserts the PROGRAM signal such that the FPGA can load the configuration data from the associated flash memory. Then FPGA then loads the configuration from the associated flash memory. When the configuration is completed and the FPGA is ready to operate, it asserts the DONE signal.

The timing and power consumption parameters that are presented in this paper, in relation to the SRAM-based FPGA, are all measured from real hardware design. The power consumption is measured by recording the current drawn through a 3.6 V main power source for the experimental hardware. The parameters associated with the flash-based FPGA, AGL1000V2 are extracted from the manufacturer's specifications and the software tools (LiberoIDE) provided by the manufacturer. The power consumption for the AGL1000V2 presented in the results corresponds to the power supply voltage of 1.2 V and 3.3 V for the core and IOs, respectively.

4.4 Energy efficiency of SRAM-based FPGA for duty-cycle applications

The typical static power consumption of the SRAM FPGA, XC6SLX16 is 24 mW. In order to conserve static power during idle state, it can be switched to the suspend state. During the suspend state, the FPGA maintains all the design states and the configurations data while reducing the static power consumption to a lower level. The average power consumption measured during the suspend state is about 11 mW, which corresponds to more than a 50% reduction as compared to that of the idle state. The important feature of this state is that the FPGA can be switched to the suspend state in less than 14 ns, and back to active state in less than 20 µs. With this short transition time to and from the suspend state, it is possible to conserve energy by frequently switching the FPGA to the suspend state.

Based on this fast transition time and a more than 50% reduction in power consumption, the suspend state can be an ideal option to conserve energy when the FPGA is idle for short durations. However, for longer durations that are typically associated with duty-cycle applications, the power consumption of 11 mW can lead to the wasting of a valuable amount of energy. For example, if the FPGA is switched to the suspend state for 5 minutes, the resulting energy consumption is 3.6 joules, which can otherwise be used to perform certain other functions. Therefore, when the FPGA is idle for long durations it is better to switch it to the power-off state so as to minimize the energy consumption. However, by switching the FPGA to power-off state, all the design states and the configuration data are lost. As a consequence, on the next power-up, the FPGA requires to be re-

configured by loading the configuration data (bit stream) from the associated non-volatile memory.

The SRAM FPGA is configured through a serial peripheral interface (SPI) interface with a selectable data bus width of single, dual and quad-bits and a maximum transfer rate of 66 MHz. In order to achieve a fast configuration time for the FPGA, a non-volatile memory that provides quad SPI interface and an operating frequency of upto 85 MHz is used. In an uncompressed format, a configuration time of 15.16 ms is recorded for loading the bit stream of 3,731,264 bits from the associated non-volatile memory to the FPGA. During this configuration process, the SPI bus width and configuration speed were set to quad-mode and 66 MHz, respectively. In order to further minimize the configuration time, the FPGA can also be configured using a compressed bit-stream. However, the size of such a bit-stream depends on the synthesized application and therefore, results in an application dependent configuration time. It should be noted that during the power-off-to-active state, an additional delay of about 8 ms, apart from the actual configuration time, was measured. This delay corresponds to the time at which the power to the FPGA is turned-on to the instant at which it becomes ready to load the bit stream and, thus, leads to the power-off-to-active time to 23.5 ms.

During the power-off-to-active state, the instantaneous current drawn from a 3.6 V power source is shown in **Figure 4.3**. It should be noted that during the configuration process, both the FPGA and the flash memory are ON and therefore, the current drawn corresponds to both of these modules. Upon power-up, the FPGA causes the instantaneous current to exceed a little over 1 A. The average current consumption during power-off-to-active state is measured to be 30.3 mA, which leads to the energy consumption of 2.56 mJ in relation to the power-off-to-active of the FPGA. This, in other words, means that the FPGA should only be switched to power-off state if the energy conserved is more than 2.56 mJ. The

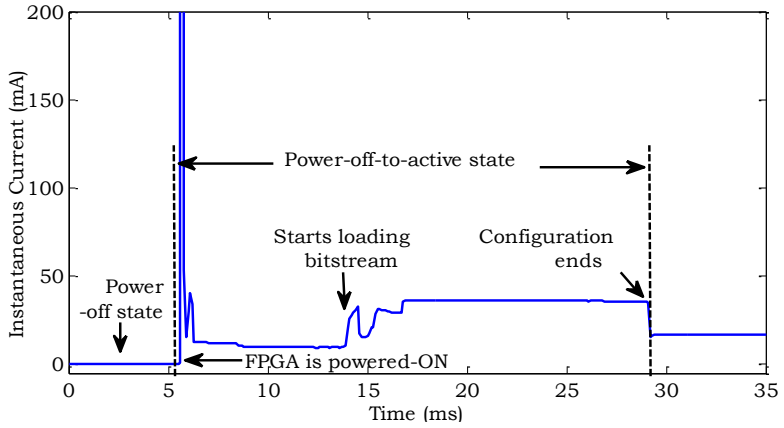

Figure 4.3 Instantaneous current consumption during power-off-to-active state of the FPGA

resulting duration for which the energy conservation is more than 2.56 mJ is 82.5 ms or more. The results relating to different states of the FPGA are also summarized in **Table 4.2**.

In order to compare the effective energy conservation that can be achieved by switching the FPGA to the suspend and power-off state in relation to that of idle state, the percentage energy conservation for a wide range of inactive duration is shown in **Figure 4.4**. For a given inactive duration, the energy conservation corresponds to the ratio of the energy consumption during the idle state and the suspend state or the power-off state. The energy consumption corresponding to the suspend and the power-off state also includes the energy consumed to switch the be switched to the suspend or power-off state. However, for an inactive duration of 38 µs to 235 ms, switching the FPGA to the suspend state results in more energy conservation as compared to the power-off state. This is mainly due to the short transition time to/from the suspend state. On the other hand, for an inactive duration longer than 235 ms, it is the power-off state that results in the maximum energy conservation.

Table 4.2 Average power consumption, switching time, and energy consumption during different states of the SRAM-based FPGA

	Idle state	Suspended state	Power-off state	Power-off-to-active state
Average power cons.	24 mW	11 mW	0	109 mW
Time required to switch the FPGA to	-	14 ns	< 10 ns	23.5 ms
Energy consumption	Time dependent	Time dependent	0	2.56 mJ

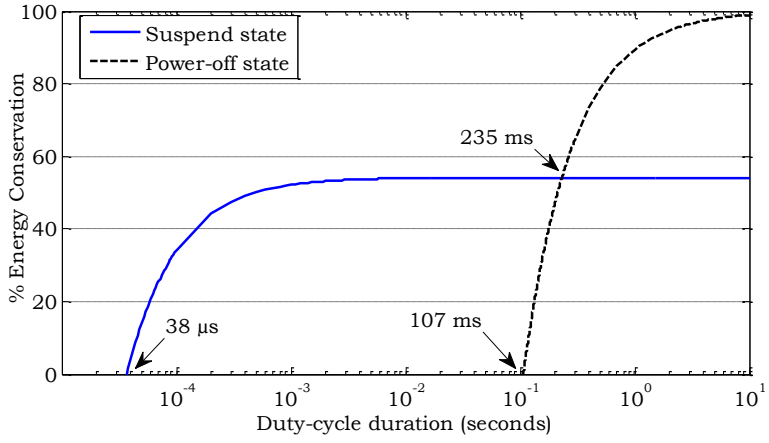

Figure 4.4 Percentage energy conservation during the suspend and power-off state in relation to idle mode

The typical static power consumption of the flash-based FPGA is 75 µW [189]. This FPGA can also be switched to a low-power state, known as the FlashFreeze state. The typical power consumption during this state is 50 µW and, the time required to enter and exit from the FlashFreeze state is about 1 µs. The energy consumption of both the flash and SRAM FPGAs for different inactive durations is shown in **Figure 4.5**. For a given inactive duration, the energy consumption of the flash FPGA corresponds to the energy consumed during the FlashFreeze state and during switching the FPGA to FlashFreeze state and back to active state. The energy consumption of the SRAM FPGA corresponds to that of the suspend state for inactive duration of up to 235 ms, and to the power-off state for durations longer than 235 ms. From **Figure 4.5**, we can observe that for an inactive duration of 100 µs to 10 ms, the difference in energy consumption between the two FPGAs is negligible. After 10 ms, the energy consumption associated with SRAM FPGA increases and leads to a maximum difference of 2.4 mJ at an inactive duration of 234 ms. However, with an inactive duration of 235 ms or more, the energy consumption of the SRAM-based FPGA remains almost constant whereas the energy consumption associated with the flash-based FPGA tends to increase. For an inactive duration of more than 54 seconds, the energy consumption of the flash-based FPGA exceeds that of the SRAM-based FPGA.

For the results shown in **Figure 4.5**, the flash-based FPGA was switched to the FlashFreeze state during which it consumed 50 µW and therefore, it can be argued that the power supply to the flash-based FPGA can also be turned-off so as to minimize the energy consumption associated with the flash-based FPGA. However, it should be noted that the time required for the power-off-to-active state and the associated the energy consumption is more than that of switching the

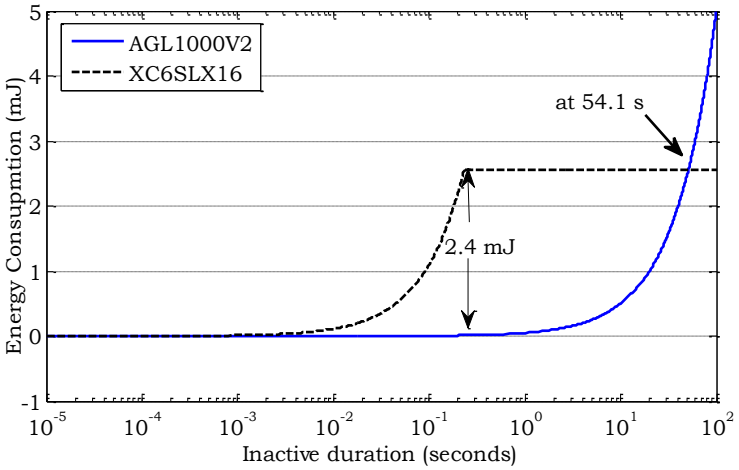

Figure 4.5 Energy consumption of XC6SLX16 and AGL1000V2 during low-power states for a range of inactive durations during which these FPGAs can be switched to their respective low-power states

flash-based FPGA from FlashFreeze to the active state. This means that by switching the flash-based FPGA to the power-off state, the maximum energy conserved in relation to the SRAM-based FPGA is likely to be less.

In relation to an equivalent flash FPGA, AGL1000V2 that consumes 50 µW during its low-power state, we observe that by switching the SRAM FPGA to the suspend and power-off states, its energy consumption is almost equivalent to that for an inactive duration less than 100 ms. On the other hand, for an inactive duration greater than 54 seconds, the energy consumption associated with SRAM FPGA is less than that of the flash FPGA. Therefore, it can be concluded that the SRAM FPGA that provides more logic resources and a higher operating frequency as compared to the flash FPGA, can be used to achieve high-throughput computation intensive processing locally in an wireless sensor node while maximizing the operational lifetime through dynamic power management and duty-cycling.

5

Design and Development of a High-performance Wireless Embedded Platform

Based on the results of evaluation and feasibility study of SRAM-based FPGA for wireless monitoring applications, a high-performance wireless embedded platform, the SENTIOF (shown in **Figure 5.1**) is designed to enables in-sensor processing for data and computation intensive wireless monitoring applications. The design is aimed at following key characteristic so as to enable energy efficient architecture.

High-Performance: In addition to sequential processing capabilities that are required to perform simple control, processing, and communication related tasks, there must be a support for hardware acceleration in order to fulfil the performance requirements of computation intensive high-sample rate monitoring applications.

Low-energy: The integrated components should be operable at different clock frequencies so as to optimize the energy consumption for a given task. In addition, the design should not only provide the means to manage the energy consumption dynamically by switching the modules to different power states but it should also

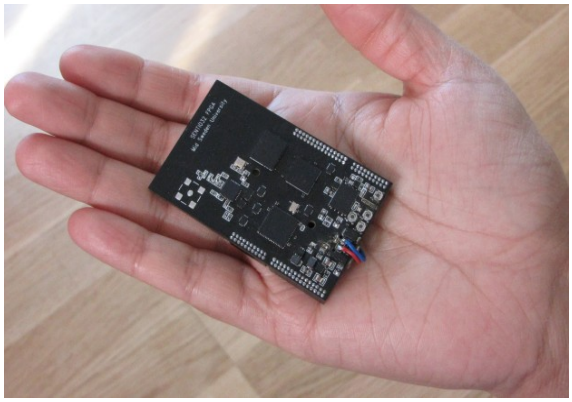

Figure 5.1 The SENTIOF platform

enable the power supplies of unused modules to be switched off whenever possible.

Sensor integration: It should support easy integration of application dependent sensors in saving the costs by re-using the hardware and associated software intellectual properties (IPs) among different applications. Therefore, in addition to providing different voltages so as to fulfil the requirements of different sensors, a large number of input-output interfaces from the data acquisition and processing module should be provided.

Flexibility: The sensor node should enable easy integration of sensors either with a basic processing unit and/or accelerator. In addition, the sensor node should also be enabled to allow application dependent communication between different modules, in particular between basic processor and accelerator.

Small size: The compact size often serves as a major characteristic for deploying such a sensor node in real world applications. For a wireless sensor node that can be used for different applications with diverse mounting and space related constraints, it is highly desirable to have small dimensions.

5.1 Architecture of the SENTIOF

A simplified architecture of the SENTIOF, representing all major components and their interconnections is shown in **Figure 5.2**. In order to achieve the above mentioned characteristics, different alternatives, especially communication, processing and memories and their possible interconnections are analyzed, as discussed below, in relation to data and computation intensive applications.

5.1.1 Communication transceiver

Among several standard based communication technologies that are analyzed in chapter **3** as possible candidates for integrating into a wireless sensor node, we observed that the infrared, mobile broadband and WiMax are best suited to certain application dependent and environmental conditions and therefore, could be excluded in the list targeting a general purpose wireless sensor node. The quantitative evaluation of other radio based short range communication technologies ZigBee, Wi-Fi and Bluetooth low-energy suggest that ZigBee is the most energy efficient for communicating small amounts (less than 500 bytes) of data, which is expected to be the case because of in-sensor processing. In addition, the larger coverage and higher scalability of this technology is important for a cost effective realization in relation to the desired applications. Therefore, an IEEE 802.15.4 compliant ZigBee transceiver, CC2520 [174], with an operational frequency band of 2.4 GHz - a licence free band worldwide, is integrated in the SENTIOF.

5.1.2 Processing unit

Regarding the choice of processing unit, we explored different options in chapter **4** so as to choose an appropriate processing unit for the energy efficient wireless sensor node architecture. Based on the findings of this study an SRAM-based FPGA, XC6SLX16 from Xilinx is chosen for the SENTIOF.

Based on the results and experiences from the evaluation study that, by using a micro-controller the control specific operation and non-intensive processing can be performed in an energy efficient manner, a micro-controller is also integrated in the SENTIOF. For this purpose, a 32-bit micro-controller AVR32UC3B is integrated as it provides a great deal of optimizing performance and power management, and can be used to perform non-intensive processing tasks. This feature is used in realizing vibration-based condition monitoring as described in chapter **6**.

5.1.3 Memories

In addition to the communication transceiver and a processing unit, the choice of the memory (with regard to type and size) is an important factor to achieve high-performance and low-power consumption. In particular, for a wireless sensor node that is expected to accumulate large amounts of raw data and partial results. To choose an appropriate memory type, there are two alternatives, dynamic random access memory (DRAM) and static random access memory (SRAM). A DRAM requiring regular refresh cycles so as to maintain its contents, consumes a higher power as compared to an equivalent sized SRAM. Therefore, in relation to energy efficient architecture, a SRAM is most suitable option. In addition to short term volatile memory, it is also important to integrate memory that retains its content if the power supply is switched off in order to conserve the energy. Therefore, a flash memory, in addition to a removable SD card is opted for the SENTIOF.

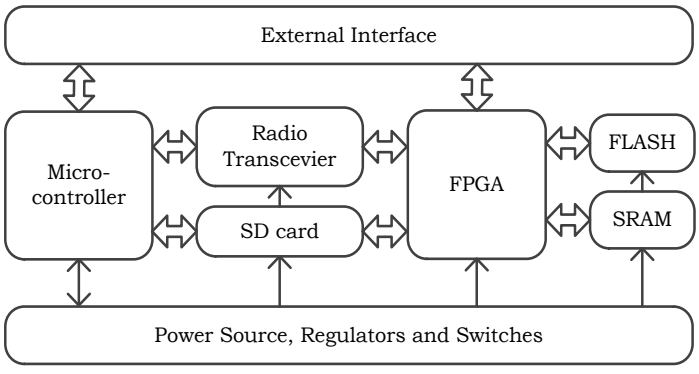

Figure 5.2 Architectural depiction of the SENTIOF

Based on above the discussion regarding the resources to be integrated, the target architecture is designed to include a micro-controller for sequential and control related tasks, an FPGA for computation intensive tasks, an IEEE 802.15.4 compliant radio transceiver for communication, a SRAM for short term data and result buffering, and a flash and removable SD card for long term storage. A simplified architecture of the SENTIOF, representing all major components and their interconnections is shown in **Figure 5.2**. With the exception of the power source, all other components are integrated in the compact sized SENTIOF. The details for which are provided in the following section.

5.2 Hardware design

In this section, the details regarding the integrated modules, PCB design, and features enabling energy efficient design are discussed.

5.2.1 Integrated modules

Regarding the specifications of each integrated module, a 32-bit micro-controller; ATUC3B512 is integrated based on the following reasons. Firstly, it provides a large memory (96 kB) that is useful for buffering acquired data before processing on the FPGA. Secondly, in addition to a wide range of operating frequencies, it provides a large number of low-power modes that are desirable in optimizing the power consumption for a given application. Lastly, it includes an extensive set of input-output (IO) interfaces that are important to connect application dependent sensors.

In order to enable wireless communication in the SENTIOF, an IEEE 802.15.4 compliant low-power radio transceiver, CC2520 is integrated in the design. The transceiver operates at the 2.4 GHz license free band and provides a maximum throughput of 250 kbps. In addition to six reconfigurable GPIOs for optional command and interrupt signals, the radio transceiver includes an SPI interface to communicate and exchange data either with the micro-controller or with the FPGA. The interface to both the micro-controller and the FPGA is achieved by means of multiplexer/de-multiplexer switches, which are controlled by the micro-controller and are therefore, dynamically configurable. This additional flexibility can be exploited to optimize the performance and the power-consumption of the communication activity according to the requirements of an application.

Table 5.1 Summary of Logic resources of the Spartan-6 FPGA XC6SLX16

Logic Cells	LUT size	Distributed RAM	Block RAM	MAC units	PLL	Number of IOs
14,579	6-input	136 kb	576 kb	32	2	106

Regarding external memory, low-power SRAM and flash memories are integrated in the SENTIOF. The 4 MB SRAM, CY62177DV30 provides a 16-bit wide data path that can either be used to perform 8 or 16 bits read and write operations in 55 ns. According to the manufacturer specifications, the SRAM typically draws 15 mA of current at its maximum operating frequency. It is worthwhile mentioning that the SENTIOF is designed to enable the expansion of the SRAM to 8 MB by replacing the above mentioned SRAM with the CY62187DV30. The 64 Mb of flash memory, W25Q64BV that is integrated in the SENTIOF can be accessed at a clock speed of 85 MHz through single, dual or quad serial peripheral interfaces (SPI).

Regarding the choice of FPGA to be integrated in the SENTIOF, there are several factors that are considered and are explained in detail in chapter **4**. By analyzing the logic resource requirements of typical computation intensive applications and the FPGAs' characteristics such as performance, re-configurability, cost, power-consumption, the underlying technology, availability of low-power states and, support for easy and fast development of hardware and software, a SRAM FPGA, Spartan 6 XC6SLX16 from the industry's leading manufacturer, the Xilinx is integrated in the SENTIOF. The selected FPGA is enriched with large amount of basic logic cells, in addition to embedded resources such as block RAM (Random Access Memory), multiply-and-accumulator (MAC) units, and PLLs. A summary of these resources is given in **Table 5.1**.

5.2.2 Power domains & dynamic power management

The SENTIOF is powered by means of a single DC power source, with an input voltage of 3.6 V to 6 V. This DC voltage is regulated and converted to four different power levels in the SENTIOF as shown in **Figure 5.3**. In relation to these four power domains, the 1.2 V regulated voltage is only used to power the FPGA′s core as it is the only device in the SENTIOF that requires 1.2 V to operate. The

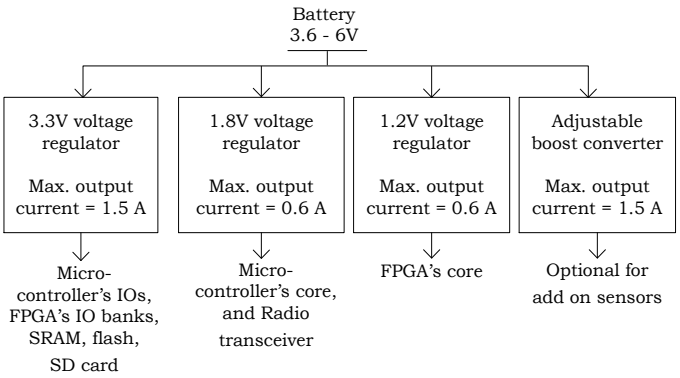

Figure 5.3 Power supply distribution in the SENTIOF

1.8 V is used to power the radio transceiver and the core of the micro-controller. Apart from the radio transceiver, the cores of the FPGA and the micro-controller, all other components integrated in the SENTIOF platform are powered by means of a 3.3 V supply. Therefore, the 3.3 V voltage regulator with a current rating of 1.5 A is used to ensure that sufficient current can be provided so as to operate all modules simultaneously at maximum operating speed.

As the majority of sensors used for monitoring applications are powered with 1.8 V to 3.3 V voltage supplies, both the 1.8 V and the 3.3 V power domains are provided on the external interface so that such sensors can be powered. In order to fulfil the power supply requirements of external devices/sensors that may require a higher voltage than the above mentioned, a boost convertor, the voltage for which can be adjusted up to 6.5 V, is integrated in the SENTIOF. In addition, different voltages and currents can be synthesized by using appropriate voltage regulators on the sensor layer, which is attached to the SENTIOF in order to realize an application. In addition to a DC power source, the SENTIOF can also be powered through a USB interface. This provides an added advantage of continuous long term power during an application development process.

In relation to dynamically managing the power consumption of integrated modules, all major components including the micro-controller, FPGA, SRAM, flash, and the radio transceiver can be switched to low-power modes at run time. This allows a reduction in the power consumption of each component to a minimum level, typically from tens of micro-watts to a few milli-watts depending upon the actual component. In relation to a battery operated wireless sensor node, the power consumption of a few milli-watts is undesirable. Therefore, to further minimize the power consumption during the idle state, the SENTIOF is designed to dynamically switch-off the power supplies to the FPGA, SRAM, flash, and SD card as they consume significant power in their low-power modes. To realize this power on/off mechanism, a low-power metal-oxide semiconductor (MOS) transistor is used as a switch between the power supply and the power connection of the components. The transistor is then switched on/off through the micro-controller, which performs all control specific operations including power management. It should be noted that the radio transceiver that typically consumes 1 µW in low-power mode, is not enabled to be switched on/off using the MOS transistor, as the MOS transistor consumes almost the same amount of power as of the radio transceiver in low-power mode and therefore, no significant power can be conserved by the power on/off method.

5.2.3 PCB design

In order to ensure the high-performance and low-production cost, special consideration was given to the Printed Circuit Board (PCB) design. This includes

the separation of different ground planes, the minimization of the trace length between the high-speed devices, limiting the design to a reduced number of layers, and avoiding micro and buried vias.

To minimize noise in relation to all the different types of components, four ground planes are used. A generic ground plane, GND, is used to connect the ground plane of the DC power source to other planes through 0 Ohm resistors. In these planes, the digital ground, DGND serves as a return path for all digital components including the FPGA, micro-controller, SRAM etc. The power ground, PGND is used to minimize the noise from other planes onto the power supply components. As there are no analog components mounted on the SENTIOF, the analog ground AGND is not used. However, it is provided to ensure that analog sensors can be reliably interfaced with the SENTIOF.

In relation to all the components mounted on the SENTIOF, it is the FPGA that has the highest pin count of 196 pins, which are packed into an 8x8 mm BGA package organized in 14 x 14 rows and columns. The 0.5 mm horizontal and vertical pin pitch resulting from this small footprint was a challenging factor, as it determined the routing and clearance rules in addition to the number of the routing layers of the SETNTIOF. For example, if the manufacturer′s guidelines regarding the PCB design of the FPGA [177] are strictly followed, then it requires a minimum of 7 PCB layers using micro-vias, while restricting the trace width and clearance to 0.075 mm. The production cost of the resulting PCB is then significantly higher in comparison to an equivalent PCB with a via diameter of more than 0.15 mm, and a trace width and clearance of 0.1 mm or more. Therefore, to relax the production requirements, the via-in-pad option was used for the FPGA. This allowed a larger pad size to be used and, eventually, a larger via hole diameter of 0.2 mm, a minimum trace width and clearance of 0.1 mm. In addition, it not only helped to reduce the PCB design to 6 layers but without the necessity of buried vias and thus, enabled the PCB cost to be reduced.

5.2.4 Physical structure

The size of the PCB is 65 x 40 mm, with 5 mm of interface height on one side that is used to attach a sensor module. A sensor module comprising of one or more sensors, is electrically connected with the SENTIOF platform through a rigid and strong header interface. In addition, two mounting holes are created in order to provide structural reinforcement of the platform and the sensor module, which may be required for some applications.

5.3 Software design

The software development for the micro-controller and the FPGA is carried out using the AVR32 Studio and the Xilinx ISE, respectively.

The integrated development environment for the micro-controller can be obtained from Atmel's website free of charge. It integrates a software framework and a GNU tool chain allowing easy and rapid application development in C/C++. To further enhance the software development process of the micro-controller in relation to the SENTIOF, application programming interfaces (APIs) are developed for all control and data transfer specific operations including power management, radio communication and reading/writing data to the SD card.

The Xilinx ISE Design Suite integrates all the tools to support complete design development, starting from an RTL or schematic design specification to the generation of a programming file for the FPGA. It also includes a wide variety of IP cores that can be integrated into a design, thus resulting in rapid development. In relation to the SENTIOF, we have developed interface APIs for both the SRAM and the flash, which can be re-used in other designs. Similar APIs for the radio transceiver and the SD card interface will also be developed in the future.

5.4 Performance and power consumption analysis

The performance of both the micro-controller and the FPGA can best be analyzed in relation to a given application and is done so by the realizing vibration and image-based monitoring applications, discussed in chapter 6 and 7, respectively. In this section, a measure of clock frequency, on which these two processing resources can be operated, is discussed briefly. In a similar manner to the performance, the power consumption is also application dependent. However, power consumption relating to fixed operations, such as during sleep mode and FPGA's configuration etc. is also presented in this section.

5.4.1 Clock frequency

The micro-controller can either be clocked from an internal oscillator producing a clock frequency of about 115 kHz or from an external oscillator producing 16 MHz of clock frequency. To achieve different performance levels, the micro-controller can be operated at a wide frequency range, with an upper limit of 60 MHz. In such cases, the desired frequency can be synthesized from a 16 MHz external oscillator by using a built-in Phase Locked Loop (PLL) in the micro-controller.

During the SENTIOF design it was observed that the available oscillators with frequencies higher than 19 MHz, as is required for the FPGA, consume a significant amount of power that is undesirable in a low-power platform. Therefore, a global clock generation feature of the micro-controller was instead used to provide the clock for the FPGA. The desired clock frequency in the FPGA, which should be within the maximum allowable operating frequency of up to 375 MHz can be synthesized using a PLL.

5.4.2 FPGA configuration

The minimum configuration time of 15.16 ms is recorded when an uncompressed bit-stream of nearly 3.6 Mb is loaded into the FPGA from the associated flash memory at the maximum allowable clock frequency of 66 MHz and the bus width option of 4 bits, also known as quad SPI. It should also be noted that, an additional delay of about 8 ms is recorded from the time the power to FPGA is switched on and the time to which it is ready to start the configuration. The detailed discussion regarding configuration procedure, time, and power consumption is analyzed in the next chapter.

5.4.3 Power consumption

5.4.3.1 Sleep mode

In the sleep mode, with the exception of the micro-controller, other components including the FPGA, SRAM, flash memory, and the SD card are switched-off. The micro-controller however, is switched to a low-power mode known as DeepStop, in which it is not only able to keep track of the sleep duration by using a real time counter (RTC), but is also capable of switching all components including itself to the active state.

In the sleep mode, the average current drawn by the SENTIOF from a 3.6 V supply source was measured to be 95 µA. It should be noted that during this mode, both the 1.8 V and 3.3 V voltage regulators which remained fully functional so as to ensure the required voltage levels to the micro-controller were responsible for nearly 70 % of the reported current consumption.

5.4.3.2 Active mode

Unlike the sleep mode, the power consumption during the active mode is dependent on the modules that are active and their operating frequencies. Nevertheless, to provide a rough idea, the current consumption is measured for a number of application scenarios involving almost all the major components on the SENTIOF, and is summarized in **Table 5.2**. Each reported value represents the average current that is calculated from the measured instantaneous current, which was drawn by the SENTIOF for a corresponding scenario, from a 3.6 V power source.

Table 5.2 The average current consumption of the SENTIOF for different application scenarios

S. No.	Tasks	Average current (mA)
1.	The micro-controller is clocked at 16 MHz and is in active mode, where it performs an addition operation repeatedly.	5.54
2.	Same as of 1, except the clock frequency is 20 MHz.	7.49
3.	Same as of 1, except the clock frequency is 60 MHz.	21.34
4.	The micro-controller is clocked at 16 MHz and is in Frozen mode, in which it generates 20 MHz of clock frequency for the FPGA.	4.28
5.	Same as of 4 + both the FPGA and the flash are ON, and the FPGA loads bit-stream from the flash memory at 66 MHz using quad SPI interface.	30.3
6.	Same as of 4 + FPGA is active and is running a design in which, a 100 MHz of clock is synthesized through internal PLL, and then is used to update a 27-bit counter.	22.31
7.	Same as of 6, except that the FPGA is in standby mode.	8.30
8.	Same as of 6 + the FPGA reads 16-bit data word from the SRAM repeatedly at a rate of 16.6 MHz.	34.58
9.	Same as of 6 + the FPGA writes 16-bit data word to the SRAM repeatedly at a rate of 16.6 MHz.	44.62
10.	Same as of 6 + the FPGA erases the flash memory.	48.17
11.	Same as of 6 + the FPGA reads data from the flash memory at a rate of 50 MHz.	29.90
12.	Same as of 6 + the FPGA writes data to the flash memory.	38.89
13.	Same as of 2 + the micro-controller reads data from the SD card at a rate of 20 MHz.	20.64
14.	Same as of 2 + the micro-controller writes data to the SD card at a rate of 20 MHz.	24.37
15.	Same as of 1 + the radio transceiver operates in receive mode and received packets are transferred to the micro-controller.	18.04
16.	Same as of 1 + the data packets from the micro-controller are transmitted by the radio transceiver at rate of 250 kpbs using 5 dBm power.	22.34

6

High-sample rate Wireless Vibration Monitoring

In the architecture exploration and evaluation study concerning data and computation intensive vibration and image-based condition monitoring applications, as discussed in chapter **3**, we observed that the in-sensor processing not only enables high-performance to be achieved but also results in an energy efficient solution. Following the results of this exploration study, we developed a compact sized wireless sensor node, the SENTIOF that provides the means to achieve high-performance and to optimize the power consumption. Using the SENTIOF, both the vibration and image-based condition monitoring applications are realized and are discussed in this and the following chapter, respectively.

In relation to vibration-based condition monitoring two case studies are conducted. One being to ensure that the performance and power consumption estimation obtained in the architecture exploration study are achieved on the target architecture, i.e. the SENTIOF. As in the vibration-based monitoring, depending upon the machinery being monitored and the expected problem, the frequencies leading to fault diagnosis vary, and so does the sampling frequency. In the second case study, the performance and energy consumption of realizing vibration-based condition monitoring on the SENTIOF is analyzed in relation to a wide sampling frequency to analyze aforementioned issue.

6.1 Case study 1 - High-sample rate vibration-based condition monitoring

In this case study, the tri-axes vibration data is acquired at a high-sampling frequency of 50 kHz and processed in the SENTIOF. Based on this real implementation and measurement, the performance and energy consumption related results are reported.

6.1.1 Algorithm and operating conditions for the vibration-based monitoring application

In this experiment, the vibration data processing algorithm used in the evaluation study, as depicted in **Figure 3.2** and described in section **3.1.2.1,** is implemented on the SENTIOF. In relation to the implementation on the SENTIOF, the processing tasks of the algorithm and their data flow is depicted in **Figure 6.1**.

In this vibration data processing algorithm, the 16-bit vibration data for each of the three axes is acquired at a sampling rate of 50 kHz, as it was in the case of evaluation test case. The acquired samples are then stored in three buffers (one for each axis) in the micro-controller's internal memory. Once there are 4096 samples in a buffer, the micro-controller sends this raw vibration data to the FPGA though the SPI. In the FPGA, digital filtering, with the help of an FIR filter of similar specification to that of the evaluation study, is performed on the raw vibration data to remove noise and to select the desired signal for further processing. Following on from the filtering, hamming windowing is performed to minimize the spectral leakage during the FFT. Following on from that the power spectrum is computed using a radix-2 FFT algorithm. After this, the computed spectra are compared to an ideal spectrum values, and the resultant information varying in size from one byte for each axis (1BPA), 2049-bits for each axis (NbPA), or full spectra (FSPA) are communicated back to the micro-controller for wireless transmission.

The micro-controller is operated at a clock speed of 16 MHz, the FPGA is operated at a clock speed of 100 MHz and it is functional only for the period during which it receives raw data from the micro-controller, processes data and delivers the results to the micro-controller. The communication between the micro-controller and the FPGA is carried out using an SPI operated at 8 MHz. In order to transmit the result wirelessly, the radio transceiver is operated at a maximum transmission throughput of 250 kbps using an output power level of 5 dBm.

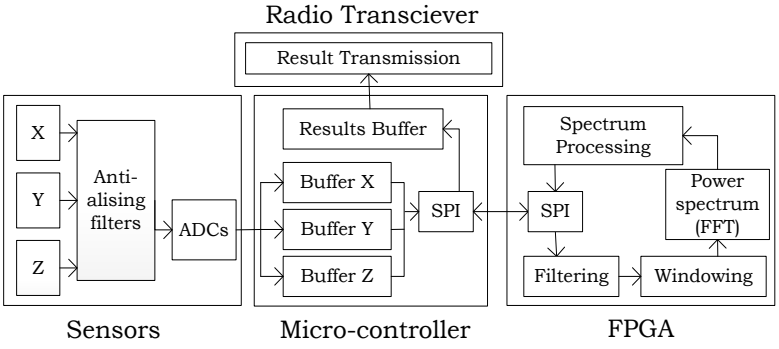

Figure 6.1 A simplified depiction of vibration data processing algorithm and data flow

6.1.2 Performance, energy consumption and lifetime of the vibration-based condition monitoring application

In order to achieve real time performance for a sampling rate of 50 kHz, the data transportation from the micro-controller to the FPGA, the data processing in the FPGA, the results transportation to the micro-controller, and transmission through radio transceiver should be completed before the acquisition of a new data set, that is less than 81.92 ms. The execution times measured for the aforementioned tasks in relation to the three different spectrum processing methods are given in **Table 6.1**. We observe that the execution time for both the 1BPA and NbPA is less than 81.92 ms, and therefore, by employing one of these methods, the high-performance goals of real time performance can be achieved using this analyzer. The execution times measured on the SENTIOF are in accordance with that of the estimated value for the evaluation study.

During the vibration data acquisition and processing, the micro-controller is active for the whole duration as it acquires vibration data, sends it to the FPGA and then transmits the results received from the FPGA using the radio transceiver. However, power supplies to the FPGA are switched-off dynamically to conserve power as soon as results are communicated to the micro-controller for transmission. In addition, the radio transceiver is also operational only during of the actual transmission, and is otherwise in switched to low-power mode. However, the micro-controller remains active all the time. The operational schedule with regards to 1BPA is shown in **Figure 6.2**. While a new data set, k+1, is being acquired, the previous data set is processed on the FPGA, and then FPGA is switched-off at 26.03 ms. It remain off until 58.42 ms when it is powered on to process the newly

Table 6.1 Execution time for tri-axes vibration data processing in relation to different spectrum processing methods

Spectrum processing methods	1BPA	NbPA	FSPA
Execution time (ms)	27.06	57.37	515.78

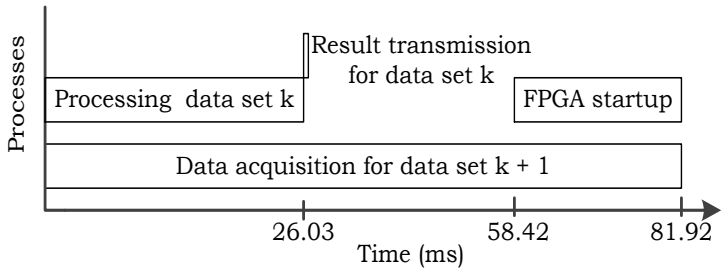

Figure 6.2 Process scheduling to realize the vibration processing algorithm for spectrum processing method 1BPA

acquired data. Note that turning on the FPGA at this time ensures that the FPGA is ready for the next computation despite the reconfiguration process.

Based on the average power consumption measured for the above mentioned vibration data processing algorithm, the lifetime is estimated. The lifetime estimation is performed in relation to a single cell (AA sized) battery that is used to power the SENTIOF during this experiment. The specifications of the battery are as follows: a capacity of 2600 mAh, chemistry of lithium-ion, a maximum discharge current of 3.5 A, and a voltage of 3.7 V[190].

In terms of continuous monitoring, the operational lifetime of about 80 hours can be achieved with the above mentioned single cell battery. As shown in **Figure 6.3,** in the case of duty-cycling practically viable operating lifetime can be achieved. For example, for a sleep period of 10 minutes and more, processing vibration data in the sensor node and transmitting a small amount of the resulting information leads to an operational lifetime of more than two years. Even with a monitoring interval of two hours, that could be considered sufficient for a large number of vibration-based condition monitoring scenarios, where the estimated lifetime reaches three years. It should be noted that in this lifetime estimation, the battery self discharge and other environmental condition are ignored for simplicity. Therefore, the lifetime obtained in a practical deployment is expected to slightly differ due to the above mentioned reasons.

6.2 Case study 2 - FPGA based in-sensor processing for vibration monitoring

In this case study, experiments involving a wide range of sampling rates and

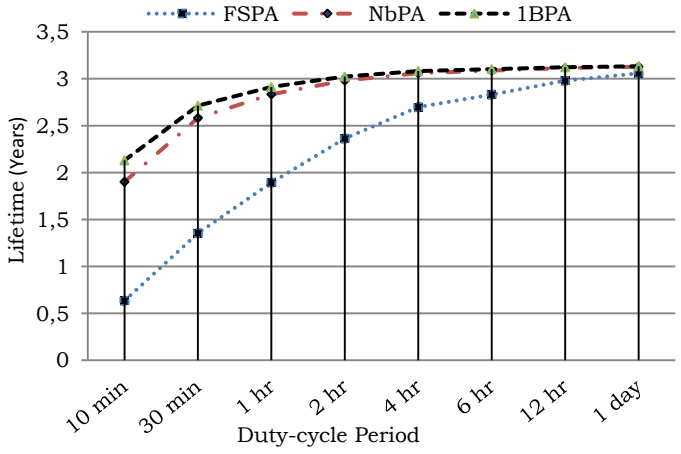

Figure 6.3 Lifetime of the wireless vibration analyzer for different duty-cycle durations

associated vibration data processing algorithms are realized on the SENTIOF and the performance and energy consumption is measured. In order to be able to compare the performance of the FPGA-based implementation in the SENTIOF to some reference architecture, two other sequential processor based architectures are realized.

6.2.1 Architectures and their evaluation criteria

6.2.1.1 Architectures

Architecture I: The majority of the low-power wireless systems used in monitoring applications are based on simple micro-controllers that typically operate at a clock frequency of a few MHz to 20 MHz. Therefore, it is interesting to investigate an FPGA based in-sensor processing in comparison with that of a micro-controller that operates within the above mentioned clock frequencies and, unlike the FPGA, processes data in a sequential manner. In this architecture, the data acquisition, processing and result transmission is performed with a micro-controller, operating at a clock frequency of 16 MHz.

Architecture II: With a sequential processor, such as a micro-controller that supports higher operating frequencies such as 50 MHz or above, higher processing performance can be achieved at the cost of increased power consumption. In relation to evaluating architectures on the basis of performance and energy consumption for a given sampling frequency, it can be valuable to include such an architecture for comparison. Therefore, in architecture II, the data acquisition, data processing, and result transmission is performed with a micro-controller operating at a clock frequency of 60 MHz.

Architecture III: In this architecture, an FPGA operating at a clock frequency of 100 MHz is used as a hardware accelerator to perform tri-axes vibration data processing. The data acquisition and result transmission is performed through a micro-controller operating at 16 MHz and the IEEE 802.15.4 compatible radio transceiver.

6.2.1.2 Scheduling

As both the processing performance and the energy consumption, for each of the above mentioned architectures, is highly dependent upon the way the processes are scheduled, and therefore, it is important to describe the corresponding scheduling.

Due to similar resources and the sequential nature of the processing in architectures I and II, their scheduling is identical as shown in **Figure 6.4(a)** In order to off-load the processor from the data acquisition process, direct memory access (DMA) is used to acquire data. This allows the acquisition of a new data set

$k + 1$ while processing and transmitting results for a previously acquired data set k. However, data processing and result transmission for a given data set is achieved in a sequential manner for each of the three axes. The scheduling for architecture III is shown in **Figure 6.4(b)**. In a similar manner to that for architectures I and II, the data acquisition and the remainder of the processing is performed in parallel. However, unlike architectures I and II, the actual vibration data processing for all three axes is also performed in parallel. The raw vibration data and results are communicated between the FPGA and the micro-controller through an SPI interface. In this architecture, the FPGA is powered-on immediately before the data transfer and remains powered-on until the results are transferred to the micro-controller. After that, it is powered-off to conserve energy.

In all three architectures, during the data acquisition and processing, the micro-controller and the radio transceiver are switched to low-power modes if the wakeup time allows them to satisfy the performance requirements for a given sampling frequency. It should also be noted that the relative execution time regarding each process shown in the **Figure 6.4** is arbitrary, as the actual execution time depends upon many factors such as sampling frequency, number of samples to be processed, architecture etc.

6.2.1.3 Sampling rates and evaluation criteria

The aforementioned architectures are realized and evaluated in relation to a wide range of sampling rates starting from 0.5 kHz to 200 kHz. The exact sampling rates and the corresponding number of samples processed as one data set are given in **Table 6.2**. For sampling rates of up to 8 kHz, the data set size is selected to match with the maximum expected frequency component ($f_s \geq 2f_{max}$) so as to achieve a resolution of 1 Hz/bin in the frequency spectrum. However, for sampling frequencies above 8 kHz the number of samples in a data set is fixed to 4096, as the number of logic resources required for bigger data sets exceeds the available resources in the SENTIOF's FPGA.

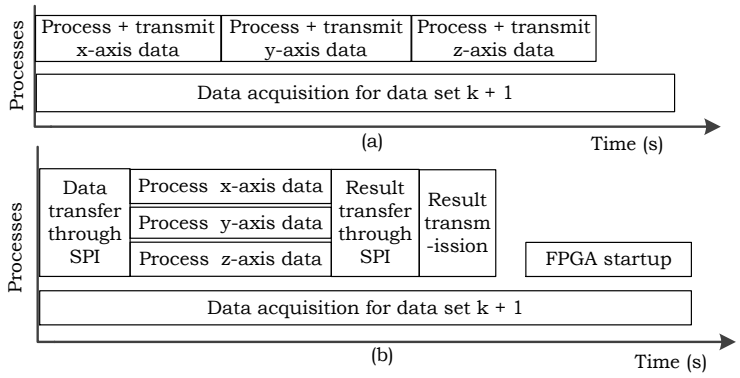

Figure 6.4 Process scheduling for (a) architecture I and II (b) architecture III

For a given sampling frequency, all three architectures are quantitatively evaluated in relation to real time processing performance (RTP) and energy consumption. The RTP is important in analyzing several consecutive data sets of vibration so as to ensure that no intermediate samples are missed in the analysis. In a similar manner, the energy consumption is important for a wireless sensor node operating on a fixed energy source, and mounted on non-stationary and hard to reach places.

Algorithm and experimental setup is the same as that of the case study 1, except the modification in the spectral processing methods. In this study, the NbPA is replaced with a No load (NL). In the case of NL, no resultant information is transmitted wirelessly when the currently computed spectra lie within the defined thresholds. It should be noted that different sampling frequencies would require different sensors for acquiring vibration data. And associated power consumption for these sensors would be different, therefore, the power consumption reported for these experiments does not include that of the sensors and associated circuits.

6.2.1.4 Quantitative evaluation of FPGA-based in-sensor processing

In relation to the spectrum processing method NL, the execution time required to process one set of vibration data on three architectures is shown in **Figure 6.5**. As the number of samples in a data set increases from 256 to 4096, so

Table 6.2 Sampling frequencies and corresponding data set sizes

Sampling rate (kHz)	0.5	1	2	4	8/16//32/50/ 100/150/200
No. of samples in a data set	256	512	1024	2048	4096

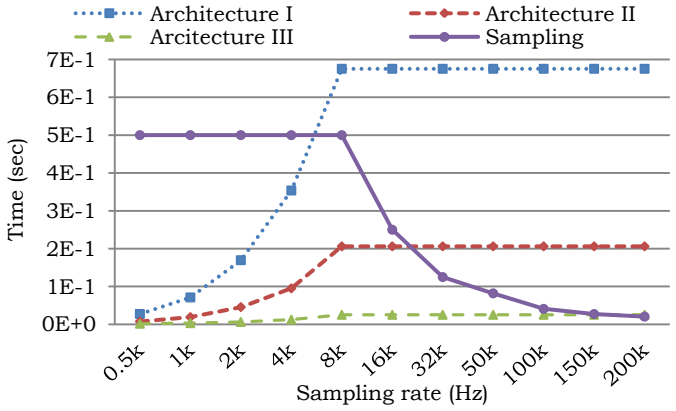

Figure 6.5 Time required for processing one data set using spectrum processing method NL for different sampling rates

does the execution time for each of the architectures. For both the architectures I and II, the percentage increase in the execution time is measured to be 150 % for sampling rates of 0.5 kHz to 1 kHz, and gradually decreases to around 100 % for sampling rates of 4 kHz to 8 kHz. For architecture III, the execution time increases at a constant rate of around 100 %. Analyzing the execution time in relation to the time required to sample one set of data (as shown **Figure 6.5**), it can be observed that the real-time processing performance for architectures I, II and III is achieved for sampling rates of up to 4 kHz, 16 kHz, and 150 kHz, respectively.

The energy consumed in processing one set of vibration data for a spectrum processing method of NL is shown in **Figure 6.6**. It should be noted that the reported energy consumption includes the energy consumed to acquire a full data set, in addition to the energy consumed in processing. In **Figure 6.6**, we observe that the energy consumption, for each architecture increases, as the number of samples to process is increased from 256 to 4096. The rate of this increase, is almost identical for both architectures I and II, however, the energy consumption of architecture II is always greater than that of architecture I, for a given sampling rate. For sampling rates of up to 8 kHz, this is because the higher processing speed of architecture II is least useful in relation to its power consumption, when performing simple tasks such as sampling the data. On the other hand, for sampling frequencies above 8 kHz, the gain in processing performance of architecture II, is measured to be less than the corresponding increase in the power consumption, as compared to those of architecture I. Therefore, the energy consumption of architecture I for sampling rates above 8 kHz is less than that of architecture II.

The architecture III not only achieves real time processing performance for sampling rates of up to 150 kHz, but also consumes the least amount of energy for sampling rates above 4 kHz. For a sampling rate of 4 kHz, though, the energy

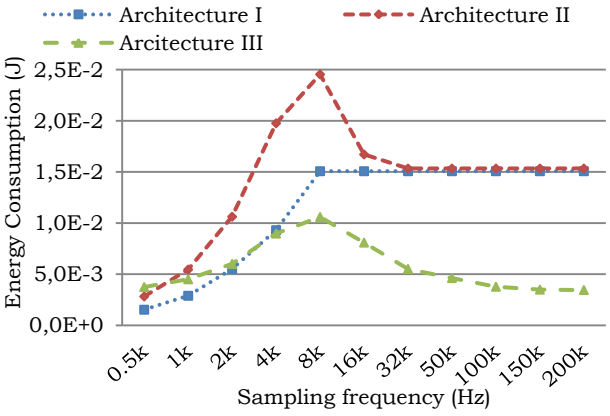

Figure 6.6 Energy consumed to process one data set using spectrum processing method NL for different sampling rates

Table 6.3 Summary of the results for spectrum processing NL

Sampling Frequency (kHz)	**0.5**		**1**		**2**		**4**		**8**		**16**		**32**		**50**		**100**		**150**		**200**	
Real-Time Processing	R	L	R	L	R	L	R	L	R	L	R	L	R	L	R	L	R	L	R	L	R	L
Performance (RPP)	P	E	P	E	P	E	P	E	P	E	P	E	P	E	P	E	P	E	P	E	P	E
/Least Energy Consuming (LEC)	P	C	P	C	P	C	P	C	P	C	P	C	P	C	P	C	P	C	P	C	P	C
Architecture I	x	x	x	x	x	x	x															
Architecture II	x		x		x		x		x		x											
Architecture III	x		x		x		x	x	x	x	x	x	x	x	x	x	x	x	x	x		x
Best architecture	**I**		**I**		**I**		**III**		**III**		**III**		**III**		**III**		**III**		**III**		**III**	

consumed by architecture III is slightly (3.5 %) less than that of architecture I, given the simple design and low-cost of architecture I, it may be preferred over architecture III for a sampling rate of 4 kHz. The results regarding the real time processing performance and energy consumption for spectrum processing NL are summarized in **Table 6.3**. For sampling rates of up to 4 kHz, architecture I stands ahead of the other architectures. On the other hand, for sampling rates of 8 kHz and above, architecture III achieves the highest performance with the least amount of energy consumption.

The results associated with the spectrum processing method, 1BPA are quite similar to those of the NL, and are therefore omitted here. In the case of FSPA, however, the real time performance is limited due to the large amounts of result transmission. The FPGA-based implementation, however, results in the least energy consumption for sampling rate for 4 kHz and more.

In summary, for sampling rates of 4 kHz and above, the SENTIOF, in which an FPGA is used to process vibration data, achieves the real time processing performance. Using such a wireless sensor node for high-sample rate vibration data monitoring and processing at regular interval, the operational lifetime of up to several years can be achieved without requiring the replacement of the battery.

7

Energy Efficient Realization of Image-based Wireless Monitoring

In this chapter, the implementation of the image-based monitoring application using the SENTIOF and the associated results are discussed. In the evaluation study regarding image-based in-sensor processing discussed in section **3.1.2.2**, it was observed that processing all of the required tasks in the sensor node and transmitting a small amount of resultant information wirelessly enables higher-performance and lower-energy consumption to be achieved when compared with alternatives involving different combinations of tasks processed on the sensor node. However, realizing all of the image-processing tasks in the sensor node, in particular an FPGA-based senor node leads to higher-design efforts and the least flexibility. Therefore, instead of processing all of the tasks in the sensor node, a balanced approach, as shown in **Figure 7.1**, is used for the in-sensor processing. With this method, the large amount of acquired data is reduced to a relatively small amount of information that is compressed and transmitted using a wireless transceiver. This method offers the advantage of realizing a generic architecture for machine vision applications while balancing the design complexity and energy consumption.

7.1 Architecture of the image sensor node

In order to monitor the presence of foreign particles in fluid, a generic architecture is realized using the SENTIOF. The complete architecture, including the tasks involved in processing the data and module on which these tasks are performed are shown in **Figure 7.2** and are explained in the following.

7.1.1 Image processing algorithm

The tasks involved in the applications are discussed in the following.

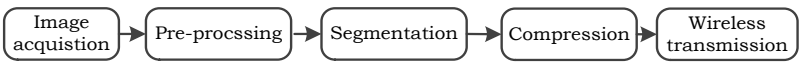

Figure 7.1 A generic architecture for wireless image-based monitoring

7.1.1.1 Data acquisition

The images processed and analyzed for detecting particles in the fluid are acquired using a CMOS sensor MT9V032 [191]. The image size of 640x400 pixels are then processed in the node in order to obtain a small amount of information containing the desired features before transmitting this information wirelessly for further processing at the remote station.

7.1.1.2 Pre-processing

During the pre-processing stage, the currently acquired image is subtracted from a background image so as to extract the potential regions with desired objects, i.e. particles. The background image that is captured at an earlier time than the currently acquired image is stored on the flash memory in the SENTIOF. Considering the amount of memory required, the complexity and communication cost involved in storing and retrieving a background image from a memory, a low-complexity background image model [192] is used in this study.

In order to store the background image, the 640x400 pixels size image, acquired through the above mentioned CMOS camera, is first scaled down by a factor of 8. It is important to mention that the scaled down version of the image can be kept in the internal memory of the FPGA. However, in the case of duty-cycling when the FPGA is powered-off, the background image is lost. Therefore, it is stored in the flash memory. During the pre-processing stage, this scaled down version of the background image is retrieved from the flash memory, and is up scaled using a near neighbour technique.

7.1.1.3 Segmentation and morphology

Threshold-based segmentation is performed to extract the mutually exclusive connected regions in the pre-processed image. Following on from segmentation,

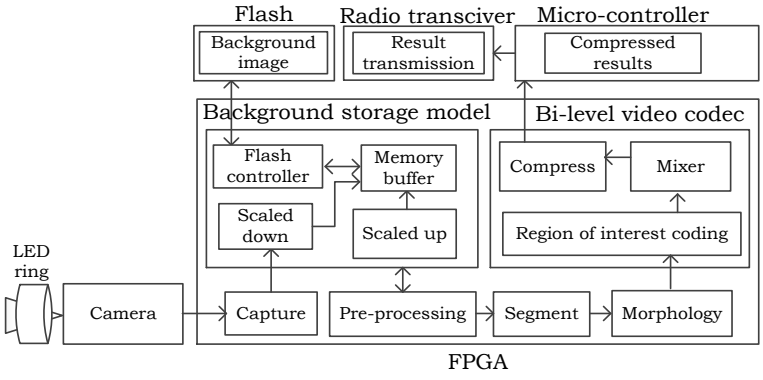

Figure 7.2 Architecture of the image-based wireless monitoring node

morphological operations are performed for removal of false objects (represented by one or two pixels). The morphological operations include the erosion followed by dilation.

7.1.1.4 Compression

After the segmentation and morphology, the gray scale image produced by the pre-processing stage is transferred in to a binary image. In order to reduce the amount of data to be transmitted, the binary image is compressed using bi-level coding technique [193] .

7.2 Operating conditions and measurement procedure for the image-based monitoring application

The above mentioned image processing algorithm was implemented on the SENTIOF and several experiments were performed to validate the functionality of the algorithm and to observe the effect of the ambient light on the results produced. In relation to energy consumption, the instantaneous current was measured using Agilent 34410A meter while operating the wireless node for the aforementioned algorithm. As shown in **Figure 7.2** the modules that were operational for realizing this image monitoring application include an LED ring, the camera, FPGA, micro-controller, radio transceiver, and the flash memory. During the operational state, the FPGA was operated at 27 MHz as it enabled to achieve desired performance while synchronizing the clock of FPGA with that of the camera. In order to conserve power, these modules were switched to their appropriate low-power mode whenever possible. In the case of duty-cycling, the power supplies to all of the modules are switched-off, except the micro-controller which was switched to its lowest power mode. The micro-controller in this power mode, DEEPSTOP, kept track of sleep duration through real time counter (RTC) and powered-on the other modules for the next operational cycle. In the case of the FPGA, upon power-up it is reconfigured by loading the bit-stream from the associated flash memory.

7.3 Performance, energy consumption and lifetime

Regarding resource utilization for synthesising the above mentioned tasks on the FPGA, 50% of slice LUTs, 8% of slice registers and 28% of the block RAM is used. In relation to the performance, for an image frame of 640x400 pixels, the sensor node architecture achieves real-time performance goals for up to 48 frames per seconds. For an average transmission data of 70 bytes, as obtained in the experiments, the energy consumed during the processing of one image frame is 17.2

mJ. It should be noted that the reported energy consumption includes the energy consumed during the FPGA's reconfiguration process. The reported energy consumption corresponding to the processing of one image frame can be divided in three major tasks, configuration, processing and communication. The energy consumption for each of these is given in **Table 7.1**.

In relation to the architecture exploration study for the image-based monitoring application, the performance and energy consumption of the above mentioned SENTIOF-based sensor node is given in **Table 7.2**. In addition, the operational life of both the exploration study and the above mentioned system is estimated for a wide range of duty-cycle intervals. The estimated lifetime, shown in **Figure 7.3** is based on a single cell battery of capacity 37.44 kJ. Furthermore, in this lifetime estimation, each time the node is activated from sleep mode it is operational for 100 ms so as to acquire images and to process them.

Based on the above mentioned performance results, it can be concluded that the image-based wireless monitoring realized on the SENTIOF delivers high-performance and also provides the operational lifetime of several years. For example, in the case of duty-cycle period of 5 minutes or more, the lifetime of more than three years can be obtained without requiring the replacement of battery.

Table 7.1 Execution time, average power and total energy consumption

Application	Config. time (ms)	Config. power (W)	Proc. time (ms)	Proc. power (W)	Comm. Time (ms)	Comm. power (W)	Total energy (mJ)
Particle detection	23.6	0.12	21	0.67	3.2	0.13	17.2

Table 7.2 Performance and energy consumption comparison with that arhicteure exploration study

Systems	Image size	E'Proc (mJ)	E'Comm (mJ)	Avg. Output data (bytes)	Bits/ pixel	Max. FPS
Evaluation study	(640x400)	9.6	3.52	500	0.016	48
Measured	(640x400)	13.9	0.44	70	0.002	48

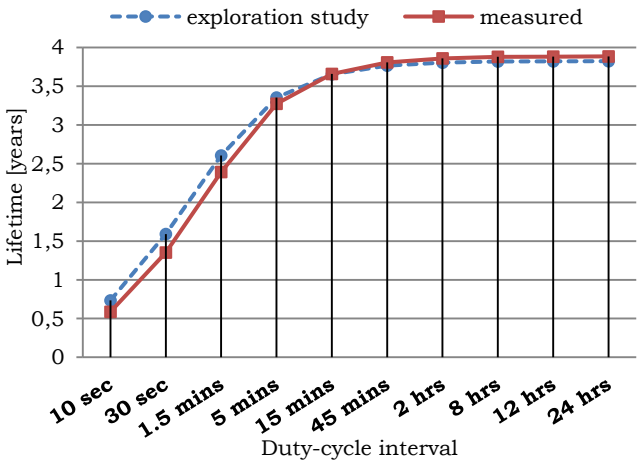

Figure 7.3 Operation lifetime of image-based wireless monitoring application

8

Conclusion and Future Work

In this chapter, the concluding remarks and the potential future work originating from this research work is presented. At the end, the authors contribution in relation to publications on which this thesis is based, are summarized.

8.1 Conclusion

Wireless sensor nodes are typically constrained by the physical size, cost and energy budget. These constraints, in conjunction with a desire for a long operational lifetime so as to reduce associated maintenance cost, typically restrict the sensor node design in different ways. One of which is to base the wireless sensor node design on low-power and low-cost modules, which in result provide limited communication and processing performance. On the other hand, monitoring applications that involve large amounts of data and complex processing algorithms for analysis of the data require higher communication rates and processing performance while demanding the long operational lifetimes of practical importance. By considering the limited communication, processing performance and energy budget of typical wireless sensor nodes, and high-communication and processing requirements of data and computation intensive applications, different architectural alternatives are explored and evaluated in an analytical and experimental manner.

In order to perform quantitative evaluation based on actual implementations, two case studies involving monitoring applications of high practical importance i.e. vibration and image-based industrial condition monitoring are performed. WSN based realization of these applications involves large amounts of raw data to be wirelessly transported across the network; for example, in the case of a tri-axis high-sample rate (e.g. 50 kHz) vibration monitoring, the amount of raw data is 2.4 Mbps. In a similar manner, a single grey scale image of 640*480 pixels resolution generates more than 2.4 Mb.

Based on communication and processing requirements of these case studies, the architecture exploration is carried out by evaluating different communication and processing technologies in relation to performance and energy consumption. From an architectural perspective, two approaches involving raw data transmission and in-sensor processing are explored and evaluated.

With regards to enabling a wireless sensor node for the above mentioned communication requirements, different standard based communication technologies including infrared, mobile broadband, WiMax, Wi-Fi, Bluetooth and ZigBee are analyzed mainly in relation to performance, energy consumption and cost. The infrared, mobile broadband and WiMax are observed to offer data rates that can fulfil the above mentioned communication requirements. However, there are several aspects that make them less suitable for generic and cost-effective wireless monitoring. In the case of infrared, it is the short range and strict alignment requirements that restrict its potential advantage of providing high-communication rates. In a similar manner, the relatively high initial and operating cost of both the WiMax and mobile broadband makes them least favourable. On the other hand, the low-data rates of Bluetooth and ZigBee technologies pose a limit in achieving continuous monitoring. The Wi-Fi is, however, observed to provide the desired communication rates for continuous monitoring at a modest cost and power consumption. With regards to operational lifetime of a Wi-Fi enabled wireless sensor node that has a limited energy budget, for example, as obtained with a typical AA sized dry cell battery, the detailed analysis suggested that a short term, i.e. few days of continuous monitoring can only be performed. In other words, in order to achieve a long operational lifetime by transmitting raw data, the data intensive monitoring applications are required to operate in an intermittent manner. In such a case, as low-data rate communication technologies such as BLE and ZigBee can also used, both of these including the Wi-Fi were then evaluated according to their energy consumption. Based on the energy consumption of the actual data transfer and that associated with establishing connection after sleep to wakeup state, and assuming maximum theoretical throughput, as attainable under ideal channel conditions the following results were obtained. The ZigBee consumes the least amount of energy for data loads of up to 500 bytes. On the other hand, the Wi-Fi appears to be best suited to data loads of 800 kB or more. For a wireless transmission load of less than 800 kB and more than 500 bytes, as is case for the data intensive monitoring applications, the results show that the BLE results in minimum energy consumption.

Regarding in-sensor processing for a vibration-based condition monitoring application that involves a series of computation intensive processing tasks such as FIR filtering, windowing, FFT, etc. three processing architectures involving a micro-controller, an FPGA, and a combination of micro-controller and FPGA were realized and first compared with each other. In this comparison, it was observed that not only the high processing performance for high-sample rate (i.e. 50 kHz)

tri-axis vibration data monitoring can be achieved with FPGA-based architectures but these also result in less energy consumption as compared to the micro-controller based in-sensor processing architecture. In addition, it was also experienced that the hybrid architecture consisting of both the micro-controller for sequential and control specific tasks and an FPGA for computation intensive processing delivers high performance with reduced design efforts in comparison to the FPGA based architecture. In relation to image-based in-senor processing, a study conducted to evaluate the performance and energy consumption associated with computing different image processing tasks and generating different amounts of resultant information suggested that processing the entire tasks in the sensor node enables achieving the high-performance while consuming the least amount of energy. It also, however, indicated that exclusion of classification and feature extraction tasks from the sensor node requires less design effort and also leads to a generic solution.

The comparison of raw data transmission using Wi-Fi, BLE or ZigBee, and in-sensor processing architectures showed that an in-sensor processing architecture consisting of an FPGA for computation intensive processing is a more energy efficient solution. With regards to choosing an appropriate FPGA technology, a literature study focusing the logic resources and performance aspects of data and computation intensive wireless monitoring applications provided the guidelines to consider SRAM FPGAs. As typical SRAM FPGAs require re-configuration and with regards to employing such an FPGA in a wireless sensor node that is operated in an intermittent manner so as to conserve energy, a feasibility study of SRAM FPGA for duty-cycle applications was carried out. In this study, focusing on Xilinx Spartan 6 LX16 FPGA that is comprised of a sufficient amount of logic resources that are typically required in data and computation intensive monitoring applications, energy consumption associated with sleep state and re-configuration processing was analyzed. From the measured results, it was found that for short intervals of 38 µs to 235 ms, up to 50% energy can be conserved by switching the FPGA to its built-in sleep state. For duration exceeding 235 ms, the FPGA can be completely switched-off to reduce the power consumption to zero. The resulting reconfiguration can be performed in 23.5 ms at the expanse of 2.56 mJ of energy consumption. In comparison to the energy consumption of the FPGA being in idle state for 10 seconds, even with the above mentioned energy consumption associated with re-configuration, the overall energy conservation of more than 99.9 % can be achieved.

Given the results of the evaluation studies that the FPGA-based in-sensor processing is a most an energy efficient solution for data and computation intensive applications, an FPGA-based wireless sensor node, the SENTIOF, is designed and developed. In addition to the FPGA, the SENTIOF is integrated with a micro-controller to perform control specific and sequential tasks. In terms of wireless communication, an IEEE 802.15.4 complaint radio transceiver is integrated as it

was observed to be the least energy consuming for small amounts of data transmission. For energy efficient design using SENTIOF, it is designed to enable dynamic power management, in which not only the integrated modules can be switched to their built-in low-power modes, but can also be switched-off to reduce the power consumption to the minimum level.

In order to validate the design and to assess the performance and energy consumption from real hardware, both the vibration and image-based monitoring algorithms are realized using the SENTIOF. In the case of the vibration-based condition monitoring, the feasibility of FPGA-based in-sensor processing is also characterized for a wide range of sampling frequencies, i.e. from 512 Hz to 200 kHz. The measured results show that for sampling rates of 4 kHz and above, FPGA-based in-sensor processing not only delivers required performance but also consumes least amount of energy. Based on the measured power consumption, the lifetime estimated using a single cell 3.6 V battery of capacity 2600 mAh, shows that the for a duty-cycle of 10 minutes or more, the vibration-based condition monitoring with in-sensor processing can be performed for more than two years without requiring the replacement of the battery. With regards to image-based monitoring application, the realized architecture, with a processing performance of 48 frames per second that is more than the required for this application, is measured to achieve the operational lifetime of more than 3 years for a duty-cycle periods of 5 minutes or more.

8.2 Future work

The possible extensions of this work are envisioned to branch into the following directions.

One aspect of the future work is to deploy a wireless sensor network for the above mentioned applications and observe the performance in the real world. In addition to real world deployment, the different variations in the application can be realized to assess the correlation of application requirements and real world performance. For example, in the case of image-based monitoring in which objects are classified on the basis of color, it would be interesting to determine the associated performance and energy consumption of in-sensor processing. In addition to varying the design requirements of the above mentioned applications, other data and computation intensive applications such as those involving audio and video data can also be explored.

Though, the operational lifetime for the above mentioned applications are estimated to be several years, it would still be interesting to explore suitable energy scavenging methods for such applications. For example, in the case of vibration-based condition monitoring, the possibility of scavenging energy from the vibrating machinery could be useful to extend the operational lifetime.

In relation to the SRAM FPGA, the re-configuration alternatives such as those involving bit-stream compression and partial/dynamic configurations are an interesting topic to be examined with regards to this FPGA.

8.3 Authors Contributions

The exact contributions of the authors of the seven central papers in this thesis are summarized in **Table 8.1**. The authors are listed in the same order as they are in the research papers.

Table 8.1 Author's Contributions

Paper #	Authors	Contributions
I	**KS**, PC, and BO	KS: Designed, developed and conducted experiments PC: Gave the idea and provided the help in software development BO: Supervised the work
II	**KS**, PC, and BO	KS: Starting with own idea, conducted all the necessary experiments PC: Provided the feedback in the article BO: Supervised the work
III	**KS** and BO	KS: Reviewed literature and conducted all the necessary experiments BO: Supervised the work
IV	**KS** and BO	KS: Reviewed literature and conducted all the necessary experiments BO: Supervised the work
V	**KS**, PC, and BO	KS: Design, developed and test the hardware platform for wireless senor node PC: Assisted in components selection and provided critical review of the design BO: Supervised the work
VI	**KS** and BO	KS: Performed all the necessary tasks including design, development, experiments and measurements BO: Supervised the work
VII	**KS** and BO	KS: Performed all the necessary tasks including design, development, experiments and measurements BO: Supervised the work
VIII	MI, **KS**, NA, MN, NL and BO	MI: Designed and developed image-processing tasks on FPGA, performed experiments and measurements KS: Design the hardware platform, developed necessary software for control and management and performed experiments and measurements NA: Provided review and feedback MN, NL and BO: Supervised the work

KS: Khurram Shahzad PC: Peng Cheng BO: Bengt Oelmann MI: Muhammad Imran NA: Naeem Ahmed MN: Mattias O'Nills NL: Najeem Lawal

References

[1] Jennifer Yick, Biswanath Mukherjee, Dipak Ghosal, "Wireless sensor network survey", Computer Networks, Volume 52, Issue 12, 22 August 2008, Pages 2292-2330, ISSN 1389-1286.

[2] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, "Wireless sensor networks: a survey" Computer Networks, 38 (4) (2002), pp. 393-422.

[3] Arampatzis, Th; Lygeros, J.; Manesis, S., "A Survey of Applications of Wireless Sensors and Wireless Sensor Networks," Intelligent Control, 2005. Proceedings of the 2005 IEEE International Symposium on, Mediterrean Conference on Control and Automation , vol., no., pp.719,724, 27-29 June 2005

[4] Kay-Soon Low; Win, W.N.N.; Meng-Joo Er, "Wireless Sensor Networks for Industrial Environments," Computational Intelligence for Modelling, Control and Automation, 2005 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, International Conference on , vol.2, no., pp.271,276, 28-30 Nov. 2005

[5] Isaac, S.J.; Hancke, G.P.; Madhoo, H.; Khatri, A., "A survey of wireless sensor network applications from a power utility's distribution perspective," AFRICON, 2011 , vol., no., pp.1,5, 13-15 Sept. 2011

[6] Durisic, M.P.; Tafa, Z.; Dimic, G.; Milutinovic, V., "A survey of military applications of wireless sensor networks," Embedded Computing (MECO), 2012 Mediterranean Conference on , vol., no., pp.196,199, 19-21 June 2012

[7] N. Xu, "A survey of sensor network applications," available: 2014 http://enl.usc.edu/~ningxu/papers/survey.pdf.

[8] Kavi K. Khedo, Rajiv Perseedoss and Avinash Mungur, "A Wireless Sensor Network Air Pollution Monitoring System", International Journal of Wireless & Mobile Networks 2.2 (2010) 31-45.

[9] Barrenetxea, G.; Ingelrest, F.; Schaefer, G.; Vetterli, M., "Wireless Sensor Networks for Environmental Monitoring: The SensorScope Experience," Communications, 2008 IEEE International Zurich Seminar on , vol., no., pp.98,101, 12-14 March 2008

[10] Corke, P.; Wark, T.; Jurdak, R.; Wen Hu; Valencia, P.; Moore, D., "Environmental Wireless Sensor Networks," Proceedings of the IEEE , vol.98, no.11, pp.1903,1917, Nov. 2010

[11] Lloret, J.; Garcia, M.; Bri, D.; Sendra, S. "A Wireless Sensor Network Deployment for Rural and Forest Fire Detection and Verification". Sensors 2009, 9, 8722-8747.

[12] Liyang Yu; Neng Wang; Xiaoqiao Meng, "Real-time forest fire detection with wireless sensor networks," Wireless Communications, Networking and Mobile Computing, 2005. Proceedings. 2005 International Conference on , vol.2, no., pp.1214,1217, 23-26 Sept. 2005

[13] Hefeeda, M.; Bagheri, M., "Wireless Sensor Networks for Early Detection of Forest Fires," Mobile Adhoc and Sensor Systems, 2007. MASS 2007. IEEE Internatonal Conference on , vol., no., pp.1,6, 8-11 Oct. 2007

[14] Lynch, J. P., and Koh, K. J. (2006). "A Summary Review of Wireless Sensors and Sensor Networks for Structural Health Monitoring", Shock and Vibration Digest, 38(2): 91-128.

[15] Sukun Kim; Pakzad, S.; Culler, D.; Demmel, J.; Fenves, G.; Glaser, S.; Turon, M., "Health Monitoring of Civil Infrastructures Using Wireless Sensor Networks," Information Processing in Sensor Networks, 2007. IPSN 2007. 6th International Symposium on , vol., no., pp.254,263, 25-27 April 2007

[16] Min Chen, Sergio Gonzalez, Athanasios Vasilakos, Huasong Cao, and Victor C. Leung. 2011. "Body Area Networks: A Survey". Mob. Netw. Appl. 16, 2 (April 2011), 171-193.

[17] Hande Alemdar, Cem Ersoy, "Wireless sensor networks for healthcare: A survey", Computer Networks, Volume 54, Issue 15, 28 October 2010, Pages 2688-2710

[18] Wright P, Dornfeld D, & Ota N. (2008) "Condition Monitoring in End-milling Using Wireless Sensor Networks (WSNs)". Transactions of NAMRI/SME. 36, 177-183.

[19] C. Emmanouilidis and P. Pistofidis, "Machinery self-awareness with wireless sensor networks: a means to sustainable operation," in Proceedings of the 2nd Workshop on Maintenance for Sustainable Manufacturing, pp. 43-50, Verona, Italy, May 2010.

[20] Tian He, Sudha Krishnamurthy, John A. Stankovic, Tarek Abdelzaher, Liqian Luo, Radu Stoleru, Ting Yan, Lin Gu, Jonathan Hui, and Bruce Krogh. 2004. "Energy-efficient surveillance system using wireless sensor networks" In Proceedings of the 2nd international conference on Mobile systems, applications, and services (MobiSys '04). ACM, New York, NY, USA, 270-283.

[21] Bokareva T, Hu W, Kanhere S, Ristic B, Gordon N, Bessell T, Rutten M, Jha S. Wireless Sensor Networks for Battlefield Surveillance. In Proceedings of the LWC—2006.

[22] A. Arora, P. Dutta, S. Bapat, V. Kulathumani, H. Zhang, V. Naik, V. Mittal, H. Cao, M. Demirbas, M. Gouda, Y. Choi, T. Herman, S. Kulkarni, U. Arumugam, M. Nesterenko, A. Vora, M. Miyashita, "A line in the sand: a wireless sensor network for target detection, classification, and tracking", Computer Networks, Volume 46, Issue 5, 5 December 2004, Pages 605-634, ISSN 1389-1286,

[23] Xiaoqing Yu, Pute Wu, Wenting Han, Zenglin Zhang, "A survey on wireless sensor network infrastructure for agriculture", Computer Standards & Interfaces, Volume 35, Issue 1, January 2013, Pages 59-64.

[24] Son, V.Q.; Wenning, B.; Timm-Giel, A.; Gorg, C., "A model of Wireless Sensor Networks using context-awareness in logistic applications," Intelligent Transport Systems Telecommunications,(ITST),2009 9th International Conference on, vol., no., pp.2,7, 20-22 Oct. 2009

[25] Pal, S.; Bhattacharyya, D.; Tomar, G.S.; Tai-hoon Kim, "Wireless Sensor Networks and Its Routing Protocols: A Comparative Study," Computational Intelligence and Communication Networks (CICN), 2010 International Conference on , vol., no., pp.314,319, 26-28 Nov. 2010

[26] C.F. Garca-Hernndez, P.H. Ibargengoytia-Gonzlez, J. Garca-Hernndez, J.A. Prez-Daz "Wireless sensor networks and applications: a survey" International Journal of Computer Science and Network Security, 7 (2007), pp. 264-273

[27] R. G. Luis, L. Lunadei, P. Barreiro, and I. Robla, "A Review of Wireless Sensor Technologies and Applications in Agriculture and Food Industry: State of the Art and Current Trends". IEEE Journal on Sensors, vol. 9, pp. 4728-4750,2009.

[28] T. Bokareva, W. Hu, S. Kanhere, B. Ristic, N. Gordon, T. Bessell, M. Rutten, S. Jha, "Wireless sensor networks for battlefield surveillance," in: Proceedings of The Land Warfare Conference, LWC Brisbane, Australia, October 24–27, 2006.

[29] Ian F. Akyildiz, Ismail H. Kasimoglu, "Wireless sensor and actor networks: research challenges", Ad Hoc Networks, Volume 2, Issue 4, October 2004, Pages 351-367,

[30] Paradiso, J.A.; Starner, T., "Energy scavenging for mobile and wireless electronics," Pervasive Computing, IEEE , vol.4, no.1, pp.18,27, Jan.-March 2005

[31] Chalasani, S.; Conrad, J.M., "A survey of energy harvesting sources for embedded systems," Southeastcon, 2008. IEEE , vol., no., pp.442,447, 3-6 April 2008

[32] Vieira, M.A.M.; Coelho, C.N.; da Silva, D.C., Jr.; da Mata, J.M., "Survey on wireless sensor network devices," Emerging Technologies and Factory Automation, 2003. Proceedings. ETFA '03. IEEE Conference , vol.1, no., pp.537,544 vol.1, 16-19 Sept. 2003

[33] Werner-Allen, G.; Lorincz, K.; Ruiz, M.; Marcillo, O.; Johnson, J.; Lees, J.; Welsh, M., "Deploying a wireless sensor network on an active volcano," Internet Computing, IEEE , vol.10, no.2, pp.18,25, March-April 2006

[34] Giuseppe Anastasi, Marco Conti, Mario Di Francesco, Andrea Passarella, "Energy conservation in wireless sensor networks: A survey", Ad Hoc Networks, Volume 7, Issue 3, May 2009, Pages 537-568

[35] Mehmet C. Vuran, Özgür B. Akan, Ian F. Akyildiz, "Spatio-temporal correlation: theory and applications for wireless sensor networks", Computer Networks, Volume 45, Issue 3, 21 June 2004, Pages 245-259,

[36] Andreas Konstantinidis, Kun Yang, "Multi-objective energy-efficient dense deployment in Wireless Sensor Networks using a hybrid problem-specific MOEA/D", Applied Soft Computing, Volume 11, Issue 6, September 2011, Pages 4117-4134,

[37] Alan Mainwaring, David Culler, Joseph Polastre, Robert Szewczyk, and John Anderson. 2002. "Wireless sensor networks for habitat monitoring" In *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications* (WSNA '02). ACM, New York, NY, USA, 88-97.

[38] Wan, Z. G.; Tan, Y.K.; Yuen, C., "Review on energy harvesting and energy management for sustainable wireless sensor networks," Communication Technology (ICCT), 2011 IEEE 13th International Conference on , vol., no., pp.362,367, 25-28 Sept. 2011

[39] Alim Dewan, Suat U. Ay, M. Nazmul Karim, Haluk Beyenal, "Alternative power sources for remote sensors: A review", Journal of Power Sources, Volume 245, 1 January 2014, Pages 129-143, ISSN 0378-7753, http://dx.doi.org/10.1016/j.jpowsour.2013.06.081.

[40] Cian Mathna, Terence O'Donnell, Rafael V. Martinez-Catala, James Rohan, Brendan O'Flynn, "Energy scavenging for long-term deployable wireless sensor networks", Talanta, Volume 75, Issue 3, 15 May 2008, Pages 613-623, ISSN 0039-9140, http://dx.doi.org/10.1016/j.talanta.2007.12.021.

[41] S. Roundy, D. Steingart, L. Frechette, P. Wright, and J. Rabaey, "Power sources for wireless sensor networks", Wireless Sensor Networks 1st Eur. Workshop (EWSN 2004)

[42] Popovici, E.; Magno, M.; Marinkovic, S., "Power management techniques for Wireless Sensor Networks: A review," Advances in Sensors and Interfaces (IWASI), 2013 5th IEEE International Workshop on , vol., no., pp.194,198, 13-14 June 2013

[43] S. Tarannum, "Energy Conservation Challenges in Wireless Sensor Networks: A comprehensive Study", Wireless Sensor Network, 2010.

[44] Gabriel Girban, Mircea Popa, Book, "A Brief Outline of Computational Limitations on Actual Wireless Sensor Networks", pp. 1-2

[45] Liu Yong-Min; Wu Shu-Ci; Nian Xiao-Hong, "The Architecture and Characteristics of Wireless Sensor Network," Computer Technology and Development, 2009. ICCTD '09. International Conference on, vol.1, no., pp.561,565, 13-15 Nov. 2009

[46] A. Sharma, R. Chaki, and U. Bhattacharya, "Applications of wireless sensor network in Intelligent Traffic System: A review", 3rd International Conference on Electronics Computer Technology (ICECT), vol.5, pp.53-57, 8-10 April 2011.

[47] A. Pantelopoulos, and N. G. Bourbakis, "A Survey on Wearable Sensor-Based Systems for Health Monitoring and Prognosis", IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, vol.40, no.1, pp.1-12, Jan. 2010.

[48] Kling, R.M., "Intel Motes: advanced sensor network platforms and applications," Microwave Symposium Digest, 2005 IEEE MTT-S International , vol., no., pp.4 pp.,, 12-17 June 2005

[49] P. Ripka, A. Tipek, Modern Sensors Handbook, ISTE, London, UK (2007)

[50] Proceedings of the Distributed Sensor Nets Workshop. Pittsburgh, PA: Dept. Comput. Sci., Carnegie Mellon Univ., 1978.

[51] Gyula Simon, Miklos Maroti, Akos Ledeczi, Gyorgy Balogh, Branislav Kusy, Andras Nadas, Gabor Pap, Janos Sallai, and Ken Frampton. 2004. Sensor network-based countersniper system. In Proceedings of the 2nd international conference on Embedded networked sensor systems (SenSys '04). ACM, New York, NY, USA, 1-12

[52] L.M.Oliveira, J.J.Rodrigues J.J, "Wireless sensor networks: a survey on environmental monitoring", Journal of Communications, vol.6, no.2, pp.143-151, Apr. 2011

[53] Ahonen, T.; Virrankoski, R.; Elmusrati, M., "Greenhouse Monitoring with Wireless Sensor Network," Mechtronic and Embedded Systems and Applications, 2008. MESA 2008. IEEE/ASME International Conference on , vol., no., pp.403,408, 12-15 Oct. 2008

[54] J. Polastre. "Design and implementation of wireless sensor networks for habitat monitoring". Master's thesis, University of California at Berkeley, 2003.

[55] Wang Huiyong; Wang Jingyang; Huang Min, "Building a Smart Home System with WSN and Service Robot,"Measuring Technology and Mechatronics Automation (ICMTMA), 2013 Fifth International Conference on , vol., no., pp.353,356, 16-17 Jan. 2013

[56] JeongGil Ko; Chenyang Lu; Srivastava, M.B.; Stankovic, J.A.; Terzis, A.; Welsh, M., "Wireless Sensor Networks for Healthcare," Proceedings of the IEEE , vol.98, no.11, pp.1947,1960, Nov. 2010

[57] Jafari, R.; Encarnacao, A.; Zahoory, A.; Dabiri, F.; Noshadi, H.; Sarrafzadeh, M., "Wireless sensor networks for health monitoring," Mobile and Ubiquitous Systems: Networking and Services, 2005. MobiQuitous 2005. The Second Annual International Conference on , vol., no., pp.479,481, 17-21 July 2005

[58] Baker, C.R.; Armijo, K.; Belka, S.; Benhabib, M.; Bhargava, V.; Burkhart, N.; Der Minassians, A.; Dervisoglu, G.; Gutnik, L.; Haick, M.B.; Ho, C.; Koplow, M.; Mangold, J.; Robinson, S.; Rosa, M.; Schwartz, M.; Sims, C.; Stoffregen, H.; Waterbury, A.; Leland, E.S.; Pering, T.; Wright, P.K., "Wireless Sensor Networks for Home Health Care," Advanced Information Networking and Applications Workshops, 2007, AINAW '07. 21st International Conference on , vol.2, no., pp.832,837, 21-23 May 2007

[59] Wark, Tim, Corke, Peter, Sikka, Pavan, Klingbeil, Lasse, Guo, Ying, Crossman, Chris, Valencia, Philip, Swain, Dave, & Bishop-Hurley, Greg (2007) "Transforming agriculture through pervasive wireless sensor networks". IEEE Pervasive Computing, 6(2), pp. 50-57.

[60] Kae Hsiang Kwong; Sasloglou, K.; Goh, H.G.; Tsung Ta Wu; Stephen, B.; Gilroy, Michael; Tachtatzis, C.; Glover, I.A.; Michie, C.; Andonovic, Ivan, "Adaptation of wireless sensor network for farming industries," Networked Sensing Systems (INSS), 2009 Sixth International Conference on , vol., no., pp.1,4, 17-19 June 2009

[61] I. Andonovic, C. Michie, M. Gilroy, H. G. Goh, K. H. Kwong, K. Sasloglou, and T. Wu, "Wireless Sensor Networks for Cattle Health Monitoring," In Proc. of ICT INNOVATIONS, 2009.

[62] Guo, Y.; Corke, P.; Poulton, G.; Wark, T.; Bishop-Hurley, G.; Swain, D., "Animal Behaviour Understanding using Wireless Sensor Networks," Local Computer Networks, Proceedings 2006 31st IEEE Conference on , vol., no., pp.607,614, 14-16 Nov. 2006

[63] Sun-Jin Kim; Jung-Hae Seo; Krishna, J.; Sun-Joong Kim, "Wireless sensor network based asset tracking service," Management of Engineering & Technology, 2008. PICMET 2008. Portland International Conference on , vol., no., pp.2643,2647, 27-31 July 2008

[64] Kay-Soon Low; Win, W.N.N.; Meng-Joo Er, "Wireless Sensor Networks for Industrial Environments," Computational Intelligence for Modelling, Control and Automation, 2005 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, International Conference on , vol.2, no., pp.271,276, 28-30 Nov. 2005

[65] Xingfa Shen; Zhi Wang; Youxian Sun, "Wireless sensor networks for industrial applications," Intelligent Control and Automation, 2004. WCICA 2004. Fifth World Congress on , vol.4, no., pp.3636,3640 Vol.4, 15-19 June 2004

[66] Gungor, V.C.; Hancke, G.P., "Industrial Wireless Sensor Networks: Challenges, Design Principles, and Technical Approaches," Industrial Electronics, IEEE Transactions on , vol.56, no.10, pp.4258,4265, Oct. 2009

[67] Salvadori, F.; De Campos, M.; Sausen, P.S.; De Camargo, R.F.; Gehrke, C.; Rech, C.; Spohn, M.A.; Oliveira, A.C., "Monitoring in Industrial Systems Using Wireless Sensor Network With Dynamic Power Management," Instrumentation and Measurement, IEEE Transactions on , vol.58, no.9, pp.3104,3111, Sept. 2009

[68] Holger Karl, Andreas Willig, "Protocols and Archiecture for Wireless Sensor Network" 2007 John Wiley & Sons, Ltd. ISBN: 978-0-470-09510-2.

[69] Healy, M.; Newe, T.; Lewis, E., "Wireless Sensor Node hardware: A review," Sensors, 2008 IEEE , vol., no., pp.621,624, 26-29 Oct. 2008

[70] C. Scheffer and P. Girdhar. "Practical Machinery Vibration Analysis and Predictive Maintenance". Elsevier, 1st edition, 2004.

[71] Chee Keong Tan, Phil Irving, David Mba, "A comparative experimental study on the diagnostic and prognostic capabilities of acoustics emission, vibration and spectrometric oil analysis for spur gears", Mechanical Systems and Signal Processing, Volume 21, Issue 1, January 2007, Pages 208-233, ISSN 0888-3270,

[72] Stephan Ebersbach, Zhongxiao Peng, "Expert system development for vibration analysis in machine condition monitoring", Expert Systems with Applications, Volume 34, Issue 1, January 2008, Pages 291-299, ISSN 0957-4174

[73] E. P. Carden, P. Fanning, "Vibration Based Condition Monitoring A Review" Journal on Structural Health Monitoring, vol. 3, issue 4, pp. 355-377, Dec. 2004.

[74] Tomasz Barszcz, Adam JabLonski, "A novel method for the optimal band selection for vibration signal demodulation and comparison with the Kurtogram", Mechanical Systems and Signal Processing, Volume 25, Issue 1, January 2011, Pages 431-451

[75] Zijun Zhang; Verma, A.; Kusiak, A., "Fault Analysis and Condition Monitoring of the Wind Turbine Gearbox," Energy Conversion, IEEE Transactions on , vol.27, no.2, pp.526,535, June 2012

[76] R. Teti, K. Jemielniak, G. O' Donnell, D. Dornfeld, Advanced monitoring of machining operations, CIRP Annals - Manufacturing Technology, Volume 59, Issue 2, 2010, Pages 717-739

[77] Immovilli, F.; Bellini, A.; Rubini, R.; Tassoni, C., "Diagnosis of Bearing Faults in Induction Machines by Vibration or Current Signals: A Critical Comparison," Industry Applications, IEEE Transactions on , vol.46, no.4, pp.1350,1359, July-Aug. 2010

[78] T Doguer and J Strackeljan, "Vibration Analysis using Time Domain Methods for the Detection of small Roller Bearing Defects", SIRM 2009 - 8th International Conference on Vibrations in Rotating Machines, 23th – 25th February, Vienna, Austria, 2009, Paper ID-16.

[79] "High Frequency Vibration Analysis", white paper, August 2011, Last accessed March 2014. http://www2.emersonprocess.com/siteadmincenter/PM%20Asset%20Optimization%20Documents/ProductWhitePapers/2130˙wp˙highfrequency.pdf

[80] Gary E. Newell, (1999) "Oil analysis cost-effective machine condition monitoring technique", Industrial Lubrication and Tribology, Vol. 51 Iss: 3, pp.119-124

[81] .H. Yin, W.H. Wang, X.P. Yan, H.L. Xiao, C.T. Wang "An integrated online oil analysis method for condition monitoring", Measurement Science and Technology, 14 (11) (2003), pp. 1973-1977

[82] X.P. Yan, C.H. Zhao, Z.Y. Lu, X.C. Zhou, H.L. Xiao, A study of information technology used in oil monitoring, Tribology International, Volume 38, Issue 10, October 2005, Pages 879-886

[83] Wang, W. and Zhang, W. (2005), A model to predict the residual life of aircraft engines based upon oil analysis data. Naval Research Logistics, 52: 276-284.

[84] M. Imran, K. Khursheed, N. Lawal, M. O'Nils and N. Ahmad, "Implementation of Wireless Vision Sensor Node for Characterization of Magnetic Particles in Fluids", IEEE Transactions on Circuits and Systems for Video Technology, vol.22, pp.1634 – 1643, Nov. 2012.

[85] M. S. Laghari, Q. A. Memon, and G. A. Khuwaja, "Knowledge Based Wear Particle Analysis," Proc. International Conference on Information Technology (ICIT), Dec. 2004, pp. 112-116.

[86] A.D.H. Thomas, T. Davies, A.R. Luxmoore, "Computer image analysis for identification of wear particles", Wear, Volume 142, Issue 2, 1 March 1991, Pages 213-226

[87] Khursheed, K.; Imran, M.; O'Nils, M.; Lawal, N., "Exploration of local and central processing for a wireless camera based sensor node," Signals and Electronic Systems (ICSES), 2010 International Conference on , vol., no., pp.147,150, 7-10 Sept. 2010

[88] Romer, K.; Mattern, F., "The design space of wireless sensor networks," Wireless Communications, IEEE , vol.11, no.6, pp.54,61, Dec. 2004

[89] Mihaela Cardei and Ding-Zhu Du. 2005. Improving wireless sensor network lifetime through power aware organization. *Wirel. Netw.* 11, 3 (May 2005), 333-340.

[90] Jin-Shyan Lee; Yu-Wei Su; Chung-Chou Shen, "A Comparative Study of Wireless Protocols: Bluetooth, UWB, ZigBee, and Wi-Fi," Industrial Electronics Society, 2007. IECON 2007. 33rd Annual Conference of the IEEE , vol., no., pp.46,51, 5-8 Nov. 2007

[91] Baker, N., "ZigBee and Bluetooth strengths and weaknesses for industrial applications," Computing & Control Engineering Journal , vol.16, no.2, pp.20,25, April-May 2005

[92] Angskog, P.; Karlsson, C.; Coll, J.F.; Chilo, J.; Stenumgaard, P., "Sources of disturbances on wireless communication in industrial and factory environments," Electromagnetic Compatibility (APEMC), 2010 Asia-Pacific Symposium on , vol., no., pp.281,284, 12-16 April 2010

[93] Hill, J.L.; Culler, D.E., "Mica: a wireless platform for deeply embedded networks," Micro, IEEE , vol.22, no.6, pp.12,24, Nov/Dec 2002

[94] T. V. Chien, H. Nguyen Chan, and T. Nguyen Huu, "A comparative study on hardware platforms for wireless sensor networks," International Journal on Advanded Science Engineering Information Technologgy, vol. 2, no. 1, 2012.

[95] Polastre, J.; Szewczyk, R.; Culler, D., "Telos: enabling ultra-low power wireless research," Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on , vol., no., pp.364,369, 15 April 2005

[96] UCLA webpage: 2014 http://nesl.ee.ucla.edu/projects/ahlos/mk2/

[97] J. Beutel, "Fast-prototyping Using the BTnode Platform," in Proc. Conference on Design, Automation and Test in Europe (DATE '06), 2006.

[98] Bader S., Krämer M. & Oelmann B. (2013). "A Domain-Specific Platform for Research in Environmental Wireless Sensor Networks". in SENSORCOMM 2013, The Seventh International Conference on Sensor Technologies and Applications, pp. 200-207

[99] Lama Nachman, Ralph Kling, Robert Adler, Jonathan Huang, and Vincent Hummel. 2005. "The Intel Mote platform: a Bluetooth-based sensor network for industrial monitoring". In*Proceedings of* the 4th international symposium on Information processing in sensor networks(IPSN '05). IEEE Press, Piscataway, NJ, USA, , Article 61

[100] "Imote2 Hardware Reference Manual", Online: 2014 http://web.univ-pau.fr/~cpham/ENSEIGNEMENT/PAU-UPPA/RESA-M2/DOC/Imote2_Hardware_Reference_Manual.pdf

[101] E. Shih, S.-H. Cho, N. Ickes, R. Min, A. Sinha, A. Wang, and A. Chandrakasan, "Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks," in Proc. ACM Mobicom, Rome, Italy, Jul. 2001, pp. 272–287.

[102] Stephen J. Bellis, Kieran Delaney, Brendan O'Flynn, John Barton, Kafil M. Razeeb, and Cian O'Mathuna. 2005. "Development of field programmable modular wireless sensor network nodes for ambient systems". Comput. Commun. 28, 13 (August 2005), 1531-1544.

[103] Dimitrios Lymberopoulos , Nissanka B. Priyantha , Feng Zhao, "mPlatform: a reconfigurable architecture and efficient data sharing mechanism for modular sensor nodes", Proceedings of the 6th international conference on Information processing in sensor networks, April 25-27, 2007, Cambridge, Massachusetts, USA

[104] J. Portilla, A. de Castro, E. de la Torre and T. Riesgo " A modular architecture for nodes in wireless sensor networks" J. Univ. Comput. Sci. (JUCS), vol. 12, no. 3, pp. 328-339, 2006

[105] Portilla, J.; Riesgo, T.; De Castro, A., "A Reconfigurable Fpga-Based Architecture for Modular Nodes in Wireless Sensor Networks," Programmable Logic, 2007. SPL '07. 2007 3rd Southern Conference on , vol., no., pp.203,206, 28-26 Feb. 2007

[106] Lombardo, M.; Camarero, J.; Valverde, J.; Portilla, J.; de la Torre, E.; Riesgo, T., "Power management techniques in an FPGA-based WSN node for high performance applications," Reconfigurable Communication-centric Systems-on-Chip (ReCoSoC), 2012 7th International Workshop on , vol., no., pp.1,8, 9-11 July 2012

[107] Valverde, J.; Otero, A.; Lopez, M.; Portilla, J.; Torre, E.D.L; Riesgo, T. Using SRAM based fpgas for power-aware high performance wireless sensor networks. Sensors 2012, 12, 2667-2692.

[108] B. Sidhu , H. Singh and A. Chhabra "Emerging wireless standards WiFi, ZigBee and WiMAX", Proc. World Academy of Science, Engineering and Technology, vol. 25, pp.308 -313

[109] Shuaib, K.; Boulmalf, M.; Sallabi, F.; Lakas, A., "Co-existence of Zigbee and WLAN, A Performance Study," Wireless Telecommunications Symposium, 2006. WTS '06 , vol., no., pp.1,6, April 2006

[110] Soo Young Shin; Hong Seong Park; Sunghyun Choi; Wook-Hyun Kwon, "Packet Error Rate Analysis of ZigBee Under WLAN and Bluetooth Interferences," Wireless Communications, IEEE Transactions on , vol.6, no.8, pp.2825,2830, August 2007

[111] Garroppo, R.G.; Gazzarrini, L.; Giordano, S.; Tavanti, L., "Experimental assessment of the coexistence of Wi-Fi, ZigBee, and Bluetooth devices," World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2011 IEEE International Symposium on a , vol., no., pp.1,9, 20-24 June 2011

[112] Eswaran, Sharanya; Edwards, James; Misra, Archan; Porta, Thomas F. La, "Adaptive In-Network Processing for Bandwidth and Energy Constrained Mission-Oriented Multihop Wireless Networks," Mobile Computing, IEEE Transactions on , vol.11, no.9, pp.1484,1498, Sept. 2012

[113] J. Meyer, J. Noguera, M. Hübner, L. Braun, O. Sander, R. Gil, R. Stewart, and J. Becker, "Fast start-up for spartan-6 fpgas using dynamic partial reconfiguration," in Design, Automation Test in Europe Conference Exhibition (DATE), 2011, march 2011, pp. 1-6.

[114] de Jesus Rangel-Magdaleno, J.; de Jesus Romero-Troncoso, R.; Osornio-Rios, R.A.; Cabal-Yepez, E.; Dominguez-Gonzalez, A., "FPGA-Based Vibration Analyzer for Continuous CNC Machinery Monitoring With Fused FFT-DWT Signal Processing," Instrumentation and Measurement, IEEE Transactions on , vol.59, no.12, pp.3184,3194, Dec. 2010

[115] Jian Huang; Peilei Rao; Xiaoguang Hu; Jin Xiao, "Research of on-line monitoring system of the condition of circuit breaker based on ARM and FPGA," Control and Automation, 2009. ICCA 2009. IEEE International Conference on , vol., no., pp.201,205, 9-11 Dec. 2009

[116] Guillermo Botella, Carlos Garcia, Uwe Meyer-Base, Hardware implementation of machine vision systems: image and video processing, EURASIP Journal on Advances in Signal Processing September 2013, 2013:152

[117] Abdul Waheed Malik ; Benny Thornberg ; Xiaozhou Meng and Muhammad Imran "Real-time machine vision system using FPGA and soft-core processor", Proc. SPIE 8437, Real-Time Image and Video Processing 2012, 84370Z (June 1, 2012)

[118] A. K. S. Jardine, D. Lin, and D. Banjevic, "A review on machinery diagnostics and prognostics implementing condition-based maintenance", Elsevier Journal on Mechanical Systems and Signal Processing, vol. 20, issue 7, pp. 1483-1510, Oct. 2006.

[119] S. Nandi and H. A. Toliyat , "Condition monitoring and fault diagnosis of electrical machines-a review", Conference proceedings on Industry Applications Conference, , vol. 1, pp. 197-204, 1999.

[120] J. Ilonen, J. Kamarainen, T. Lindh, ,J. Ahola, H. Kalviainen and J. Partanen, "Diagnosis tool for motor condition monitoring", IEEE Transactions on Industry Applications, vol. 41, issue 4, pp. 963-971, July 2005.

[121] S. A. Dyer, Survey of Instrumentation and Measurement, John Wiley & sons, 2001, pp. 535.

[122] F. Al-Badour, M. Sunar, L. Cheded, Vibration analysis of rotating machinery using time-frequency analysis and wavelet techniques, Mechanical Systems and Signal Processing, Volume 25, Issue 6, August 2011, Pages 2083-2101

[123] Yang, H., Mathew, J. and Ma, L. (2003a), "Vibration feature extraction for diagnosis of rotating machinery faults: a literature survey", Proceedings of the 10th Asia-Pacific Vibration Conference, Gold Coast, Australia, pp. 801-7.

[124] Clarence W. de Silva (Editor) "Vibration Monitoring, Testing, and Instrumentation" pp. 7-11 to 7-13, CRC Press, Taylor & Francis Groups 2007.

[125] T Doguer and J Strackeljan, "Vibration Analysis using Time Domain Methods for the Detection of small Roller Bearing Defects", SIRM 2009 - 8th International Conference on Vibrations in Rotating Machines, 23th-25th February, Vienna, Austria, 2009, Paper ID-16.

[126] A. Wetula, "A Hilbert transform based algorithm for detection of a complex envelope of a power grid signals - an implementation," in Nonsinusoidal Currents and Compensation, 2008. ISNCC 2008. International School on, June 2008.

[127] Bouchot, J.; Himmelbauer, J.; Bernhard, M., "On autocorrelation based on Hermann Weyl's discrepancy norm for time series analysis," Neural Networks (IJCNN), The 2010 International Joint Conference on , vol., no., pp.1,7, 18-23 July 2010

[128] Daubechies, I. and W. Sweldens. "Factoring wavlet transform into lifting steps". J. Fourier Anal. Appl., Vol. 4, Nr. 3, 1998.

[129] John S. Mitchell, San Juan Capistrano, "From Vibration Measurements to Condition Based Maintenance - Seventy Years of Continuous Progress", Sound and Vibration, January 2007, pp. 62-75.

[130] Wright, Paul K; Dornfeld, David; Hillaire, R. G; & Ota, Nathan K. (2006). "A Wireless Sensor for Tool Temperature Measurement and its Integration within a Manufacturing System". UC Berkeley: Laboratory for Manufacturing and Sustainability. Retrieved from: http://escholarship.org/uc/item/1zr548s8

[131] Yadong Wan; Lei Li; Jie He; Xiaotong Zhang; Qin Wang, "Anshan: Wireless Sensor Networks for Equipment Fault Diagnosis in the Process Industry," Sensor, Mesh and Ad Hoc Communications

and Networks, 2008. SECON '08. 5th Annual IEEE Communications Society Conference on , vol., no., pp.314,322, 16-20 June 2008

[132] Bin Lu; Gungor, V.C., "Online and Remote Motor Energy Monitoring and Fault Diagnostics Using Wireless Sensor Networks," Industrial Electronics, IEEE Transactions on , vol.56, no.11, pp.4651,4659, Nov. 2009

[133] Tiwari, A.; Lewis, F.L.; Shuzhi Sam Ge, "Wireless sensor network for machine condition based maintenance,"Control, Automation, Robotics and Vision Conference, 2004. ICARCV 2004 8th , vol.1, no., pp.461,467 Vol. 1, 6-9 Dec. 2004

[134] P. Wright, D. Dornfeld, and N. Ota. "Condition Monitoring in End-milling Using Wireless Sensor Networks (WSNs)." Transactions of NAMRI/SME, 36:177--183, 2008.

[135] B. Lu, L. Wu, T. G. Habetler, R. G. Harley, and J. A. Gutiérrez, "On the application of wireless sensor networks in condition monitoring and energy usage evaluation for electric machines," in Proc. 31st Annual Conf. of the IEEE Industrial Electronics Society, Nov. 2005.

[136] J. Koskinen, P. Kilpelainen, J. Rehu, P. Tukeva and M. Sallinen. "Wireless sensor networks for infrastructure and industrial monitoring applications."Conf. ICTC 2010. pp 250 - 255. [15 May 2012].

[137] Yuan Hong-fang; Wang Hua-qing; Qi He; Ke Xi-yong, "A ZigBee wireless sensor network for fault diagnosis of pump," Natural Computation (ICNC), 2010 Sixth International Conference on , vol.7, no., pp.3778,3782, 10-12 Aug. 2010

[138] Moreno-Tapia S.V., Vera-Salas L.A., Osornio-Rios R.A., Dominguez-Gonzalez A., Stiharu I., Romero-Troncoso R.D.J. "A field programmable gate array-based reconfigurable smart-sensor network for wireless monitoring of new generation computer numerically controlled machines". Sensors.2010;10:7263-7286.

[139] Liqun Hou; Bergmann, N.W., "System requirements for industrial wireless sensor networks," Emerging Technologies and Factory Automation (ETFA), 2010 IEEE Conference on , vol., no., pp.1,8, 13-16 Sept. 2010

[140] S. A. Ansari and R. Baig, "A PC-based vibration analyzer for condition monitoring of process machinery", IEEE Transactions on Instrumentation and Measurement, vol.47, issue 2, pp. 378-383, April 1998.

[141] A. Gani and M. J. E. Salami, "A LabVIEW based data acquisition system for vibration monitoring and analysis", Student Conference on Research and Development SCOReD 2002, pp. 62- 65, 2002.

[142] P. Bilski, and W. Winiecki, "Virtual spectrum analyzer based on data acquisition card," IEEE Transactions on Instrumentation and Measurement , vol.51, no.1, pp.82-87, Feb 2002.

[143] P. Bilski, and W. Winiecki, "A Low-Cost Real-Time Virtual Spectrum Analyzer," Proceedings of the IEEE Conference on Instrumentation and Measurement Technology, vol.3, pp.2216-2221, 16-19 May 2005.

[144] G. Betta, C. Liguori, A. Paolillo and A. Pietrosanto, "A DSP-based FFT-analyzer for the fault diagnosis of rotating machine based on vibration analysis," IEEE Transactions on Instrumentation and Measurement, vol. 51, issue 6, pp. 1316- 1322, December 2002.

[145] G. Betta, M. D'Apuzzo, C. Liguori, and A. Pietrosanto, "An intelligent FFT-analyzer," IEEE Transactions on Instrumentation and Measurement, vol.47, no.5, pp.1173-1179, Oct 1998.

[146] G. Betta, C. Liguori, and A Pietrosanto, "A multi-application FFT analyzer based on a DSP architecture," IEEE Transactions on Instrumentation and Measurement, , vol.50, no.3, pp.825-832, Jun 2001.

[147] L. M. C Medina,R. J. R. Troncoso, J. R. M. Almaraz and C. R. Donate, "FPGA based multiple-channel vibration analyzer embedded system for industrial applications in automatic failure

detection", International Symposium on Industrial Embedded Systems SIES 2008, pp. 229-232, June 2008.

[148] L. M. C. Medina, R. J. R. Troncoso, E. C. Yepez, J. J. R. Magdaleno and J. R. M. Almaraz, "FPGA-Based Multiple-Channel Vibration Analyzer for Industrial Applications in Induction Motor Failure Detection", IEEE Transactions on Instrumentation and Measurement, vol. 59, issue 1, pp. 63-72, Januarary 2010.

[149] A. W. Malik, B. Thörnberg, X. Cheng, and N. Lawal, "Real-time Component Labelling with Centre of Gravity Calculation on FPGA," ICONS 2011, The Sixth International Conference on Systems, Jan. 2011.

[150] M Sonka, V Hlavac, R Boyle, "Image Processing, Analysis and Machine Vision" 1993 - Springer, pp 56-111.

[151] S. Bazeille, L. J. I. Quidu, and J. P. Malkasse. "Automatic underwater image pre-processing". In Proc. of CMM, 2006.

[152] M. Spann, C. Horne, "Image segmentation using a dynamic thresholding pyramid, Pattern Recognition", Volume 22, Issue 6, 1989, Pages 719-732, ISSN 0031-3203, http://dx.doi.org/10.1016/0031-3203(89)90008-3.

[153] Nikhil R Pal, Sankar K Pal, "A review on image segmentation techniques, Pattern Recognition", Volume 26, Issue 9, September 1993, Pages 1277-1294,

[154] Rafael C. Gonzalez and Richard E. Woods, "Digital image processing", 3rd edition. Pearson International Edition.

[155] Ian F. Akyildiz, Tommaso Melodia, Kaushik R. Chowdhury, " A survey on wireless multimedia sensor networks" , Computer Networks, Volume 51, Issue 4, 14 March 2007, Pages 921-960, ISSN 1389-1286,

[156] Mohammad Rahimi, Rick Baer, Obimdinachi I. Iroezi, Juan C. Garcia, Jay Warrior, Deborah Estrin, and Mani Srivastava. 2005. " Cyclops: in situ image sensing and interpretation in wireless sensor networks" . In Proceedings of the 3rd international conference on Embedded networked sensor systems (SenSys '05). ACM, New York, NY, USA, 192-204.

[157] Chao Hu Zhiyong; Liu Yingzi Pan; Zhenxing Zeng; Meng, M.Q.-H., " A novel FPGA-based wireless vision sensor node," Automation and Logistics, 2009. ICAL '09. IEEE International Conference on , vol., no., pp.841,846, 5-7 Aug. 2009

[158] Bakkali, M.; Carmona-Galan, R.; Rodriguez-Vazquez, A., "A prototype node for wireless vision sensor network applications development," I/V Communications and Mobile Network (ISVC), 2010 5th International Symposium on , vol., no., pp.1,4, Sept. 30 2010-Oct. 2 2010

[159] Taslidere, E.; Cohen, F.S.; Reisman, F.K., "Wireless Sensor Networks—A Hands-On Modular Experiments Platform for Enhanced Pedagogical Learning," Education, IEEE Transactions on , vol.54, no.1, pp.24,33, Feb. 2011

[160] Khursheed, K.; Imran, M.; Malik, A.W.; O'Nils, M.; Lawal, N.; Thörnberg, B., "Exploration of Tasks Partitioning between Hardware Software and Locality for a Wireless Camera Based Vision Sensor Node," Parallel Computing in Electrical Engineering (PARELEC), 2011 6th International Symposium on , vol., no., pp.127,132, 3-7 April 2011

[161] Med Lassaad Kaddachi, Adel Soudani, Vincent Lecuire, Kholdoun Torki, Leila Makkaoui, Jean-Marie Moureaux, "Low power hardware-based image compression solution for wireless camera sensor networks", Computer Standards & Interfaces, Volume 34, Issue 1, January 2012, Pages 14-23,

[162] Purushottam Kulkarni, Deepak Ganesan, Prashant Shenoy, and Qifeng Lu. 2005. "SensEye: a multi-tier camera sensor network". In Proceedings of the 13th annual ACM international conference on Multimedia (MULTIMEDIA '05). ACM, New York, NY, USA, 229-238.

[163] Hengstler, S.; Prashanth, D.; Sufen Fong; Aghajan, H., "MeshEye: A Hybrid-Resolution Smart Camera Mote for Applications in Distributed Intelligent Surveillance," Information Processing in Sensor Networks, 2007. IPSN 2007. 6th International Symposium on , vol., no., pp.360,369, 25-27 April 2007

[164] A. Rowe, A. Goode, D. Goel, and I. Nourbakhsh, "CMUcam3: an open programmable embedded vision sensor," Tech. Rep. RI-TR-07-13, Carnegie Mellon Robotics Institute, 2007.

[165] Chen, P.; Ahammad, P.; Boyer, C.; Shih-I Huang; Leon Lin; Lobaton, E.; Meingast, M.; Songhwai Oh; Wang, S.; Posu Yan; Yang, A.Y.; Chuohao Yeo; Lung-Chung Chang; Tygar, J. D.; Sastry, S.S., "CITRIC: A low-bandwidth wireless camera network platform," Distributed Smart Cameras, 2008. ICDSC 2008. Second ACM/IEEE International Conference on , vol., no., pp.1,10, 7-11 Sept. 2008

[166] T. Doguer and J. Strackeljan,"Vibration Analysis using Time Domain Methods for the Detection of small Roller Bearing Defects," SIRM 2009 - 8th International Conference on Vibrations in Rotating Machines, Vienna, Austria, Paper ID-16, February 23-25, 2009.

[167] D. Anguita, D. Brizzolara, and G. Parodi, "Optical wireless communication for underwater Wireless Sensor Networks: Hardware modules and circuits design and implementation," OCEANS 2010 , vol., no., pp.1,8, 20-23 Sept. 2010

[168] J. Zhang, L. Shan, H. Hu, and Y. Yang, "Mobile cellular networks and wireless sensor networks: toward convergence," Communications Magazine, IEEE , vol.50, no.3, pp.164,169, March 2012

[169] T. Li, M. Huang, J. Shi, J. Yang, J. Yu, and J. Hu, "A pilot AMR system based on WIMAX and WSN," 9th International Symposium on Propagation and EM Theory (ISAPE), pp.1176,1179, Nov. 29 2010-Dec. 2 2010.

[170] Sarkar, N.I.; Sowerby, K.W., "Wi-Fi Performance Measurements in the Crowded Office Environment: a Case Study," Communication Technology, 2006. ICCT '06. International Conference on , vol., no., pp.1,4, 27-30 Nov. 2006

[171] Kawade, S.; Hodgkinson, T.G., "Analysis of Interference Effects between Co-Existent 802.11b and 802.11g Wi-Fi Systems," Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE , vol., no., pp.1881,1885, 11-14 May 2008

[172] MODEL TL-2100, Lake Success, NY: Tadiron Batteries, 2014. [Online]: Available: www.tadiranbat.com.

[173] M. Imran, K. Khursheed, N. Lawal, M. O'Nils and N. Ahmad, "Implementation of Wireless Vision Sensor Node for Characterization of Magnetic Particles in Fluids", IEEE Transactions on Circuits and Systems for Video Technology, vol.22, pp.1634-1643, Nov. 2012.

[174] CC2520, Dallas, TX: Texas Instruments, 2014. [Online]. Available: www.ti.com.

[175] http://en.wikipedia.org/wiki/List˙of˙CPU˙power˙dissipation˙figures; Last accessed on March 11, 2014.

[176] David Barrie Thomas, Lee Howes, and Wayne Luk. 2009. A comparison of CPUs, GPUs, FPGAs, and massively parallel processor arrays for random number generation. In Proceedings of the ACM/SIGDA international symposium on Field programmable gate arrays (FPGA '09). ACM, New York, NY, USA, 63-72.

[177] "Spartan-6 FPGA PCB Design and Pin Planning", San Jose, CA: Xilinx Inc. 2014. [Online]. Available: www.xilinx.com.

[178] K. Shahzad, P. Cheng, and B. Oelmann, "Architecture exploration for a high-performance and low-power wireless vibration analyzer," IEEE Sensors Journal, vol.13, no.2, pp.670-682, Feb. 2013.

[179] C. H. Zhiyong, L. Y. Pan, Z. Zhenxing, and M. Q. H. Meng, "A novel FPGA-based wireless vision sensor node," IEEE International Conference on Automation and Logistics, 2009, pp.841-846, 5-7 Aug. 2009.

[180] Y. Sun, L. Li, and H. Luo , "Design of FPGA-Based Multimedia Node for WSN", 7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM), pp.1-5, 23-25 Sept. 2011.

[181] J. Portilla, T. Riesgo, and A. de Castro, "A Reconfigurable FPGA-Based Architecture for Modular Nodes in Wireless Sensor Networks", 3rd Southern Conference on Programmable Logic, pp.203-206, 28-26 Feb. 2007.

[182] J. Henaut, D. Dragomirescu, and R. Plana, "FPGA Based High Date Rate Radio Interfaces for Aerospace Wireless Sensor Systems," 4rth International Conference on Systems, ICONS '09, pp.173-178, 1-6 March 2009.

[183] A.A.M Bsoul, R. Hoskinson, M. Ivanov, S. Mirabbasi, H. Abdollahi, "Implementation of an FPGA-based low-power video processing module for a head-mounted display system", IEEE International Conference on Consumer Electronics (ICCE), pp. 214,217, 11-14 Jan. 2013.

[184] E. C. Yepez, R. A Osornio-Rios, R. J. Romero-Troncoso, J. R. Razo-Hernandez, and R. Lopez-Garcia, "FPGA-Based Online Induction Motor Multiple-Fault Detection with Fused FFT and Wavelet Analysis," International Conference on Reconfigurable Computing and FPGAs, pp. 101,106, 9-11 Dec. 2009.

[185] M.L. Kaddachi, L. Makkaoui, A. Soudani, V. Lecuire, and J. Moureaux, "FPGA-based image compression for low-power Wireless Camera Sensor Networks," 3rd International Conference on Next Generation Networks and Services (NGNS), pp. 68-71, 18-20 Dec. 2011.

[186] D. G. Bailey, "Design for embedded image processing on FPGAs", 2011 John Wiley & Sons (Asia) Pte Ltd, ISBN 978-0-470-82849-6, pp. 54-72, 2011.

[187] R. Maheswar, P. Jayarajan, and F. N. Sheriff. "A Survey on Duty Cycling Schemes for Wireless Sensor Networks." International Journal of Computer Networks and Wireless Communications (IJCNWC), vol. 3, no. 1,pp. 37-40, February 2013.

[188] H. Yoo, M. Shim, and D. Kim, "Dynamic Duty-Cycle Scheduling Schemes for Energy-Harvesting Wireless Sensor Networks," IEEE Communications Letters, vol.16, no.2, pp.202-204, February 2012.

[189] IGLOO AGL1000V2FPGA144 Aliso Viejo, CA: Microsemi Corporation. 2014. [Online]. Available: www.actel.com.

[190] Battery model # 30006, CA, USA. [Online]. Available: www.tenergy.com.

[191] Aptina Imaging Corporation., MT9V011, CMOS Image Sensor (2009) [online]. Available: http:// www.aptina.com/.

[192] M. Imran, N. Ahmad, K. Khursheed, M. O'Nils and N. Lawal, "Low Complexity Background Subtraction for Wireless Vision Sensor Node", 16TH Euromicro Conference on Digital System Design, Spain, Sept. 2013.

[193] M. Imran, N. Ahmad, K. Khursheed, M. A. Waheed, N. Lawal, M. O'Nils, "Implementation of Wireless Vision Sensor Node With a Lightweight Bi-Level Video Coding", IEEE Journal on Emerging and Selected Topics in Circuits and Systems, vol.PP, pp.1,12, May. 2013.