A Wireless IP Multisensor Deployment

Diana Bri, Hugo Coll, Miguel Garcia, Jaime Lloret
Integrated Management Coastal Research Institute, Universidad Politécnica de Valencia
Camino Vera, s/n, 46022, Valencia
diabrmo@posgrado.upv.es, hucolfer@posgrado.upv.es, migarpi@posgrado.upv.es, jlloret@dcom.upv.es

***Abstract*— Every sensor node in a Wireless Sensor Network has a microcontroller, a transmitter/receiver and a sensor. It is able to acquire data from specific point in a real environment and transmit it through the Wireless sensor Network. Sometimes it is useful to gather different types of data from the same place in order to obtain a final result. In the related literature, very few works are about sensing different parameters using a unique sensor. In this paper we present a Wireless IP multisensor that is able to gather several types of data from the environment and transmit the result of their combination. Our decision has being mainly based on its development costs, its expansion capacity, the possibilities provided by the operating system, and its flexibility to add more features to the sensor node. We will show all the characteristics of our proposal and the hardware needed for its expansion. Then we will discuss its main application areas. A comparison with many wireless IP sensors offered in the market is also provided.**

Keywords – Wireless multisensor, environment monitoring, Wireless Sensor Deployment.

I. INTRODUCTION

A sensor is any kind of transducer that transforms the magnitude we want to measure in another which is easier to measure. There can be direct indication sensors (e.g., a mercury thermometer) or there can be connected to an indicator (possibly by means of an A/D converter). The measured values have to be read by a human.

Nowadays, a sensor is considered as a basic electronic device for our lives. It covers a wide range of possibilities and applications and, furthermore, it can be used to sense and monitor different parameters according to our necessities. Several aspects can be related with the sensor nodes: a simple physical sensor or a multiple physical sensor, and a wired network of sensors or a wireless network of sensors. We should distinguish between multiple sensors, i.e. many sensor nodes, and a multisensor node, that is able to sense several magnitudes.

Sometimes, it is necessary to control several parameters simultaneously in the same place. The main aim is to have greater control on specific application in order to do these measures more efficient. Therefore, we can use the combination of several physical sensors, forming a multisensor [1]. An obvious example would be the combination of responses from different physical sensors, for determining the trajectory of a wildfire. This combination could be the wind direction, a large and quick increment of the temperature, increment of the CO_2 concentration and a very low value of relative humidity. The right process of this information will report that a fire is happening and what is its direction. This information will allow the firefighters to provide evacuation plans and think on efficient strategies to combat the fire, with the goal of minimizing the loss of forest mass and materials (houses and facilities). If only one sensor was used, the system would not be able to provide whether there is fire, or if it is just a very hot day, or a person who is smoking near the sensor. There are a lot of fields where the use of multisensor can be benefit, such as, home control, building automation, medical applications or robots, among others. It will be later discussed in this paper.

Before we get to combine sensors, we should know how they can be classified. On one hand, many different types of magnitudes can be measured and, on the other hand, there are many types of sensors to measure any physic magnitude. In order to do a study it is needed to classify them according to some criterion. The following classification of sensors has been done considering each sensor like an isolated device which function is to detect sign of qualities or physic phenomenon according to the kind of sensor, and they are converted to useful signals for a measurement or control system:

- According to the energy contribution, sensors can be divided into modulators and generators. In modulators or active sensors, the energy of the output signal almost always comes from an auxiliary energy. However, the output energy in generator or passive sensors is provided by the input. Generally, active sensor circuits require more connexions than passive sensors in order to supply energy. On the other hand, modulator or

2010, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

active sensors' sensibility can be modified with the supplying signal (it can not be modified in generator or passive sensors).

- According to the output signal. They can be analog or digital sensors. The output of the analog sensors is continuous. The information is in the amplitude of the signal. The output of the digital sensors is discrete. Discrete signals are more reliable.
- According to the operation mode. They can be defection or comparison sensors. In defection sensors, the measured magnitude produces a physical effect that causes a similar effect, but opposite, in another part of the instrument, and related with some useful variable. A comparison sensor tries to keep the defection null by means of the application of a well-known effect and it is opposite to the effect generated by the magnitude to measure. Usually, measurements given by comparison are more precise because the opposite known effect is calibrated with a quality's reference magnitude.
- According to the input/output relationship. They can be classified in zero-order, first-order, and second-order or superior order. The order is related with number of independent storage elements of energy. This classification is very important when a sensor is used in a loop control system [2].
- According to the input signal [3]:

  a) *Mechanical:* longitude, area, volume, mass, flux, force, pressure, speed, acceleration, wavelength, position or acoustic intensity.

  b) *Thermal:* temperature, heat, entropy or flux of heat.

  c) *Electric*: voltage, current, resistance, inductance, capacity, load, electric field, frequency, dielectric constant or bipolar moment.

  d) *Magnetic:* intensity of field, density of flux, magnetic moment or permeability.

  e) *Radiation*: intensity, wavelength, polarization, phase, reflectance, transmittance or refraction index.

  f) *Chemical:* concentration, potential redox or PH. Together with electronic sensors, chemical sensors are the most important sensors due to their application fields. These sensors are being used successfully in environments, medicine and industrial processes [4].

In table I, a classification according to three criterions and some examples for each one of them is shown.

However, there are some cases where it is needed to gather several types of measurements, by only one device, in order to obtain more reliability or to obtain specific results. To achieve this goal, it is needed a multisensor.

TABLE I. SENSORS CLASSIFICATION

Criterions	***Types***	***Examples***
Energy contribution	Modulators Generators	Delta-Sigma Thermocouple
Output Signal	Analog Digital	Potentiometer Position coder
Mode of operation	defection comparison	Defection accelerometer Servo-accelerometer

The remainder of this paper is as follows. Section 2 gives all the works we have found related with multisensors. A multisensor is presented in section 3. We will give its main hardware and software features, its implementation, how it can be extended and the operative system used by the device. Section 4 compares the wireless device of the selected hardware with other wireless devices in the market. Some performance test results are shown in Section 5. Section 6 discusses the main application environments where it can be applied. In section 7, we have compared our proposal with many other developed sensor nodes that can be found in the related literature. Section 8 gives the conclusion and future works.

II. RELATED WORKS

In the literature, we can find some works related with multisensors, but almost all of them have been deployed for robots [5] [6].

In 1989, Ren C. Luo and Michael G. Kay [7] introduced the interest and the importance of developing multisensors to increase the capabilities of intelligent systems. The paper describes the multisensor integration and fusion and gives some approaches to the problem. It shows some methods for integrating and fusing multisensory information, to existing multisensor systems used in different areas of application. In 2002, Ren C. Luo et al. extended this information in [8]. They provided an overview of the sensor technologies and described the paradigm of multisensor fusion and integration as well as fusion techniques at different fusion levels. Applications of multisensor fusion in robotics, biomedical system, equipment monitoring, remote sensing, and transportation system were also discussed.

David L. Hall and James Llinas presented a theoretical introduction to multisensor data fusion in [9]. They provided a tutorial on data fusion, introducing data fusion applications, process models, and identification of applicable techniques. Their aim was to show how to fussion sensor measures to obtain a result. They also show a flow chart to explain different

2010, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

manners to interconnect multiple sensors in a single device.

A multisensor example can be found in reference [10]. J. D. Cullen et al. presented a multisensor for online monitoring of the spot welding in automotive industry. The multisensor was able to measure the current, the voltage and the welding force.

Another multisensor example was presented in reference [11]. It was a novel design of the light-addressable potentiometric sensor (LAPS) for realisation of a portable multisensor device. Light sources and electronics including an oscillator, a multiplexer, a pre-amplifier and a high-pass filter were encapsulated in a pen-shaped case, on which the sensor plate was mounted. This sensor device is capable of measuring up to four different ion species by integrating different ion-selective materials on the sensing surface, each illuminated with an independent light source.

E. Kuljanic et al. [12] presented a multisensor approach for chatter detection in milling and compared a single-sensor systems and multisensor systems in terms of accuracy and robustness against malfunctions in order to demonstrate that machines with more sensors give better results for the system. The signal characteristics both in time and frequency domain were condensed into a set of chatter indicators, which were further elaborated by means of statistical basic concepts.

On the other hand, we can find other works where their authors developed a multisensor kernel system, in order to manage the multisensor such as the one presented by T. C. Henderson et al in [13].

No one of the works shown presents a multisensor node that is able to have configured an IP address and join an IP wireless sensor network.

III. MULTISENSOR PROPOSAL

In this section we are going to present a sensor node that is able to sense several parameters from the same place while it is able to form a Wireless IP network of multisensors. The IP protocol is one of the most widely used network interconnection protocol. IP provides a way to transport datagrams from any source to a destination, regardless of whether these machines are in the same network. Moreover, in case of being other networks between the sensors, the IP protocol allows to access them remotely.

In order to achieve our aim, we looked for a device with a control unit. This control unit will manage and control all sensors connected to the device. On the other hand, the electronic circuit must have several input interfaces in order to connect several physical sensors. One of the main aspects taken into account, in order to decide which the best election was, was the circuit costs, its expansion capacity, the possibilities provided by the operative system, and its flexibility to add more features to the sensor node.

Our proposal is based on the use of the Linksys WRT54GL router, from Cisco Systems inc., as the core controller [14]. It is an embedded system that has two serial interfaces in its board, so it meets our pre-requisites. First of all, let's see the router's hardware and software main features [15].

A. *Linksys WRT54GL hardware features*

This sub-section shows the main features of the Linksys WRT54G version 4.0.

The core of this router is a Broadcom BCM5352E processor, working at 200 Mhz. It is a Microprocessor without Interlocked Pipeline Stages (MIPS), which is a RISC microprocessor architecture. It has 256 Bytes prefetch cache, 4 MB INTEL TE28F320 flash memory and 16 MB RAM at 100 MHz clock rate. All the system has a 12 Volts DC power supply.

In order to communicate with other devices, and interconnect them, WRT54GL is equipped with 4 port full-duplex 10/100base TX switch to connect wired Ethernet devices, and a wireless access point, which lets connect both Wireless-G (IEEE 802.11g at 54Mbps) and Wireless-B (802.11b at 11Mbps) devices to the network. The transmitting power for wireless connection is up to 18 dBm. At last, 1 full-duplex 10/100base TX WAN port allows connecting the network to the WAN.

In addition to the mentioned ports, Linksys WRT54GL offers internally GPIO (General Purpose Input/Output), UART (Universal Asynchronous Receiver-Transmitter) and JTAG (Joint Test Action Group) ports. Some extensions can be made to the router by using these ports. Fig. 1 shows the embedded board and its hardware distribution. The UART is signalled as JP2 and the JTAG port is signalled as JP1.

B. *Linksys WRT54GL software features*

One of the main features that have caused the use of the Linksys WRT54GL as a sensor node was the possibility of installing a Linux Kernel 2.4. On one hand, it is a well known operative system, so we didn't need to learn how to work with a new operating system and, on the other hand, we know all the possibilities that a Linux is able to provide us. So, at software level, this model is based on open source software, causing the development of different specific software applications for it and expanding the factory default capabilities. Several of its main software features that can be mounted over this platform are:

- Web server.
- DHCP Client and DHCP Server.

- Telnet server and client.
- SSH server and client.
- RADIUS server.
- WPA/TKIP and AES encryption.
- WPA2 encryption.
- Wireless MAC address filtering.
- Powerful SPI firewall.
- VPN client and server.
- VLAN.
- VoIP switchboard.
- QoS Bandwidth Management.
- Wake On LAN.
- MMC/SD Card Support.
- USB port support.

C. *Implementation*

In order to connect two sensors directly to the board, we made an extension using the GPIO of the Linksys WRT54GL router. It provided us two serial ports through the JP1 port.

Fig. 2 shows a serial port interface connected on board by welding pins on JP1. Then, we added two DB9 Female DCE ports because we wanted flexibility in order to change the type of sensor connected to our device.

Fig. 3 shows the integrated circuit used to provide two serial ports. A RS-232 line converter is needed to go from +3.3V to +5V. In our design we propose this implementation with a MAXIM MAX233 integrated circuit. It contains four sections: dual charge-pump DC-DC voltage converters, RS-232 drivers, RS-232 receivers, and receiver and transmitter enable control inputs

On the other hand, we connected a SD/MMC card reader to the GPIO pins of the CPU found inside the Linksys and with the help of a little driver we can use as a block device from Linux. Compiling the kernel for the Linksys with e.g. support for MSDOS partitions and VFAT we are able to mount, read, write, partition and so on in normal SD and MMC cards. The speed for reading and writing is about 200 KB/s.

The SD/MMC card reader has been tested for a 1GB SD and MMC cards. It allows us to install applications for signal processing and store and manage acquired data from both sensors. Now, the sensor node is able to store and process the data stored without the need of sending the measures from its sensors continuously. The system is now able to gather data, process it internally and send only alarms or statistical data spontaneously to the network, thus saving energy.

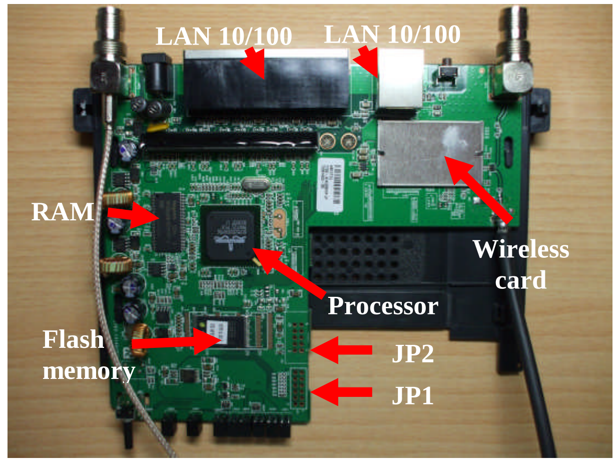

Figure 1. Hardware distribution on board

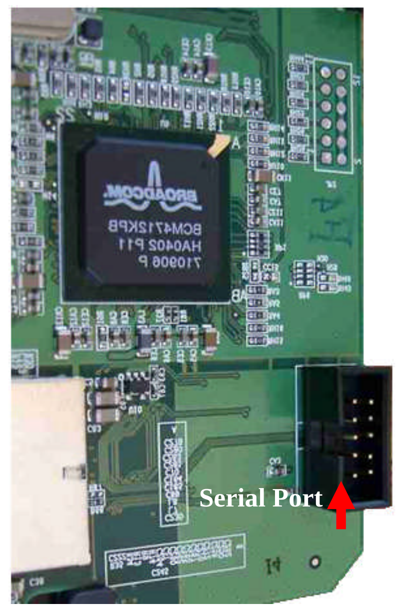

Figure 2. Serial ports connected on board

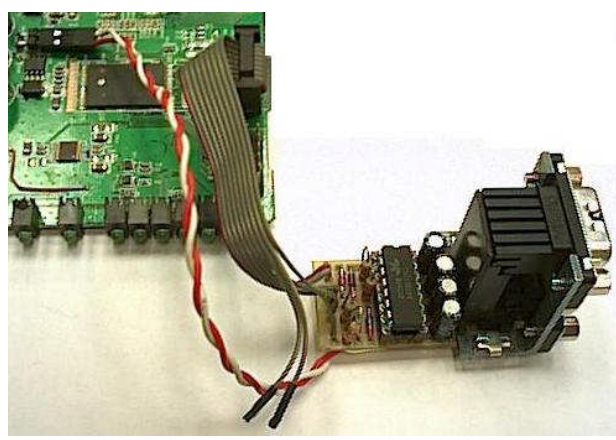

Figure 3. Integrated Circuit used to provide two serial ports.

2010, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

D. Proposal extension

The main drawback of our proposal was that Linksys WRT54GL router only allows two physical sensors because it has only two available serial ports. This limitation would limit us the number of application environments. So, we tried to extend this feature by adding an external hardware. One of them is the use of a JTAG to parallel port interface, and the other is given by the Arduino open hardware platform [16].

There are several different kinds of interfaces for hooking up to the JTAG headers (signalled as JP1 in the Fig. 1). We are going to talk about JTAG to Parallel port interface. It adds a parallel port to the WRT54GL providing another interface to connect more sensors. In the design stage we have two different variants of interfaces, the unbuffered and the buffered one. The first way is the unbuffered one, this is the cheapest and the easiest way to add a parallel port to the WRT54GL, only with 100 ohm resistors connected directly to the specified JTAG pins. The other variant is the buffered one. It is more complex that the unbuffered interface but it is more immune to noise and static. It provides a higher data transfer rate than the unbuffered one. In our proposal we use a PHILLIPS 74HC244 buffer to implement it. The 74HC244 is an octal buffer/line driver, 3-state. The JTAG to parallel port interface provides slow data rates. This is due to the nature of the parallel port, but, these data rates are really enough to the multisensor of our proposal.

Arduino is an open-source electronics prototyping platform based on flexible, easy-to-use hardware and software. It's intended for anyone interested in creating interactive objects or environments. The board is available to buy, but schematics are free distributed in order to let anyone mount its own board. Controlling software can be downloaded for free from the Arduino website. Fig. 4 shows the Arduino board. It is a cheap, robust input and output board based on the ATmega168. It has 13 digital pins (6 of which allow PWM output) and 6 analog inputs. There are many different versions that allow adding USB ports and/or serial ports, Bluetooth connexions and so on.

Arduino can sense the environment by receiving input from a great variety of sensors and can affect its surroundings by controlling lights, motors, and other actuators. Arduino hardware can be stand-alone or can communicate with software on running on a computer. So, according to it, we connected the Arduino board to the WRT54GL by using the serial port on the router board. Now, we have deployed a real multisensor system dotted with multiple in and out ports to connect sensors and any kind of devices. The core controller, placed in the Linksys WRT54GL router, is strong enough to compute all the data received from the Arduino board.

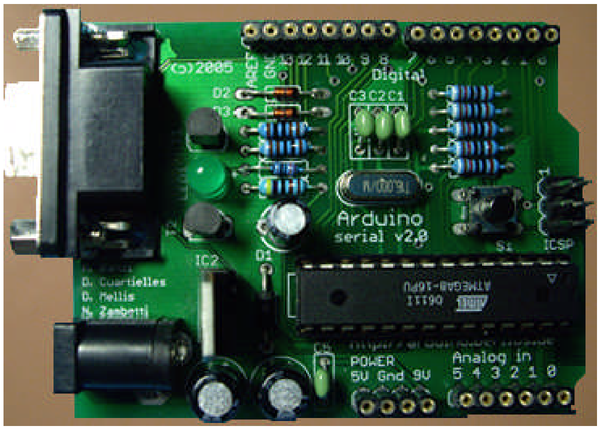

Figure 4. Arduino serial board

Due to the very small size of the Arduino board, it can be allocated inside the WRT54GL cage.

E. Operating system and communications

There are several Linux-based distributions built for the Linksys WRT54GL router, with a high variety of functionalities. The main distributions are Wrt54g-linux [17], BatBox [18], OpenWRT [19] and DD-WRT [20]. Wrt54g-linux and BatBox are loaded on RAM, so they have very light and have limited functionalities. Moreover, they are not stored in the router after it has been reset. Those are the reasons why both operative systems have not been considered to be used in our proposal.

We have chosen DD-WRT for our implementation due to the high quantity of features included and the open source community support. In order to change the operative system, first of all we have to erase the flash memory. Then, we can install DD-WRT operating system. This operation can be done by Web GUI, TFTP or just typing directly in command line. The final result is the same, so there is no matter the way chosen for it. After the operative system is changed, we have a Linux inside the device and we can work with it as with a regular Linux.

Once the operating system is installed, the following step (at software level) is to communicate the Arduino board with the sensors connected to its ports. The Arduino programming language is based on C/C++, so data can be sent to the sensors or received from them by using serialWrite() and serialRead() functions (when devices are connected to the serial ports). Also speed can be modified by typing begin Serial (speed) [21]. Depending on the device connected to the board and the kind of port used to communicate, the C/C++ code of the communication script could be a bit different. The procedure will be to change references to pins involved in the transmission and reception and the specific commands for each port connection.

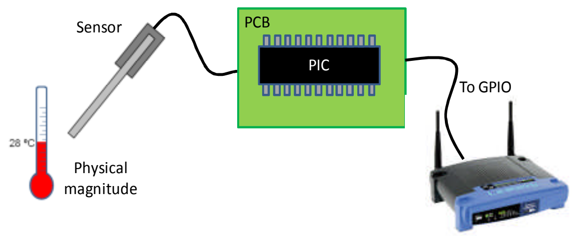

Figure 5. Block diagram

Keeping in mind that we can install an open source Operative System and the GPIO connections which can be found on the Linksys WRT54GL, we can program a script that allows us to take data from the sensors to be processed by our multisensor device. To this end, we design a small electronic circuit that works as an interface between the sensor and multisensor device, in order to adapt the sensor signal to the type of signals that GPIO terminals support. Fig. 5 shows a block diagram with the proposed system. This proposal allows connecting other sensors with a higher bit rate. The sensors, whose interface is based on RS-232 standard, can have bit rates up to 128.000 bit/s. SO, it would enable us to reach speeds up to 200 KBytes/s, namely, the read and write speed of a SD card.

IV. Wireless interface comparison

We can find several wireless interfaces in the market that incorporate the IEEE 802.11 standard and can be used for a multisensor node. They are low energy consumption solutions. At the same time, these devices integrate great variety of resources and functions. Their sizes are quite small; they can be built-in inside portable devices. Because their low power consumption, they can also be fed by batteries. All these integrated circuits are also designed to enable the communication between wireless devices and a serial interface, so a sensor can be connected to it. In this sub-section we will analyze only the chipsets that have their power consumption specifications published. There are other IEEE 802.11 compliant chipsets in the market but they have not embedded serial interfaces. Table II shows some of these devices. n/a means that this information is not provided by the manufacturer.

Most of these devices (those whose price is below $ 300) need a development kit in order to use the device. The price of these kits could be $ 150-200 and can reach to $ 1,480 in some cases. The sizes of these evaluation kits are roughly similar in size to Linksys WRT54GL. We choose the Linksys, because this device offers us the same features than the others, but it is cheapest and allows us to add a Linux-based Operative System based to program and run applications in order to process information gathered by the sensors. Furthermore, this device has the faster microprocessor of the comparison table.

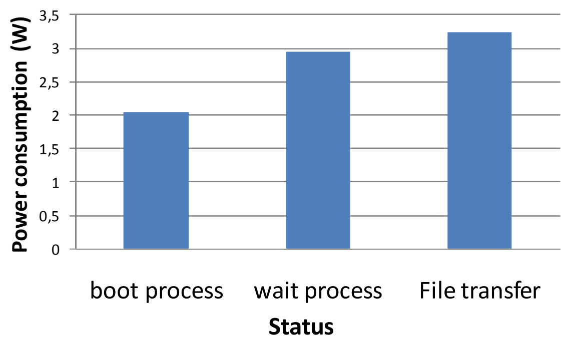

Figure 6. Energy consumption

V. Performance Tests

In this section we can see the tests made with the device. There have been several testbenches.

In the first one we measured the energy consumption in each of the operating modes of the IP multisensor. The second test measured the typical parameters of the network. The tests were performed under the 802.11 b technology.

A. *Energy consumption*

In order to perform the power consumption tests, a power supply set to 12V (DC) and limited to 1000mA was used. It has also been necessary to use a wattmeter to measure the power consumed in each case.

First, we measured the energy consumed in the boot process of the device. This process takes approximately 15 seconds. Then, we measured the energy obtained when the device is transmitting and receiving information and, finally, the energy when the device is waiting to receive data. Fig. 6 shows the results.

One of the major requirements in WSN is to have very low energy consumption. It is known that IEEE 802.11 protocols are inadequate for energy constrained devices.

Therefore, the operation and the utilities of the different hardware parts of the Linksys WRT54GL have been studied. We have determined that there are some parts that could be unused in our sensor implementation, getting so, that these components will not consume energy if we remove them. For example, the WAN port and 3 of the 4 LAN ports can be removed and with them, all of their passive components that consume energy (such as resistors and diodes). Moreover, the integrated circuits which are not used for sending information via wireless can be also removed. With these modifications, we would reduce the global system consumption of about 15%.

TABLE II. SENSORS NODES COMPARISON

2010, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Device	Manufacturer	Current consumption in TX	Current consumption in Rx	Current consumption in Standby	Core CPU	Operating voltage	IEEE Especification	Size in (mm x mm)	Price in $
Nano WiReach [22]	Connect One Ltd	250 mA	190 mA	8 mA	32-bit RISC ARM7TDMI, low-leakage, 0.13 micron, running at 48MHz	3.3 V	802.11 b 802.11 g	17 x 33	545.75
Mini Socket iWiFi [23]	Connect One Ltd	250 mA	190 mA	8 mA	32-bit RISC ARM7TDMI, low-leakage, 0.13 micron, running at 48MHz	3.3 V	802.11 b 802.11 g	41 x 31	545.75
S103 WLAN Compact Serial Module [24]	Rf solutions	325 mA	210 mA	80 mA	n/a	5 V	802.11 b	40 x 60	75
"WiFly" 802.11B Module (RN-111B) [25]	Roving networks	110 mA	40 mA	35 mA	n/a	3.3 V	802.11 b	29 x 50	69
"WiFly GSX" 802.11G Module (RN-131G) [26]	Roving networks	210 mA	40 mA	15 mA	n/a	3.3 V	802.11 b 802.11 g	38 x 26	45
WiFly GSX Super Module "SuRF Board" (RN-134) [27]	Roving networks	212 mA	50 mA	15 mA	n/a	3.3 V	802.11 b 802.11 g	17 x 33	99
MatchPort b/g [28]	Lantronix	360 mA	225 mA	76 mA	Lantronix DSTni-EX x86 CPU, on-chip 256 KB zero wait static SRAM, 2,048 KB Flash, 16 KB Boot ROM, 8 GPIO	3.3 V	802.11 b 802.11 g	29 x 50	86.46
WiPort [29]	Lantronix	650 mA	395 mA	91 mA	Lantronix DSTni-EX 186 CPU, on-chip 256 KB zero wait static SRAM, 2,048 KB Flash, 16 KB Boot ROM,	3.3 V	802.11 b 802.11 g	27 x 51	139.66
MatchPort® b/g Pro [30]	Lantronix	350 mA	260 mA	160 mA	Lantronix 32-bit processor, 166Mhz (159 MIPS - Dhrystone 2.1)	3.3 V	802.11 b 802.11 g	45 x 45	210
RCM4400W RabbitCore [31]	RABBIT	450 mA	450 mA	80 mA	Rabbit 4000 Microprocessors (60 MHz)	3.3 V	802.11 b	13 x 14	131.38
Connect Wi-ME [32]	DigiBoard	346 mA	186 mA	34 mA	32-bit, 55 MHz NS7520 processor ARM7TDMI ARM core	3.3 V	802.11 b	80 x 110	410.41
ConnectCore Wi-9C [33]	DigiBoard	900 mA	700 mA	10 mA	32-bit, 155 MHz NS9360 processor ARM926EJ-S ARM core	3.3 V	802.11 b 802.11 g	91 x78	379
WRT45GL [14]	Linksys by Cisco	270mA	270mA	245mA (regular mode)	Broadcom BCM5352E processor, (200 Mhz). 16 MB RAM	12V	802.11 b 802.11 g	186 x 154	69.99

2010, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

B. *Measures of the wireless network parameters*

In order to show the network parameters we measured the worst case, that is, when the sensor node gathers the data from the physical sensor, process it, and sends it to another device using the wireless network. The data collected by the physical sensor could be very different. It can be a temperature value which packet size will not reach more than a few bytes. But the physical sensor could be a video images sensor, which can reach sizes of several MBytes.

We used two identical wireless IP sensors to perform our test bench. They had the same hardware and software described in the previous sections. In order to see what happens in the worst case, we stored a large file in a single sensor which contains different types of data obtained from each physical sensor connected to the multisensor.

Thus, we transmitted a large file from sensor 1 to sensor 2, under the 802.11b technology. The file size was 210 MBytes and the transfer time was approximately 15 minutes.

We measured several parameters in order to know the network performance in such conditions. These parameters were the number of errors, drops, collisions, broadcast traffic, octets, packets and utilization (%). In the first three parameters we obtained a value of zero, because in this test, we only had two sensor nodes one transmitting and one receiving, so there were not errors, drops, collisions during our experiment.

Fig. 7 shows the broadcast packets sent in the experiment. There has not been more than one packet per second. There have been only 7 broadcasts during the 15 minutes (an average value of 0.009).

Fig.8 shows the number of packets per second along the time. An average value of 192 packets per second has been obtained. The maximum number of packets per second was 340.

Fig. 9 shows the number of bytes per second that have been sent over the network. There has been a maximum value of 336575 Bytes per second (in a theoretical channel of 11Mbps). The mean value during the 15 minutes has been 264680.13 Bytes.

Fig. 10 shows the utilization se of the bandwidth of the channel in %. It has been estimated that the system has 95% of channel utilization. The 77.42% of the time the wireless channel had 100% of utilization.

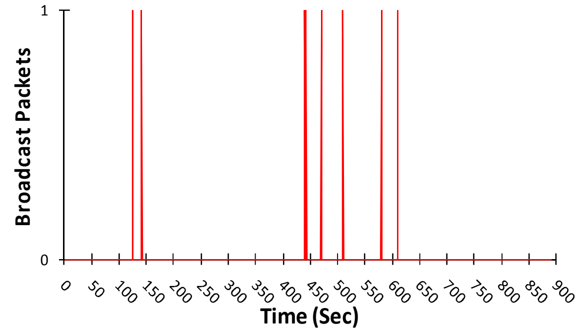

Figure 7. Broadcast packets

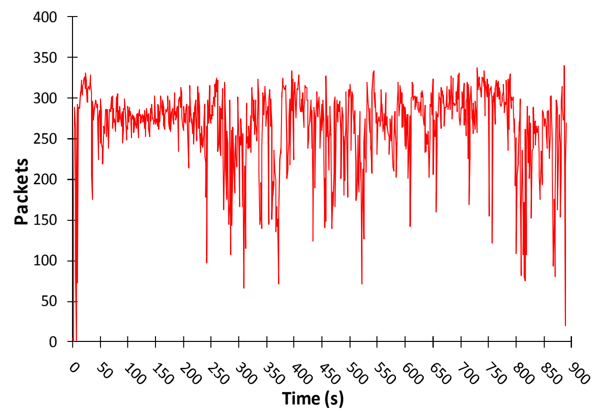

Figure 8. Packets per second.

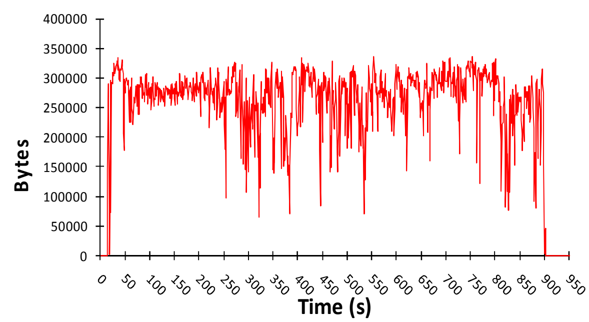

Figure 9. Bytes per second.

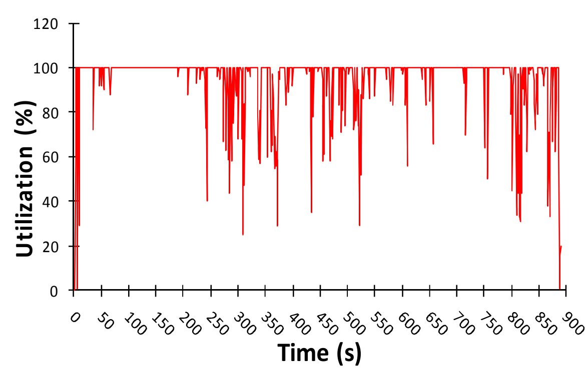

Figure 10. Utilization in %.

VI. Application environments

The presented model can be applied to any environment that needs to be sensed by several types of variables. It is also able to process internally the measures taken from the connected sensors and send the processed information to a remote site. It is flexible and it could be adapted to any type of environment and to any type of sensor with a serial output. The change of the programming code is only needed to adapt the control management to different sensors.

There are more needs multisensor nodes than it is expected. The following list gives some examples:

- *Home Control:* There are several possible home applications where a multisensor node can be used. E.g. in automate control of multiple home systems to improve conservation, convenience, and safety. A multisensor node on the wireless sensor network can check some parameters (temperature, light, humidity, etc) continually in order to make flexible management of lighting, heating, and cooling systems from anywhere in the home. Another home application could be a sensing application that captures highly detailed electric, water, and gas utility usage data. With these types of applications one user could control his/her home expenses easily without the need of place many sensors inside the house.
- *Building Automation:* A multisensor node can be used for multiples purposes in building control. A multisensor could measure some parameters simultaneously as light, switches on, etc. in order to enable the rapid reconfiguring of lighting systems to create adaptable workspaces. The combination of the parameters measured could give better decisions. Also, the wireless sensor network, with multisensors, could be used for building energy monitoring and control. They could improve living conditions for the building's occupants, resulting in improved thermal comfort, improved air quality, health, safety, and productivity.
- *Industrial automation:* Industrial automation applications provide control, conservation, efficiency, and safety. E.g. a multisensor can provide the sound level and the temperature of machines for monitoring the proper functioning of a factory's area. The multisensor can take decisions, when both measures exceed the threshold, before sending an alert to the central site. Moreover the sensing applications using multisensor devices reduce energy costs through optimized manufacturing processes.
- *Medical applications:* A number of hospitals and medical centres are exploring applications of wireless sensor networks technology to a wide range of medical applications, including pre-hospital and in-hospital emergency care, disaster response, and stroke patient rehabilitation. These systems can use multisensor devices to collect several data from the patients simultaneously, such as body temperature, heart rate, etc. This information will be sent to a local central server and to the hospital.
- *Robots:* This is the most common application environment. Robots need to combine multiple sensed measurements in order to take a decision (to walk, to avoid an obstacle, to take something from the floor, and so on). One of the major issues in a mobile robot acting as a gateway is the communication between the robot and the sensor network. Using our multisensor node this issue will be easily arranged, especially if the robot uses serial communication, because this node can manage several serial ports.
- *Habitat monitoring:* Sensor networks represent a significant advance over traditional invasive methods of monitoring. Sensors can be deployed prior to the onset of the breeding season or other sensitive period (in the case of animals) or while plants are dormant or the ground is frozen (in the case of botanical studies). Using a multisensor node we can obtain several parameters in a single device. A multisensor will allow building parallel specialized networks [34]. Thus, we have a system that collects lots of data without providing high visual impact (an important factor in environmental settings).
- *Fire detection and monitoring [35]:* In order to obtain a true fire alarm, it is better to develop a multisensor which combines several sensor measurements such as temperature variation, the humidity, wind direction and CO_2 level. The multisensor can check the history of the sensor measurements and process the data in order to give statistical results and combine the sensor values thresholds. The result will be more precise than a single sensor. Currently, there are many critical areas that may be at risk of fire. These multisensor systems could be used to control the zones, so we have a simple system based on wireless sensor networks to monitor fires.
- *Traffic monitoring:* Traffic flow is a sector that is expected to benefit from increased monitoring and surveillance. These systems use multisensor nodes to collect data of many parameters. The systems are installed along major highways; the digital sensor network gathers lane-by-lane data on travel speeds, lane occupancy, and vehicle counts. These basic data elements make possible to calculate average speeds and travel times.

2010, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

- *Military applications:* For military users, an application focused on wireless sensor networks technology has been area and theater monitoring. Wireless Sensor Networks can replace single high-cost sensor assets with large arrays of distributed sensors for both security and surveillance applications. The Wireless Sensor nodes are smaller and more capable than sensor assets presently in the inventory; the added feature of robust, self-organizing networking makes Wireless Sensor Networks deployable by untrained troops in essentially any situation. A multisensor could improve the system being able to sense environment parameters while detecting the presence of other humans. It can also inform about the combination of the parameters sensed.
- *Indoor people location and behaviour:* In this case, for example, the multisensor could be used in order to know the walking direction of a person in a corridor. It could be easily known if the wind direction is combined with a presence detection sensor. It will allow us to know if a person is coming in or leaving a zone. If we place just one of those sensors it will not give us the right result, because it could be open windows and the wind could cheat us. This system can be also used for intrusion detection, as a visitor counter or to be sure that all people have left a building.
- *To measure the optimum water of a pool in a spa:* There has to be a sensor able to measure the optimum measures of the temperature, the chrome, the water pH and the aroma. In large pools the nodes of the wireless sensor network could be able to sense multiple parameters simultaneously. It will let us have a global view of the state of the pool.
- *Automobile applications:* A modern automobile has about 8 Km of cables to connect hundreds of sensors. A Wireless Sensor Network allows not only reducing the volume and weight required by the cabling, but also the deployment of sensors with more freedom. A multisensor would be able to measure different parameters simultaneously in an automobile in order to have a distributed network.

There are many more specific applications, but in this section we have discussed the most important areas where they can be used.

VII. WIRELESS SENSOR NODES COMPARISON

Nowadays, many sensor nodes can be bought in the market. In this section, we are going to compare our proposal with the most well known sensor nodes.

Table 2 shows a summary with the name of the sensor, its microcontroller, the type of transceiver or wireless technology, the internal and external memory, and its operative system. We have provided only the public features of the devices, so, some hardware characteristics are not shown because the owner does not provide them. We can observe that the multisensor node is the sensor node with more internal and external memory. Each node has an operating system, TinyOS is the most common, but the operating system used in our sensor node can be modified to be as efficient as TinyOS. Another important feature of all sensor nodes is the use of UNIX-based operating systems.

On the other hand we can see that the sensor node with worst features is the Dot node. This node only has 1 KB of RAM and up to 16 KB of external memory.

If we look the microcontroller feature of the sensor nodes described in Table 3, we can see that there are nodes that have processors working with around 400 MHz. The worst processors have a speed around 10 MHz. In our case, the multisensor presented in this work uses a processor with a speed of 200 MHz. The device has not the most powerful microcontroller, but it has enough power for its proper operation.

Finally, we have to take into account that our proposal is the only one that is able to have 2 physical sensors directly connected at least, but we are supposing that the other deployments are able to be changed to acquire data from 2 or more physical sensors (making some adjustments in the electronic board). We can observe that our proposal has the most powerful processor and the one with most internal and external memory (we can even add 1 GB in a SD card). On the other hand, Linux is the most well known operative system in that table. But as drawbacks, our proposal works with IEEE 802.11 b/g (although we can build our proposal over an IEEE 802.11n device, if it has the same features) and the energy consumption is higher than the others because of its hardware requirements.

VIII. CONCLUSIONS

We have presented a multisensor for Wireless Sensor Networks using IEEE 802.11b/g. It is flexible and it could be adapted to any type of environment and to any type of physical sensor with a serial output. Using the described system with the appropriate Arduino board and code programming for each specific application, we can create a high–complex multisensor system which can be able to sense many physical magnitudes. In future works we will add to our multisensor a JTAG to USB interface to support sensors with the requirement of high bit rates. This is possible to the nature of the JTAG interface that can provide a theoretical data rate of 25 Mbps. We are now

2010, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

studying how to make the Linksys WRT54GL router consume less energy.

2010, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

TABLE III. SENSOR NODES COMPARISON

Sensor Node	Microcontroller	Wireless Transceiver or Wireless Technology	Internal Memory	External Memory	Operative System
Our proposal	Broadcom BCM5352E at 200 Mhz processor	IEEE 802.11g and IEEE 802.11b	16 MB RAM	4 MB Flash	Linux
BEAN [36]	MSP430F169	CC1000 (300-1000 MHz) with 78.6 kbit/s		4 Mbit	YATOS
BTnode [37]	Atmel ATmega 128L (8 MHz @ 8 MIPS)	Chipcon CC1000 (433-915 MHz) and Bluetooth (2.4 GHz)	64+180 K RAM	128K FLASH ROM, 4K EEPROM	BTnut and TinyOS
Dot [38]	ATMEGA163	ChipCon CC1000 916MHz	1K RAM	8-16K Flash	TinyOS
EPIC Mote [39]	Texas Instruments MSP430 microcontroller	250 kbit/s 2.4 GHz IEEE 802.15.4 Chipcon Wireless Transceiver	10k RAM	48k Flash	TinyOS
Eyes [40]	MSP430F149	TR1001		8 Mbit	PeerOS
EyesIFX v1 [40]	MSP430F149	TDA5250 (868 MHz) FSK		8 Mbit	TinyOS Support
EyesIFX v2 [40]	MSP430F149	TDA5250 (868 MHz) FSK		8 Mbit	TinyOS Support
FlatMesh FM2 [41]	16MHz	802.15.4-compliant		660 sensor readings	Commercial system, for digital sensors
Firefly [42]	Atmel ATmega 1281	Chipcon CC2420	8K RAM	128K FLASH ROM, 4K EEPROM	Nano-RK RTOS Support
GW node [43]	PIC18LF8722	BiM (173 MHz) FSK	64k RAM	128k flash	Custom OS
IMote 1.0 [44]	ARM 7TDMI 12-48 MHz	Bluetooth	64K SRAM	512K Flash	TinyOS
Imote 2.0 [44]	Marvell PXA271 ARM 11-400 MHz	TI CC2420 802.15.4/ZigBee compliant radio	32MB SRAM	32MB Flash	Microsoft .NET Micro, Linux, TinyOS Support
Iris Mote [45]	ATmega 128	Atmel AT86RF230 802.15.4/ZigBee compliant radio	8K RAM	128K Flash	TinyOS, MoteWorks Support
Kmote [46]	TI MSP430	250 kbit/s 2.4 GHz IEEE 802.15.4 Chipcon	10k RAM	48k Flash	TinyOS and SOS
Mica2 [47]	ATMEGA 128L	Chipcon 868/916 MHz	4K RAM	128K Flash	TinyOS, SOS and MantisOS
MicaZ [48]	ATMEGA 128	Transceptor 802.15.4/ZigBee	4K RAM	128K Flash	TinyOS, SOS and MantisOS
Mulle [49]	Renesas M16C	Bluetooth 2.0	31K RAM	256K Flash	
Neo Mote [50]	ATmega 128L	TI CC2420 802.15.4/ZigBee compliant radio	4K RAM	128K Flash	TinyOS, SOS, MantisOS, Nano-RK and Xmesh Support, Industrial end-use product.
Redbee [51]	MC13224V	2.4 GHz 802.15.4	96 KB RAM + 120KB Flash		Contiki; standalone
Rene [52]	ATMEL8535	916 MHz with 10 kbit/s	512 K RAM	8K Flash	TinyOS
SenseNode [53]	MSP430F1611	Chipcon CC2420	10K RAM	48K Flash	GenOS and TinyOS Support
SunSPOT [54]	ARM 920T	IEEE 802.15.4	512K RAM	4 MB Flash	Squawk J2ME Virtual Machine
Telos [55]	Motorola HCS08		4K RAM		
Telos B [56]	Texas Instruments MSP430 microcontroller	250 kbit/s 2.4 GHz IEEE 802.15.4 Chipcon Wireless Transceiver	10k RAM	48k Flash	Contiki, TinyOS, SOS and MantisOS Support
Tinynode [57]	Texas Instruments MSP430 microcontroller	Semtech SX1211	8K RAM	512K Flash	TinyOS
T-Mote Sky [58]	Texas Instruments MSP430 microcontroller	250 kbit/s 2.4 GHz IEEE 802.15.4 Chipcon Wireless Transceiver	10k RAM	48k Flash	TinyOS, SOS and MantisOS
WeC [59]	Atmel AVR AT90S2313	RFM TR1000 RF			
XYZ [60]	ML67 series ARM/THUMB microcontroller	CC2420 Zigbee compliant radio from Chipcon	32K RAM	256K Flash	SOS Operating System

2010, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

REFERENCES

[1] Diana Bri, Hugo Coll, Miguel Garcia, Jaime Lloret, A Multisensor Proposal for Wireless Sensor Networks. Second International Conference on Sensor Technologies and Applications (Sensorcomm 2008). Cap Esterel, France, August 25-31, 2008. Pp.270-275

[2] I. R. Sinclair, Sensors and Transducers (Third Edition). Elsevier. UK. December 2000.

[3] H. Pasterkamp, S.S. Kraman, P.D. DeFrain and G.R. Wodicka, "Measurement of respiratory acoustical signals. Comparison of sensors", Chest: Official publication American College of Chest Physicians, Vol. 104, pp. 1518-1525, USA, November 1993.

[4] S. J. Birrell, K. A. Sudduth and S. C. Borgelt. "Comparison of sensors and techniques for crop yield mapping". Computers and Electronics in Agriculture, Vol. 14, Issues 2-3, pp. 215-233, USA, February 1996.

[5] R. C. Luo, M.-H. Lin, and R. S. Scherp, "Dynamic multi-sensor data fusion system for intelligent robots", IEEE J. Robot. Automat., vol. 4, Aug. 1988.

[6] K. Hirai, M. Hirose, Y. Haikawa, and T. Takenaka, "The development of Honda humanoid robot". IEEE Int. Conf. Robot. Automat. Vol. 2. Pp.1321-1326. 1998

[7] R. C. Luo, and M. G. Kay, "Multisensor Integration and Fusion in Intelligent Systems", IEEE Transactions on Systems, Man, and Cybernetics, Vol.19, No. 5, pp. 901-931, September/October 1989.

[8] R. C. Luo, C-C. Yih and K. L. Su, "Multisensor Fusion and Integration: Approaches, Applications, and Future Research Directions", IEEE Sensors journal, Vol. 2, Issue 2. Pp. 107–119. USA. April 2002

[9] D. L. Hall and J. Llinas, "An Introduction to Multisensor Data Fusion", Proceedings of the IEEE, Vol. 85, Issue: 1, pp. 6-23, New York, USA, Jan 1997.

[10] J.D. Cullen, N. Athi, M. Al-Jader, P. Johnson, A.I. Al-Shamma'a and A. Shaw, A.M.A. El-Rasheed. "Multisensor fusion for on line monitoring of the quality of spot welding in automotive industry". Measurement 41 (2008) 412–423.

[11] T. Yoshinobu, M. J. Schöning, R. Otto, K. Furuichi, Yu. Mourzina, Yu. Ermolenko and H. Iwasaki, "Portable light-addressable potentiometric sensor (LAPS) for multisensor applications", Sensors and Actuators B: Chemical, Volume 95, Issues 1-3, pages 352-356, 15 October 2003.

[12] E. Kuljanic, M. Sortino and G. Totis. "Multisensor approaches for chatter detection in milling". Journal of Sound and Vibration 312 (2008) 672–693.

[13] T. C. Henderson. W. S. Fai, and C. Hansen, "MKS: A multisensor kernel system". IEEE Transactions on Systems, Man and Cybernetics, Vol. SMC-14, No 5. Pp. 784-791, 1984.

[14] Cisco Systems Inc. At http://www.cisco.com [Last Access 7th June 2010]

[15] Paul Asadoorian and Larry Pesce, Linksys WRT54g Ultimate Hacking. Elsevier. USA. June 2007.

[16] Arduino Web page. At http://www.arduino.cc [Last Access 7th June 2010]

[17] Wrt54g-linux web page. At http://freshmeat.net/projects/wrt54g [Last Access 7th June 2010]

[18] BatBox web Page. At www.batbox.org/wrt54g-linux.html [Last Access 7th June 2010]

[19] Open WRT web page. At http://openwrt.org/ [Last Access 7th June 2010]

[20] DD-WRT web page. At http://www.dd-wrt.com [Last Access 7th June 2010]

[21] Nano WiReach specifications, Available at: http://www.connectone.com/media/upload/Nano_WiReach_PB.pdf [Last Access 17th June 2010]

[22] Mini Socket iWiFi, specifications, Available at: http://www.connectone.com/media/upload/Mini_Socket_iWiFi_DS.pdf [Last Access 17th June 2010]

[23] S103 WLAN Compact Serial Module specifications, Available at: http://www.rfsolutions.co.uk/acatalog/DS-S103.pdf [Last Access 17th June 2010]

[24] RN-111B specifications, Available at: http://www.rovingnetworks.com/documents/rn-111b-ds.pdf [Last Access 17th June 2010]

[25] RN-131G specifications, Available at: http://www.rovingnetworks.com/documents/rn-131-ds.pdf [Last Access 17th June 2010]

[26] RN-134 specifications, Available at: http://www.rovingnetworks.com/documents/rn-134-ds.pdf [Last Access 17th June 2010]

[27] Matchport b/g, Web site. Available at: http://www.lantronix.com/device-networking/embeddeddevice-servers/matchport.html [Last Access 17th June 2010]

[28] WiPort Web site. Available at: http://www.lantronix.com/device-networking/embedded-deviceservers/wiport.html

[29] Matchport b/g Pro, Web site. Available at: http://www.lantronix.com/device-networking/embeddeddevice-servers/matchport-bg-pro.html [Last Access 17th June 2010]

[30] RCM4400W RabbitCore module specifications, Available at: http://www.rabbit.com/products/RCM4400W/#description [Last Access 17th June 2010]

[31] Connect Wi-ME specifications, Available at: http://ftp1.digi.com/support/documentation/90000897_G.pdf [Last Access 17th June 2010]

[32] ConnectCore Wi-9C specifications, Available at: http://ftp1.digi.com/support/documentation/connectcore_wi9c_eu_doc.pdf

[33] J. Lloret, M. Garcia, D. Bri and J. R. Diaz. A Cluster-Based Architecture to Structure the Topology of Parallel Wireless Sensor Networks. Sensors. Vol. 9 Issue: 12. Pp. 10513-10544. December 2009.

[34] J. Lloret, M. Garcia, D. Bri and S. Sendra. A Wireless Sensor Network Deployment for Rural and Forest Fire Detection and Verification. Sensors. Vol. 9 Issue: 11. Pp. 8722-8747 October 2009.

[35] M. A. Menezes. "BEAN: Uma Plataforma Computacional para Rede de Sensores Sem Fio". April 2004. Available at http://homepages.dcc.ufmg.br/~mmvieira/publications/bean.pdf [Last Access 7th June 2010]

[36] J. Beutel, O. Kasten and M. Ringwald, "BTnodes -- a distributed platform for sensor nodes", Proceedings of the 1st international conference on Embedded networked sensor systems, November 05-07, 2003, Los Angeles, California, USA

[37] S. Coleri, S. Y. Cheung, and P. Varaiya, "Sensor networks for monitoring traffic". In 42nd Annual Allerton Conference on Communication, Control, and Computing, University of Illinois, September 2004.

[38] P. Dutta and D. Culler. "Epic: An Open Mote Platform for Application-Driven Design". In Proceedings of the 7th international Conference on information processing in Sensor Networks. pp. 547-548. Washington, DC, U.S.A. April 22 - 24, 2008.

[39] EYES Project. Available at http://www.eyes.eu.org/ [Last Access 7th June 2010]

2010, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

[40] FlatMesh FM2. Available at http://www.senceive.com/Products.aspx [Last Access 7th June 2010]

[41] G. Werner-Allen, G. Tewari, A. Patel, M. Welsh, and R. Nagpal. Firefly-inspired sensor network synchronicity with realistic radio effects. In Proceedings of the 3rd international Conference on Embedded Networked Sensor Systems (SenSys '05). Pp 142-153. San Diego, California, USA, November 2-4, 2005.

[42] Mikko Kohvakka. “Medium Access Control and Hardware Prototype Designs for Low-Energy Wireless Sensor Networks”. Thesis for the degree of Doctor of Technology. May, 2009. Available at http://www.tkt.cs.tut.fi/research/daci/pub_open/Kohvakka-Medium_Access_Control_and_Hardware_Prototype_designs_for_Low-Energy_Wireless_Sensor_Networks.pdf [Last Access 7th June 2010]

[43] R. Kling, R. Adler, J. Huang, V. Hummel and L. Nachman. “Intel mote: Sensor network technology for industrial applications”. The Fourth International Symposium on Information Processing in Sensor Networks (IPSN'05). April 25-27, 2005, UCLA, Los Angeles, California, USA.

[44] Iris datasheet. Available at http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/IRIS_Datasheet.pdf [Last Access 7th June 2010]

[45] N. Madabhushi. “KMote-Design and Implementation of a low cost, low power hardware platform for wireless sensor networks”. May 2007. Available at http://www.cse.iitk.ac.in/~moona/students/Y5111028.pdf [Last Access 7th June 2010]

[46] Mica2 datasheet. Available at http://www.xbow.com/products/Product_pdf_files/Wireless_pdf/MICA2_Datasheet.pdf [Last Access 7th June 2010]

[47] MicaZ datasheet. Available at http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MICAZ_Datasheet.pdf [Last Access 7th June 2010]

[48] J. Johansson, M. Völker, J. Eliasson, A. Östmark, P. Lindgren, and J. Delsing., “Mulle: A minimal sensor networking device - implementation and manufacturing challenges” in IMAPS Nordic 2004, pp. 265–271, Helsingør, Germany, September 26-28, 2004.

[49] Neo mote web. Available at http://www.xbow.jp/neokit-e.html [Last Access 7th June 2010]

[50] Redbee web. Available at http://redwirellc.com/store/index.php?route=product/product&product_id=53 [Last Access 7th June 2010]

[51] F. Zhao, L. Guibas, “Wireless Sensor Networks – An information processing approach”, Boston: Elsevier-Morgan Kaufmann publishers, 2004.

[52] Marcos Augusto M. Vieira, Claudionor N. Coelho. Jr, Diógenes Cecílio da Silva Junior, José M. da Mata1, “Survey on wireless sensor network devices”. Proceedings of IEEE Emerging Technologies and Factory Automation 2003, vol. 1, p. 537-544

[53] Sun spot website. Available at http://www.sunspotworld.com/docs/index.html [Last Access 7th June 2010]

[54] J. Polastre, R. Szewczyk, D. Culler. "Telos: Enabling Ultra-Low Power Wireless Research", The Fourth International Conference on Information Processing in Sensor Networks: Special track on Platform Tools and Design Methods for Network Embedded Sensors, 2005.

[55] TelosB datasheet. Available at http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/TelosB_Datasheet.pdf [Last Access 7th June 2010]

[56] TinyNode datasheet. Available at http://www.tinynode.com/uploads/media/SH-TN184-103_rev1.1.pdf [Last Access 7th June 2010]

[57] T-mote sky datasheet. Available at http://www.moteiv.com/products/docs/tmote-sky-datasheet.pdf [Last Access 7th June 2010]

[58] J. McLurkin, “Algorithms for distributed sensor networks,” Master’s thesis, University of California, Berkeley, 1999.

[59] D. Lymberopoulos and A. Savvides. “XYZ: a motion-enabled, power aware sensor node platform for distributed sensor network applications”. In Proceedings of the 4th international Symposium on information Processing in Sensor Networks. Los Angeles, California, April 24 - 27, 2005.

2010, © Copyright by authors, Published under agreement with IARIA - www.iaria.org