# Wireless Sensor Networks' Node Supported with Remote Hardware Modifications Capability

Ali El Kateeb
University of Michigan-Dearborn
*Email: elkateeb@umich.edu*

**Abstract**

Wireless Sensor Networks (WSNs) nodes are generally have simple hardware design that can be classified as inflexible, support one or more applications, and inexpensive. These nodes are designed and programmed before their release on the application field and will stay on the field to support the need of the target application.

Although the existing wireless sensor nodes are designed to support remote software modifications, they are, however, are not supporting remote hardware modifications. Using wireless sensor nodes that can support remote nodes' modification capability for both hardware and software will provide WSNs with flexible infrastructures that can support nodes with over-the-air design modification even after the deployment of WSNs on the sensing field.

In this paper, we are presenting the design concept and challenges of such infrastructure. Also, we present the use of such flexible infrastructure in potential WSNs applications.

**Key Words:** WSNs, Node design, FPGA

*****

## I. INTRODUCTION

The Wireless Sensor Networks (WSNs) are generally use many tiny nodes that integrated sensing, computing, communication units, and may use some other application-dependent units such as power generators and location finding units [1-3]. These nodes are characterized as low-cost, low-power, and have multifunctional sensors. These WSNs offer economically viable solutions for applications that can be found in different settings such as industry, military, environment, health, etc. [4, 5]. Currently, the sensor nodes are designed and programmed to match the needs of the target applications before they are released to the field. We present flexible WSN node that will be capable of supporting remote hardware modification in addition to software reprogramming feature. This type of nodes will be called "RH-node" throughout this paper. This type of nodes is not available in any commercially available of WSNs infrastructure. Therefore, it is crucial to explore the design challenges and the use of such nodes in next-generation WSNs, which require on-field system modifications. The design and the development of the proposed RH-node will provide WSNs with dynamically adaptable feature. Such feature will provide nodes with design adjusting capability to the surrounding environment even after the deployment of the RH-nodes on the sensing field.

Supporting the proposed RH-node design with remote hardware modifications capability will require the use of Field Programmable Gate Array (FPGA) technology in WSNs' nodes design. However, nodes that currently used in WSNs have not supported with FPGA technology in their designs due to the main drawback of the FPGA, which is high power consumption. However, such drawback can be overcome by using variety of different design approaches such as using modern FPGA chips that consume low power, supporting nodes with solar panel to recharge node batteries, and the use of partial reconfigurations technique that available in some FPGA design tools such as Xilinx.

In addition, the use of partial reconfiguration in FPGA design can reduce the bit file size of the target design, which will result in supporting nodes' with fast hardware modification and at low power consumption. Despite the use of low power FPGA technology and partial reconfiguration technique, the use of FPGA in nodes design will still consume more power than those nodes designed using microcontrollers. The availability and low-cost tiny solar units have made these units useful to be used to provide recharging batteries of the RH-nodes. This will provide long operation time and elevate the drawback for the large power consumption of using FPGA in the proposed infrastructure. Typically, increasing nodes' cost due to the use of solar units will not be acceptable in low-cost WSNs applications. Nevertheless, nodes increasing cost is acceptable when they are deployed in some WSNs applications, as in forest fire monitoring application, as will be discussed later on in this paper.

## II. RH-NODE'S DESIGN CONCEPT AND CHALLENGES

Sensor nodes in some existing WSNs have the support for software reprogramming capability. For example, the SOS project at UCLA provides software modifying on individual nodes of a sensor network after the network has been deployed and initialized. The incremental update ability for new software modules and

removed unused software module after network deployment have been implemented in SOS project [6]. Also, the lightweight virtual machines such as Mate and modular operating systems such as Contiki are using incremental code updates.

The design of the RH-node is focused on supporting WSNs with node's remote hardware modifications capability, which is a new feature for WSNs and has not been investigated thoroughly yet [7]. The main design challenges of supporting RH-node with remote hardware modifications are as follows:

i) **Hardware Technology**: The Field Programmable Gate Array (FPGA) technology can provide very fast design modifications through the use of the computers' USB. The USB cable can be used to download the design to program or reprogram the FPGA chip. However, the remote FPGA design modification has not been provided yet with any FPGA design tools. Therefore, it is required to develop a design tools for remote FPGA design modifications to enhance the use of existing FPGA technology. As the JTAG protocol is used to modify the FPGA design, it will be required to develop a remote JTAG unit that can be used to modify the designs implemented in the FPGA chips. The FPGA technology can be considered as the most effective technology that can support fast and efficeint hardware design modifications.

ii) **Minimizing RH-node's power consumption**: One of the main characteristics of the WSNs is the use of low power nodes. Typically, nodes should be capable to work for a year using an AA battery. However, the RH-node infrastructure design has to use FPGA technology to support hardware modifications, which is generally a power hungry technology. The RH-node's power consumption can be reduced by using some of the recently developed FPGA chips that consume low power such as Actel Igloo, Xilinx Spartan 6, etc. In addition, using partial reconfiguration technique supported by some FPGA design tools, such as with using Xilinx PlanAhead, could reduce the size of the FPGA design file, which significantly reduces power consumption. Transferring design file is usually consume large power, for instance it is known that the power consumption of transferring one bit is equal to the execution of 1000 instructions [8].

To elevate the FPGA high power consumption problem, the use of a large battery unit may support node operation for extended period of time. However, the battery charge will eventually be depleted when a node serves the target application for long period of time. We believe that using a solar-based charging battery unit in the sensor node design can be used to extend the RH-node's batteries life even when the FPGA is used in node design. As the solar panel used for charging battery of WSNs nodes are small in size and inexpensive, it is feasible and acceptable for using these units for recharging the RH-node batteries. When nodes are placed in the application field and are exposed to Sun light, nodes' operation time can vary and depends on the existing sun light period during the day time.

iii) **Wireless JTAG:** The JTAG port supported with FPGA chips is commonly used for hardware modifications. Typically, the design is downloaded from a computer to FPGA chip using the JTAG port. To remotely modify nodes' design on the application field, a wireless JTAG unit needs to be used to receive the design file and deliver it to the JTAG port of the FPGA chip of the RH-nodes. However, since the wireless JTAG unit is currently not available as a commercial product; therefore, such unit should be designed and developed to support remote hardware modification for RH-nodes.

iv) **Library development for system design components**: The infrastructure of the RH-nodes is intended to be a flexible in order to support the needs of the different requirements required by the target applications. Therefore, a library of components is required to be developed to help design and develop RH-nodes quickly and efficiently and to fulfill the applications' needs. Components that useful to be supported in such library are those that can be used in different applications such as soft-core processor component, wireless transceiver interface, and JTAG controller. In addition, sensors interfaces such as CMOS pixels, temperature, and humidity sensors are important to be included in the library since such sensors are used in many WSNs applications. As the FPGA is the technology that will be used in RH-node design, these components will be developed using VHDL language and synthesized using the FPGA design tools.

## III. RH-NODE DEVELOPMENT

The System-On-Chip (SOC) design approach is used for RH-node development where components have integrated on FPGA processing core of the RH-node. The RH-node design has included different soft components such as MIPS processor core, sensor interfaces, JTAG controller, memory interface, and wireless transceiver controller. In addition to these soft components, other off-the-shelf components such as the temperature/humidity sensor, CMOS pixels sensor, memory, wireless transceiver, solar unit, and node's rechargeable batteries have been integrated with the soft components and placed on the RH-node's PCB board [Figure 1].

 ISSN: 2321-8169
1979– 1984

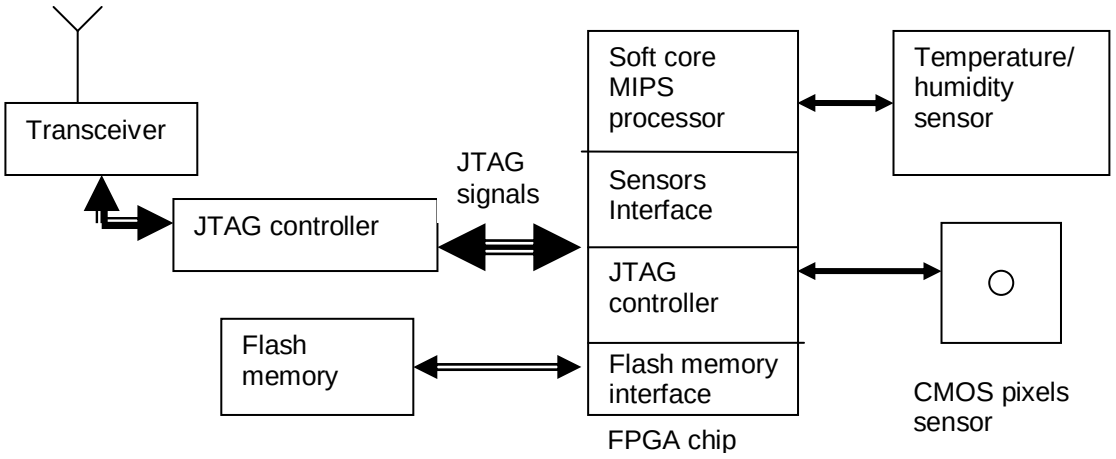


Figure 1: components used for RH-node design

The main components of the RH-node are as follow:

### 3.1. Soft-core MIPS processor

Existing WSNs nodes are generally using standard processor core for their processing need. The ATmega128 micro-controller can be considered the most commonly used processing element for sensing nodes. However, the ATmega128 uses an 8-bit architecture and is thus cannot be as efficient as 16 and 32-bit architectures. Generally, the processing cores used in existing WSNs nodes are off-the-shelf custom made processors that provide reprogramming capability and have no support for remote hardware modification capability.

Some soft-core processors are currently available such as Microblaze, ARM Cortex, etc. Using such cores in sensor nodes can be useful in supporting a variety of applications since they are designed to serve general purpose computations. However, using these processing cores has a drawback in which they have many resources that may not be required for the target application. Therefore, they will consume more power and their cost is high. Design a special soft-core processor for WSNs nodes will provide some advantages such as optimized, flexible, and efficient processor that can match the RH-nodes need of the target WSNs applications. Moreover, using FPGA technology for the soft-core processor development used for the RH-node is useful to support remote node hardware modifications on the applications field.

We have used non-pipeline MIPS architecture for the soft-core processor since it has a small; simple; and efficient processing architecture, which can be integrated in a low-power FPGA chip. A specialized 32-bit non-pipelined MIPS soft-core processor has been developed for RH-node [Figure 2]. The developed soft-core processor has a performance of 10 MIPS, which is acceptable for many sensor Network applications [9, 10]. Moreover, developing MIPS pipelined core can be used in next-generation WSNs applications to support higher processing performance, if required by specific application.

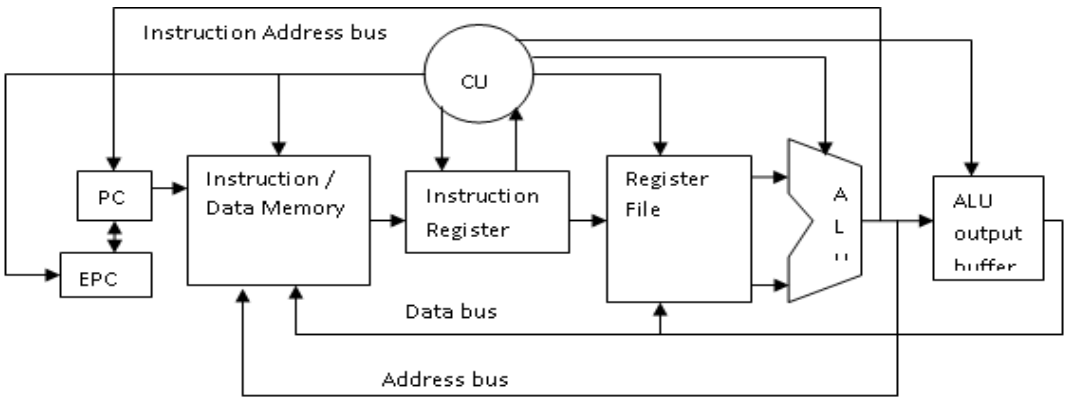


Figure 2: Soft-core processor architecture developed for RH-node

### 3.2 Wireless JTAG unit

Most FPGA chips use a JTAG port to download the design file to the target FPGA chip. As the RH-nodes will support remote hardware modifications capability on the application field, a wireless JTAG unit is required to be used with each RH-node. Developing a wireless JTAG port is essential since no commercially available wireless JTAG that can be used to download design file to the target FPGA chip used in RH-node. The JTAG unit design can be designed using a wireless transceiver and Complex Programmable Logic Device (CPLD) based controller to generate the signals and timing for the wireless JTAG port.

The design file size produced by the Xilinx design tools has a default size of 344,468 bytes. However, using the file compression option that supported with Xilinx design tools, the size of the design file can be reduced significantly as shown in Table 1. Therefore, the remote hardware modifications can be done faster. For example, the design file size that support 16-bit and 32-bit of two CPU components,namely ALU and Register File (RF), to support heterogeneous RH-nodes using Spartan chips shown in Table 1. The time required for achieving remote hardware modifications for these two components will depend on the transfer time of the design file. For example, when we have used 56Kbit/sec transceiver, the 16-bit ALU is remotely modified to 32-bit has required 6.1 second using XA3S200A chip. This time requirement is achieved by segmented the design file into packets and the packet transferred using a data packet frame format, which has 8-bit header and 8-bit trailer, 8-byte size of data, and a CRC field. The transmission range for the used transceiver was 100-feet. Using faster transceiver will help to achieve RH-node design modifications faster. This transmission range is acceptable for some applications where the sensing area is small and few sensor nodes are required to monitor that area.

| CLASS | 16 bit, 16 register | | 32 bit, 32 register | |
|---|---|---|---|---|
| SPARTAN 3A | ALU | RF | ALU | RF |
| Device:XA3S200A | 28KB | 100KB | 81KB | 116KB |
| Device:XA3S400A | 33KB | 145KB | 117KB | 174KB |
| Device:XA3S700A | 40KB | 54KB | 43KB | 281KB |

Table 1: ALU and RF design file size

### 3.3. Sensor interfaces

There are many off-the-shelf sensors available today. However, only those sensors required for serving the target application are placed on the PCB board of the proposed RH-nodes. Off-the-shelf sensors, such as CMOS pixels and temperature/humidity sensors have been utilized by the RH-node to provide a proof-of-concept for using remote hardware modification capability feature in remotely modify the sensors controllers and interfaces that placed on RH-node.

The VHDL language has been used for soft-core processor core and all other sensors interfaces and controllers. Also, a tiny LCD has been used with the RH-node at the base station since it is useful in some applications where WSNs used for security and monitoring applications where the operators located at the base station site can watch the events around the RH-nodes located at the applications filed where they capture the images and transfer them to the base station for observation and analyzes by the operator. The pixels sensor placed on the PCB of each RH-node can be programmed to capture and transfer the image to the base station. Also, RH-nodes can be programmed to process the image at the application site and analyze the image and only transfer the analyzing results to the base station. However, any one of these two methods can be used by RH-node and that depends on the target application needs.

### 3.4. Recharging the battery unit

Despite the use of low power FPGA chip and partial reconfiguration technique in the design of the RH-node to reduce the power consumption, a small solar unit is used with each node for recharging nodes' batteries. Using a solar unit in a well-setting environment will provide the RH-node with a sustain power. A tiny and low-cost off-the-shelf battery recharging solar unit is available that can recharge one AA battery and cost less than $10 for a unit [Figure 3]. Clearly, the cost of the proposed node will be increased due to the use of the

solar recharging unit. However, such increasing cost will be acceptable for the use in WSNs. The RH-node has been developed using Xilinx Spartan FPGA chip and two solar recharging units to recharge two AA batteries to power the FPGA of the RH-node.

## IV. USING REMOTE HARDWARE/SOFTWARE MODIFICATION IN WSNS

A small size WSNs prototype of ten RH-node is under development to provide a proof-of-concept for using remote hardware modifications and software reprogramming capability of the RH-node in WSN applications [Figure 4]. The remote software reprogramming capability has been used in different WSNs applications. However, the WSNs applications that use remote hardware modifications capability is not fully implemented on any existing WSNs yet and they are still under research. Therefore, we used one possible application, namely the forest fire monitoring, to provide the proof of concept of providing both the remote hardware and software modifications capability in WENs applications.

This forest fire monitoring application could demonstrates the remote modifications of the soft-processing cores of RH-nodes on their application field to match the needs of the surrounding application environment. Each RH-node is supported with three sensors: humidity, temperature, and CMOS pixels. There are different possible scenarios of using the RH-nodes on forest fire monitoring application. However, one possible simple scenario is presented to explore the dynamic modification of the nodes' soft-core processors to support the processing required for each RH-node that deployed on the application field. Simple processing required by all nodes at the start of the WSNs and under normal WSNs operation mode, i.e., no fire is detected. Each RH-node at the WSNs will check the humidity level at surrounding environment. When the humidity drops to a low level, this is either an indication of dry weather or a fire has occurred around certain node's location. The node processing unit for the node that closely located near the fire area will then switch to check the temperature and when the temperature level is reached above a certain high level, the node will switch to scan the captured image of the surrounding forest to verify occurrence of fire, which require the use of most of the soft-core processor resources. In the case that no fire is detected, the node's processing core will be restored to its temperature sensing mode.

While the node located at the fire area is locally processed the captured images to identify the existence of fire, other nodes located distantly from the fire will continue to process the basic and simple operation of checking humidity or temperature. Clearly, the processing required for capturing and analyzing the fire image is more complex than the processing required for analyzing the sensors' reading of humidity and temperature. Therefore, all ALU instructions and soft-processor capabilities will be used by the nodes located near the fire location while other nodes will use fewer ALU instructions and potion of the soft-processor capabilities for processing the humidity and temperature sensing.

Figure 3
Tiny solar unit for recharging WSN node

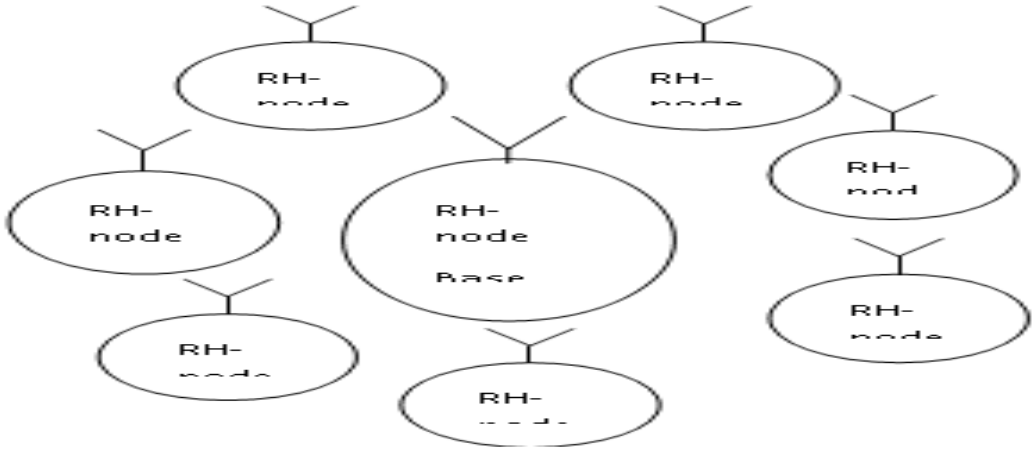


Figure 4

An example of one WSN cluster use RH-nodes

The implementation of the RH-nodes' remote hardware modification capability can be used in this application to save the network power consumption. For instance, the node resources can be set to the level that can be used to conduct the image processing and communicate the result to the base station or to use the node minimum resources that serve the basic node function of checking the surrounding temperature and humidity level. Therefore, the network power consumption will be optimized by using only the node's full resources for processing the image of the fire at the fire surrounding area and transmitting the image to the base station and keep all other nodes located far from the fire location to run using their minimum resources.

## V. CONCLUSION

We presented the research work underway at University of Michigan-Dearborn on the design of the next-generation of WSNs, which use FPGA technology to support remote hardware/software modifications. The concept and the challenges in designing the flexible RH-node has been highlight in this paper. The development of such node infrastructure will advance different research areas of WSNs in applications, architecture, routings, etc. One example of using RH-node's infrastructure in environmental monitoring such as forest fire monitoring application has been discussed in this paper. Using FPGA chips in RH-nodes will consume large power. However, we propose using a low-cost tiny solar unit to recharge the RH-node's batteries. This will provide long operation time and elevate the drawback for the large power consumption of using FPGA in the RH-node design.

## REFERENCES

[1] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks," *Proc. ACM MobilCom'00,* Boston, MA, 2000, pp. 56-67.

[2] G. J. Pottie and W. J. Kaiser, "Wireless Integrated Network Sensors," *Commun. ACM*, vol. 43, no. 5, May 2000, pp. 551-558.

[3] J. M. Kahn, R. H. Katz, and K. S. Pister, "Next Century Challenges: Mobile Networking for Smart Dust," *Proc. ACM MobiCom'99*, Washinton, DC, 1999, pp. 271-278.

[4] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks," *Proc. ACM MobilCom'00,* Boston, MA, 2000, pp. 56-67.

[5] G. J. Pottie and W. J. Kaiser, "Wireless Integrated Network Sensors," *Commun. ACM*, vol. 43, no. 5, May 2000, pp. 551-558.

[6] V. Mudagal, "Methods for Upgrading Reconfigurable Hardware use in Mobile Phones", Freescale Semiconductor, Inc., White paper, 2006.

[7] Ali El Kateeb, "Hardware Reconfiguration Capability for Third-Generation Sensor Nodes", *IEEE Computer*, May 2009, pp. 95-97.

[8] J. Culler, "Incremental Network Programming for Wireless Sensors", *Proc. 1st Annual IEEE ComSoc. Conf. Sensor and Ad Hoc Commun. And Networks (SECON),* 2004, pp.25-33.

[9] Ali El Kateeb, Aiyappa Ramesh, and Lubna Azzawi, " Wireless Sensor Nodes Processor Architecture and Design", *the 22nd IEEE International Conference on Advanced Information Networking and Application*, March 2008, Japan, pp892-897.

[10] Ali Elkateeb, "Remote Modification of Sensor Mote Design for Next-generation Wireless Sensor Networks", *IEEE International Conference on Wireless for Space and Extreme Environment, WiSEE 2013*, November 2013, Baltimore, USA.