IJERST

International Journal of Engineering Research and Science & Technology

ISSN : 2319-5991
Vol. 5, No. 2
May 2016

www.ijerst.com

Email: editorijerst@gmail.com or editor@ijerst.com

Int. J. Engg. Res. & Sci. & Tech. 2016

International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com
Vol. 5, No. 2, May 2016

Research Paper

DESIGN AND IMPLEMENTATION OF ENERGY EFFICIENT AND COVERAGE AWARE SELF ORGANIZING WIRELESS SENSOR NODES IN FAULT CONDITIONS

U Radhika[1]* and B Selva Lakshmi[1]

Corresponding Author: **U Radhika ✉ radhirose31@gmail.com*

In recent years, Wireless Sensor Networks (WSNs) research has undergone a quiet revolution, providing a new paradigm for sensing and disseminating information from various environments. In reality, the wireless propagation channel in many harsh environments has a significant impact on the coverage range and quality of the radio links between the wireless nodes (motes). Therefore, the use of diversity techniques (e.g., frequency diversity and spatial diversity) must be considered to be unpredictable point-to-point radio communication links. However, in order to determine the space and frequency diversity characteristics of the channel, accurate measurements need to be made. However they suffer poor accuracy owing to their low-cost and compromised Radio Frequency (RF) performance. The proposed strategy provides us with good knowledge of the actual mote transmit power, RSSI characteristics and receive sensitivity by establishing calibration tables for transmitting and receiving mote pairs over their operating frequency range. The validated results show that our automated calibration system can achieve an increase of <1.5 dB in the RSSI accuracy. In addition, measurements of the mote transmit power show a significant difference from that claimed in the manufacturer's data sheet. The proposed calibration method can also be easily applied to wireless sensor motes from virtually any vendor, provided they have a RF connector.

Keywords: Wireless Sensor Node, RSSI, ADC, Microcontroller, DSP, ASIC, FPGA

INTRODUCTION

The main aim of our project is to provide energy efficient and coverage aware sensor nodes. Because sensor plays a vital role in all fields. If any sensor node fail in a wireless network then it will be difficult to cover the range.

Wireless Sensor Networks (WSNs) promise to have a significant impact on a broad range of applications relating to structural engineering, agriculture and forestry, national security, energy, food, logistics and transportation (Lau and Lyons, 2007). A WSN is a collection of self-contained,

---

[1] Department of Electronics and Communication, Star Lion College of Engineering and Technology, Manankorai, India.

micro-electro-mechanical devices; these nodes are colloquially referred to as motes. Each of them contains a computational unit, memory, a wireless network radio and a power supply. They are deployed in the environments of interest to monitor various physical conditions, such as temperature, relative humidity, pressure, vibration, sound, motion, and various pollutants. Furthermore, recent advances in technology of hybrid circuits, Micro-Electro-Mechanical Systems (MEMS), and mixed signal Application-Specific Integrated Circuit (ASIC) design have already decreased the size, weight and cost of sensors (e.g., pressure transducers, hydrophones, multi-parameter water probes) by orders of magnitude. Many aspects of WSN design, including network architecture, protocols, signal processing and hardware have previously been studied (Kemighan, 1988). However, the planning, deployment and management of WSN systems have received relatively little attention to date. This is unfortunate since in most applications, a WSN needs to be carefully planned, deployed and managed if it is to meet service expectations. It is also the case that planning, deployment and management tools, where they exist, are poorly integrated and posses limited functionality. This is partly as a result of simplistic assumptions concerning wireless propagation and its effect on communication link performance. In fact, radio communication links between wireless motes are notoriously variable and unpredictable, which has a significant impact on the coverage range and quality of the radio link. A link that is acceptable today may be poor tomorrow due to environmental conditions, e.g., new obstacles, unanticipated interferers and myriad other factors (Han *et al.,* 2005). That is, propagation path loss, channel fading, RF interference and changes in the physical environment that block communication links all contribute to the dramatic variation in the received signal strength and error rate of the communication links. These effects need to be taken into account in the design of effective Planning and deployment tools.

In addition to improved planning and deployment tools, the task of implementing WSNs can be eased by improving the performance of the communication links between motes. To do this, it is worthwhile to investigate techniques such as frequency and spatial diversity to see if they can improve the coverage and robustness of WSNs in the environments of interest. The potential benefits of using the proposed diversity techniques need to be quantified by performing accurate propagation measurements in the environments of interest. To do this in a realistic and inexpensive way we propose the use of standard motes manufactured by Crossbow Technology Inc. (Akyildiz, 1999), namely Micas motes. However, we need to undertake a comprehensive analysis of mote RF performance to determine their suitability for this task. Specifically, we require knowledge of the mote transmit power and Receive Signal Strength Indicator (RSSI) characteristics over the system operating frequency range. The use of standard motes will also permit us to take channel measurements in situations that closely replicate actual WSN deployments. However, it is necessary that we calibrate them in advance. This is because these motes are typically deployed en masse and so must be inexpensive and have low power consumption. Based upon preliminary measurements it appears that the mote RF performance is usually below the manufacturers' quoted specifications and that their characteristics vary from one to another after production. Therefore, it appears that conducting measurements using uncaliberated motes is not to be recommended (Gay *et al.,* 2003; Abrach *et al.,* 2003).

We show in this report how mote variability can be overcome using our proposed method for automatically calibrating pairs of motes using a networked computer driven instrument system. Our approach uses a MATALB based computational environment running on a PC to control a signal generator, a spectrum analyzer and a pair of transmitting (Tx) and receiving (Rx) motes to establish an accurate relationship between the hardware dependent RSSI values and the true input power at the receiver antenna connector. Moreover, accurate knowledge of the transmit power at the transmitter antenna connector enables accurate estimation of path loss over the frequency band of interest to be performed. These measurements form the basis of lookup tables for any pair of TX and Rx motes. Finally, the actual receive sensitivity can be revealed from this tabulated information. Accurate knowledge of these parameters as a function of operating frequency enables us to use the motes as a calibrated measurement system. The automated calibration system targets the Berkeley MicaZ platform from Crossbow Technology Inc. However, this method can be easily adapted and applied to any other platforms from different vendors, provided they have a RF connector.

We begin in Section 2 by presenting some background concerning our research and introducing the components that are involved in the proposed calibration system. Section 3 details the design of the calibration subsystem for the mote receiver and the design of the calibration subsystem for the mote transmitter. Both contain an overview of the subsystems and the algorithms that are used to control the system. This leads naturally to the implementation of the two automated calibration subsystems, as will be described in Section 4. Section 5 describes how the performance improvements yielded by the proposed calibration lookup tables for both the transmitter and the receiver motes are quantified.

BACKGROUND AND RELATED WORK

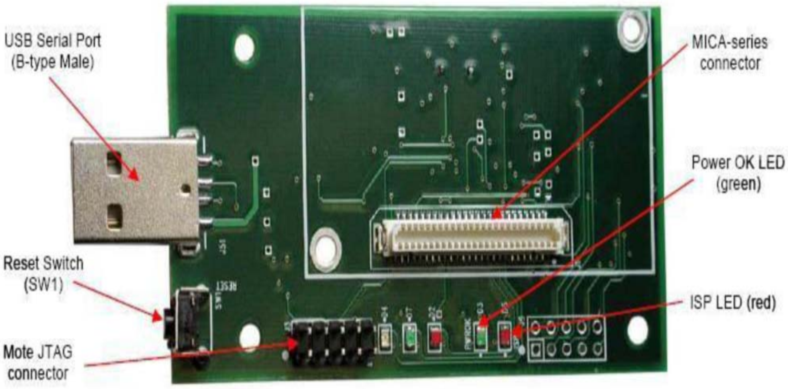
Device calibration is the process of forcing a device to conform to a given input/output mapping. This is often done by adjusting the device internally but can equivalently be done externally by passing the device's output through a calibration function that maps the actual device response to the desired response (Hill *et al.,* 2000). A more formal description of calibration is this: each device has a set of parameters. The purpose of calibration is to choose the correct parameters for each device such that they, in conjunction with a calibration function, will translate any actual device output r into the corresponding desired output r^*. The calibration function must therefore be of the form

A. MICAZ

The Berkeley MicaZ (Whitehouse, 2002) mote was selected as our measurement platform. It is a 2.4 GHz Mote module used for enabling low-power, wireless sensor networks. The platform board MPR2400 is based on the Atmel ATmega128L. The ATmega128L is a low-power microcontroller which runs programs from its internal flash memory. Figure 1 shows the MPR2400-MicaZ.

A single processor board MPR2400 can be configured to run developers' sensor application/processing and the network/radio communication stack simultaneously. The 51-pin expansion connector supports Analog Inputs, Digital I/O, I2C,

Figure 1: Photo of the MPR2400-MicaZ



SPI and UART interfaces. These interfaces make it easy to connect to a wide variety of external peripherals.

1. CC2420 Radio Transceiver

The CC2420 RF transceiver is mounted on the MPR2400 board for the purpose of wireless communication. It is a single-chip 2.4 GHz IEEE 802.15.4 compliant RF transceiver designed for low power and low voltage wireless applications (Mini Circuits). CC2420 includes a digital Direct Sequence Spread Spectrum (DSSS) baseband modem providing a spreading gain of 9 dB and an effective data rate of 250 kbps. The MicaZ's CC2420 radio can be tuned from 2.048 GHz to 3.072 GHz which includes the global Industrial, Scientific and Medical (ISM) band at 2.4 GHz. IEEE 802.15.4 channels are numbered from 11 (2.405 GHz) to 26 (2.480 GHz) each separated by 5 MHz.

The CC2420 provides one very important piece of metadata about received packets. This is Received Signal Strength Indicator (RSSI), which is a measurement of the power in dBm present in a received radio signal. It is calculated over the first eight symbols after the start of a packet frame. RSSI can also be sampled at other times, to detect the ambient RF energy. RF transmission power is programmable from 0 dBm to -25 dBm. Typically, the CC2420 consumes the current of 18.8 mA in the transmit mode and that of 17.4 mA in the receive mode and have a typical sensitivity of –95 dBm.

B. MIB520 USB Interface Board

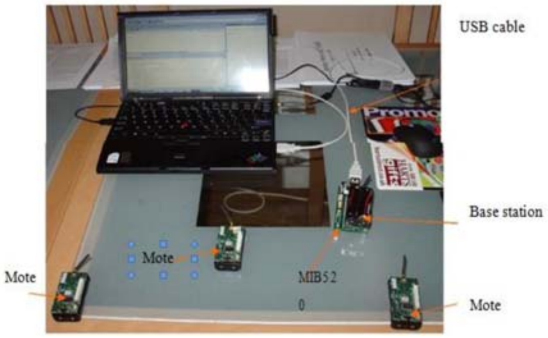
The MIB520 shown in Figure 2 is a multi-purpose USB interface board that provides USB connectivity to the Mica family of motes for communication and in-system programming. The

Figure 2: Top View of an MIB520CB

MIB 520 has an on-board in-system processor (ISP) – an ATmega16L to program the motes. Code is downloaded from a PC to the ISP through the USB port. Next the ISP programs the code into the mote.

The mote which is attached to the MICA-series connector of the MIB520 is defined as the base station. It allows the aggregation of sensor network data onto a PC.

Figure 3: Demonstration of a Simple WSN



Any MicaZ mote can function as a base station when it is connected to the MIB520. Therefore, the MIB520 provides a fundamental serial/USB interface for both programming and data communications for any WSN.

C. Tinyos Operating System And NESC Language

Programming MicaZ motes requires having the TinyOS operating system installed in the host PC. The MicaZ motes connect to the MIB520 for UISP (Micro in-system programmer) programming from the USB connected host PC. TinyOS (MS2721A) is an open-source operating system specifically designed for network embedded system. Moreover, it is the dominant operating system for WSNs, partly because it was the first one released for widespread use and was the only one available for a significant period of time in comparison to the alternatives, e.g., SOS (Levis *et al.,* 2005), MANTIS OS (Saha, 2003) and T2 (Yao, 2002).

TinyOS uses an event-driven concurrency model and utilises a component-based architecture, which makes it easy to move the boundary between dedicated hardware and software emulation. This allows power saving and efficient usage of memory space by shutting down or disabling unused components. Therefore, this component-based architecture minimizes the overall program size since only essential components are included. The core TinyOS requires 400 bytes of code and data memory combined TinyOS is implemented in the nesC language, an extension to C. The nesC programming language supports the TinyOS component and concurrency model as well as extensive cross-component optimizations and compile-time race detection. A complete TinyOS system consists of a tiny scheduler and a graph of components, each of which is an independent computational entity that exposes one or more interfaces. There are two types of components in nesC: modules and configurations.

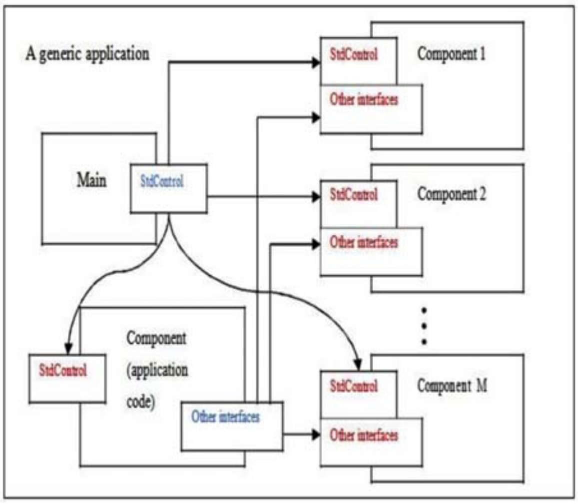
Applications written in nesC that run on wireless sensor motes are built by writing and assembling different components as required. The interfaces are the only point of access to the component, and they are also bidirectional: they contain commands and events. Commands and events are mechanisms for inter-component communication. A command is typically a request to a component to perform some service, such as initiating a sensor reading, while an event signals the completion of that service. Events may also be signaled asynchronously, for example, due to hardware interrupts or message arrival. From a traditional OS perspective, commands are analogous to down calls and events to up calls. Generally speaking, every nesC application is described by a top level configuration that wires together the components used. This arrangement is shown in Figure 4.

TinyOS as well as nesC provide a revolutionary solution for networked embedded systems such as motes by implementing an efficient framework for modularity and a resource-constrained, event-driven concurrency model. TinyOS is by no means a finished system. It continues to evolve and grow, and the latest version of TinyOS is TinyOS 2.x. During the time when the proposed auto-calibration system was developed, TinyOS 1.1.15 and nesC 1.1-1w were used.

D. Matlab and Measurement Instruments

It is very useful for us to be able to interact with a wireless mote or sensor network using MATLAB. MATLAB is a powerful matrix-based programming language with a number of libraries available for plotting and analysing data. It provides a command-like environment from which users can send or receive messages to or from their WSN and analyze data after collection. Users can also use all of TinyOS's Java tools from within the MATALB environment for easy visualisation and manipulation of incoming and outgoing data. In addition to this, MATLAB provides a very rich library of toolboxes to users for various purposes. The version of MATLAB used here is 7.5.0.342 (R2007b).The ultimate goal of the proposed automated calibration system is to control every component using MATLAB in order to establish calibration lookup tables for both receive and transmit motes. The utilization of the Instrument Control Toolbox built in MATLAB makes it possible to interact with the measurement instruments involved. The Instrument Control Toolbox of version 2.5 lets users communicate with instruments, such as oscilloscopes, function generators, and analytical instruments, directly from MATLAB. With the toolbox, users can generate data in MATLAB to send out to an instrument, or read data into MATLAB for analysis and visualization. The toolbox realizes the communication between MATLAB and instruments by providing a consistent interface to all devices independent of hardware manufacturer, protocol, or driver. Generally speaking, the Instrument Control Toolbox provides three ways to communicate with instruments, including:

Figure 4: Top Level Configuration



Instrument drivers: The toolbox supports VXI plug & play, IVI, and MATLAB instrument drivers. VXI plug & play and IVI instrument drivers often ship with instruments and are also available from the instrument manufacturers' web sites. MATLAB instrument drivers can be created by using driver development tools provided in the Instrument Control Toolbox. The biggest advantage of this approach is that learning instrument-specific commands, such as Standard Commands for Programmable Instruments (SCPI), is avoided and common MATLAB terminology is used to interact with instruments.

Communication protocols: The toolbox supports communication protocols, including GPIB, serial, TCP/IP, and UDP, for directly communicating with instruments. Alternatively, instruments can be accessed by using Virtual Instrument Software Architecture (VISA) over GPIB, VXI, USB, TCP/IP, and serial buses. The SCPI is required to control programmable test and measurement devices.

Test and Measurement Tool: This is the GUI which allows users to communicate with and configure instruments without writing code. This tool can also automatically generate M-code from instrument control sessions for fine-grained modification.

In our application, MATLAB needs to control two types of measurement instruments, specifically the Agilent E4432B ESG-D3000B RF signal generator and the Anritsu MS2721A spectrum analyzer. The signal generator operates over a frequency range from 250 kHz to 3 GHz, and it uses a USB/GPIB interface that enables fast and direct communication between the host PC and the instrument. Therefore, MATLAB can send commands to the signal generator via the USB/GPIB bus. The spectrum analyzer is small, portable and easy to use with measurement capability for applications up to 7.1 GHz. It is used as a part of the automated calibration system for Tx motes, and a direct Ethernet connection via a RJ-45 cable is required to connect the spectrum analyzer to the PC for programming the instrument and capturing the measured data into MATLAB. 5

SYSTEM ARCHITECTURE

In a wireless senor network, one of the main design challenges is to achieve long network lifetime by turning off some redundant sensor nodes while maintaining sufficient coverage for sensing and connectivity. In this project, an improved relocation algorithm is proposed to find if a sensor is completely covered by its neighbors so as to decide whether it is redundant and can be used to replace failed nodes.

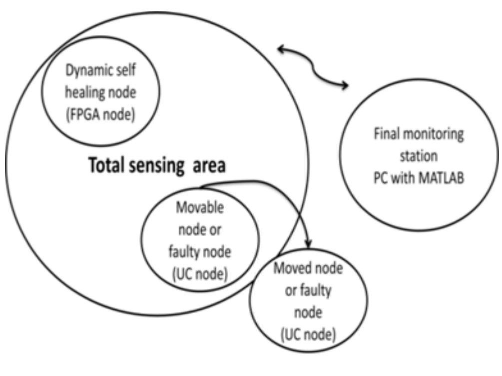
This project proposes a recovery algorithm, which is energy-efficient and responsive to network topology changes due to sensor node failures. The proposed failure-recovery mechanism recovers the connectivity of the cluster in almost less than of the time taken by already proposed techniques. Also it used both received signal strength and MEMS based x y z co-ordinates calculation techniques.

A. Key Design

- Detection and identification of sensor node failure
- Estimation of the sensor node displacement
- Self healing of coverage area using transmit power control algorithm

The two major process involved in it are First, Coverage aware sensor deployment scheme

Figure 5: System Architecture



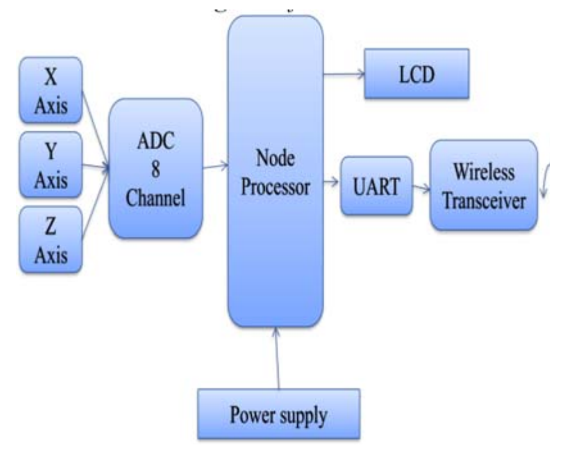
should be developed to ensure sufficient sensing coverage.

Second, in the face of sensing node failures, a sensor self-organizing mechanism needs to be devised to efficiently recover the sensing void and restore the required sensing coverage.

B. System Architecture

In this wireless sensor network, we are using two node one is dynamic node and other one is movable node. When the movable node move out of sensing range or anything wrong in it that will be detected by the dynamic node which is self healing node. That detectable node will start the self healing process based on transmission power control method and received signal strength that node mode according to provide full coverage in the sensing area even though if one node failed in it.

Figure 6: Block Diagram of Sensor Node



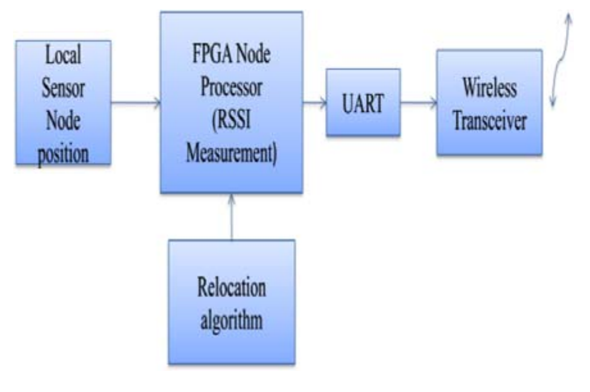
1. Block Diagram of Sensor Node

MEMS sensor –to find the x,y and z position of nodes.8ADC-X, Y, Z Values are send to it. Node processor-process the information If it moves then it will send the position through UART.

2. Block Diagram of Relocation Node

Local sensor node- Provide the process and also it check the neighbor node position. Node-if the node moves or received signal reduced then it make to move the node Based on RSSI Measurement and TPC algorithm.

Figure 7: Block Diagram of Relocation Node



C. Advantages

1) Total Area coverage
2) Dynamic movement on sensor failures
3) Low power
4) 24*7 monitoring

D. Application

Figure 8: Schematic View

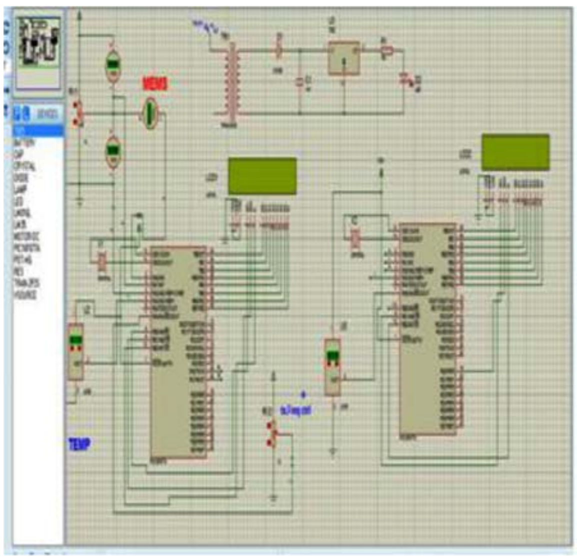


Figure 9: Simulation Result

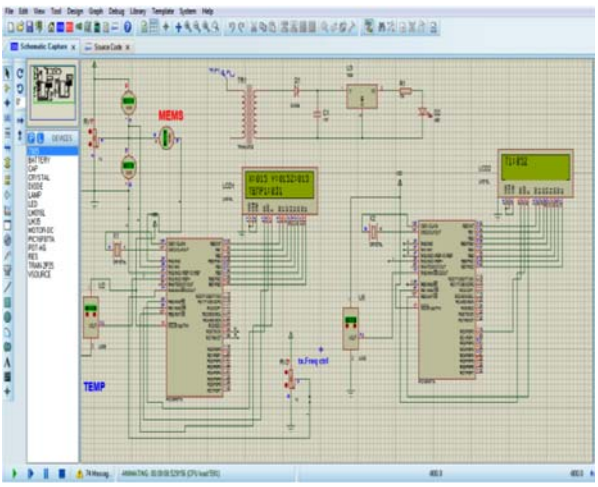


Figure 10: Before Node Get Move

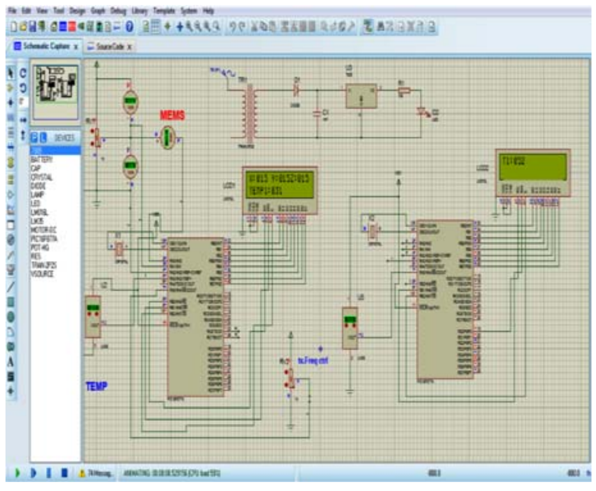
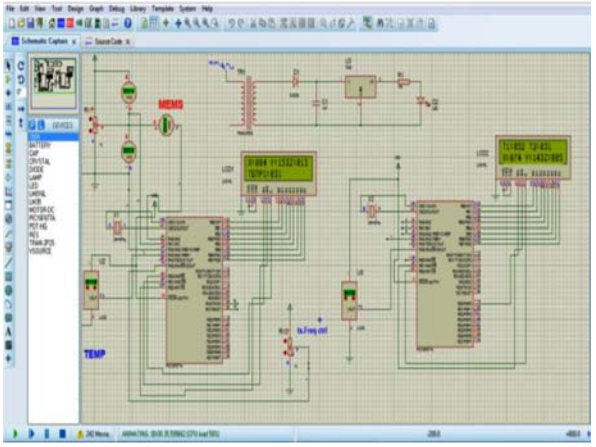


Figure 11: After the Node Movement



1) Industrial monitoring-Temperature monitoring of a boiler.
2) Environment monitoring-Aware about natural disasters like tsunami.
3) Home monitoring-Security purpose.

SIMULATION RESULTS DISCUSSION

A. Xbee-pro S2b Zigbee Coverage Range And Power Consumption

The results of our measurements show that ZigBee and XBee modules are reliable in the presence of various obstacles in both indoor and outdoor scenarios. In outdoor measurement the maximum, line of sight, range of XBee-Pro S2B at power level 0 is 160 m. This can be due to many factors which cause signal outage propagation, such as trees and the presence of people. This can be a problem for designing a WSN which can be avoided by rising the transmit power of module to P4 then the distance will become 220 m as is shown in our measurements. On the other hand, applications such as those for monitoring agriculture fields may use XBee modules at power level P0 in case the modules are placed on a level higher than the plants and hence facesless obstacles. For indoor environments, the discussion follows the same track as the outdoor case where the maximum, non-line of sight, range at transmit power level 0 is 35 m and at level 4 is 44 m.On the other hand, we can see that the difference in power consumption between every two consecutive levels is not large, through it can have an influence on the network life time in the case of P0 and P4. Thus we can conclude that the transmit power of XBee-Pro S2B can be reduced in an environment with less obstacles, then the power consumption of the node can be reduces.

B. Choosing Network Topology

Our measurements on ZigBee devices show that the XBee end device consumes less power than the router as is expected. The significant difference in power consumption between router and end device is present in the idle mode of operation since the router has to listen to the channel all the time. According to this, we can conclude that the type of node is a very important factor in designing the WSN with ZigBee protocol.In accordance with the above results, the choice of topology plays an important role in defining the operating nodes in the network. In other words, reducing power consumption in the network can be achieved by replacing as many routers as possible with an end device. But still, the tradeoff between power consumption and reliability must be taken into consideration since using end devices will reduce the number of links, hence a drop of one link can present a significant problem in some applications. This requires an intelligent network design.

ZigBee protocol supports networks with star topology and mesh topology with the ability to extend the network by using cluster tree topology. An overview of our measurements; star topology can present the best designing choice when the application demand is within the capability of the XBee ZigBee coverage range. In star topology all devices can be configured as end devices and they can only communicate with the coordinator. To extend the range of the network, the designer can use cluster tree topology. In cluster trees only the PAN coordinators can communicate with each other. Routers can connect each cluster head and end devices are considered as leaves which are communicating with the coordinator either directly or via routers. Mesh topology can be another choice for large networks. Since mesh topology is a self-healing, reliable and self-organizing network, it can be the choice for some applications such as industrial monitoring applications which demand a high level of reliability. Therefore by using mesh topology the designer increases the number of links but consequently a high number of routers will be existed in the network. Consequently, this lasts with significant cost of power consumption in comparison to a topology with more end devices.

Hence, the choice of topology depends on the application and the availability of energy for powering nodes and the required level of reliability.

C. Further Work

Future work includes implementing several nodes in order to extend the measurements in larger network. Future measurements include the effect of other parameters on designing WSNs such as packet delay. To this aim, a software tool should be designed to study the packet delay in multi-hop networking since we find the X-CTU tool is not flexible enough for this experiment. We also intend to investigate in the possible measurement tools in order to measure the power consumption more precisely.

Future work can also include the effect of encryption on network life time, packet delay and power consumption. An automatic measurement tool of RSSI in real life conditions and large network is another future aim.

CONCLUSION

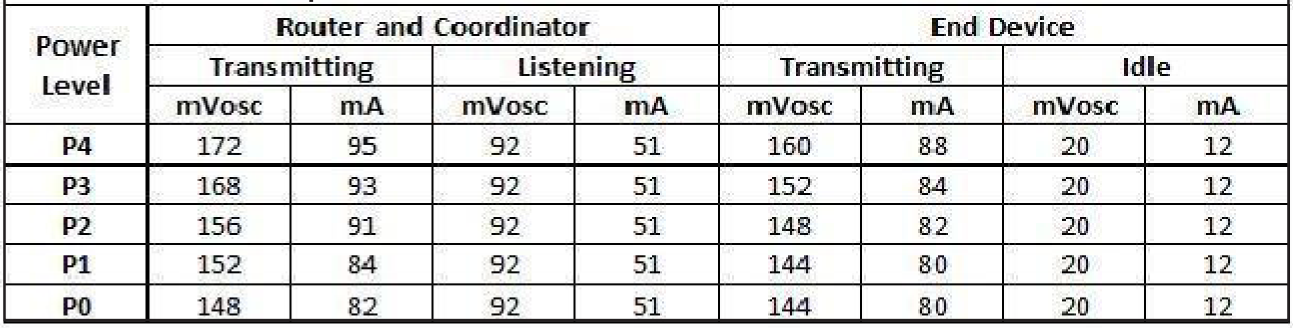
In this work, we intensively studied different architectural aspects and requirements for designing WSNs. Later we investigated different technologies and protocols of WSNs Including a comprehensive study of XBee ZigBee modules capabilities, an analysis of the power consumption of XBee ZigBee modules as well as a comparison with other wireless devices. We studied the structure of ZigBee protocol in order to design a multi-hop wireless sensor network including addressing, ZigBee devices operational modes and communication modes. Then, we implemented our embedded temperature wireless sensor node using a PIC microcontroller utilizing an XBee-Pro S2B module to create a sensing phenomenon in the network. Using our sensor node along with other development kits and software tools we were able to design a self-organizing WSN.

We went through a series of experiments to measure different parameters in our WSN. We measured the power consumption in ZigBee protocol devices; coordinator, router and end device during their different operational modes. We also estimated the power consumption of these devices at different transmit power levels (P4, P3, P2, P1, P0) which are supported by XBee-Pro S2B. We noticed that the difference in the power consumption of end devices and router/coordinator is large. The summary of our measurements is shown in Table 1.

To evaluate the capability of XBee-Pro S2B transceiver coverage range in real life conditions, we measured the XBee coverage range in outdoor and indoor scenarios at transmit power level P0 and P4. Table 2 summarizes the maximum distance in every scenario with the power consumption at every level.

We discussed the trade-off between the coverage range of XBee-Pro S2B in real life conditions at different transmit power levels and its power consumption. We also discussed the effect of the topology used in designing WSN on the power consumption. Finally we discussed further developments in order to study the effect of more parameters such as multi-hop delay on WSN design. Also be easily applied to any other wireless sensor mote platforms from different vendors, provided they have an RF connector. In the future the research could be extended by investigating the possibility of reducing the time taken to perform the proposed automated calibration processes to enable mass calibration of wireless sensor motes. Moreover, we have shown that standard motes can be used as the low cost portable measurement equipment after going through the automated calibration process.

Table 1: Power Consumption of XBee-pro s2b in Different Operational Modes at Different Transmit Power Levels



Power Level	Router and Coordinator				End Device			
	Transmitting		Listening		Transmitting		Idle	
	mVosc	mA	mVosc	mA	mVosc	mA	mVosc	mA
P4	172	95	92	51	160	88	20	12
P3	168	93	92	51	152	84	20	12
P2	156	91	92	51	148	82	20	12
P1	152	84	92	51	144	80	20	12
P0	148	82	92	51	144	80	20	12

Table 2: Measured Coverage Range of XBee-Pro S2B

Transmission power level	Indoor Coverage range	Outdoor coverage range
P0	35m	160m
P4	44m	220m

For example the calibrated motes could be used to perform path loss measurements as a function of frequency in order to quantify the likely benefits of applying frequency diversity techniques.

REFERENCES

1. Abrach H, Bhatti S, Carlson J, Dai H, Rose J, Sheth A, Shucker B, Deng J, and Han R (2003), "MANTIS: System support for multimodal N etworks of in-situ sensors", in Proceedings of the 2nd ACM international conference on Wireless Sensor Networks and Applications (WSNA), pp. 50-59.
2. Akyildiz I F, Su W, Sankarasubramaniam Y, and Cayirci E (1999), "Wireless sensor networks: a survey 2002", *Computer Networks,* (Amsterdam, Netherlands), Vol. 38, No. 4, pp. 393-422.
3. Gay D, Levis P, von Behren R, Welsh M, Brewer E, and Culler D (June 2003), "The nesC language: A holistic approach to networked embedded systems", in Proceedings of Programming Language Design and Implementation (PLDI).
4. Han C C, Rengaswamy R K, Shea R, Kohler E, and Srivastava M (2005), "SOS: A dynamic operating system for sensor networks", in Proceedings of the Third International Conference on Mobile Systems, Applications, and Services (Mobisys).
5. Hill J, Szewczyk R, Woo A, Hollar, S, Culler D E and Pister K S J (2000), "System Architecture Directions for Networks Sensors", in *Architecture Support for Programming Languages and Operating System*, pp. 93-104.
6. Kernighan B W and Ritchie D M (1988), *The C Programming Language*, Second Edition, Prentice Hall.
7. Lau B and Lyons M (2007), *Lazy calibration for wireless sensor networks.*
8. Levis P, Gay D, Handziski V, Hauer J H, Greenstein B, Turon M, Hui J, Sharp C, Szewczyk R, and others (2005), "T2: A second gener ation os for embedded sensor networks", Telecommunication Networks Group, Technische Universitat Berlin, Tech. Rep. TKN-05-007.
9. Mini-Circuits: Power Splitter/Combiner: ZN2PD2-50-SMA, Mini-Circuits.
10. MS2721A Spectrum Analyzer Programming Manual, Anritsu Company.
11. Romer K and Mattern F (2004), "The design space of w ireless sensor networks", *IEEE Wireless Communications*', Vol. 11, No. 6, pp. 54-61.
12. Saha S and Bajcsy P (2003), "System design issues fo r applications using wireless sensor networks", Automated Learning Group, National Centre for Supercomputing Applications, Tech. Rep. Technical Report alg03.
13. Whitehouse K and Culler D (2002), "Calibration as parameter estimation in sensor networks", in Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications, pp. 59-67.
14. Yao Y and Gehrke J (2002), "The cougar approach to i n-network query processing in sensor Networks".

IJERST

International Journal of Engineering Research and Science & Technology
Hyderabad, INDIA. Ph: +91-09441351700, 09059645577
E-mail: editorijlerst@gmail.com or editor@ijerst.com
Website: www.ijerst.com

ISSN 2319-5991
02
9 772319 599001