Hindawi Publishing Corporation
Journal of Sensors
Volume 2016, Article ID 2978073, 16 pages
http://dx.doi.org/10.1155/2016/2978073

Research Article

An Open and Modular Hardware Node for Wireless Sensor and Body Area Networks

L. Ciabattoni,[1] A. Freddi,[2] S. Longhi,[1] A. Monteriù,[1] L. Pepa,[1] and M. Prist[1]

[1] *Università Politecnica delle Marche, Via Brecce Bianche, 60131 Ancona, Italy*
[2] *Università degli Studi eCampus, Via Isimbardi 10, 22060 Novedrate, Italy*

Correspondence should be addressed to M. Prist; m.prist@univpm.it

Received 14 August 2015; Revised 24 November 2015; Accepted 1 December 2015

Academic Editor: Christos Tsamis

Health monitoring is nowadays one of the hottest markets due to the increasing interest in prevention and treatment of physical problems. In this context the development of wearable, wireless, open-source, and nonintrusive sensing solutions is still an open problem. Indeed, most of the existing commercial architectures are closed and provide little flexibility. In this paper, an open hardware architecture for designing a modular wireless sensor node for health monitoring is proposed. By separating the connection and sensing functions in two separate boards, compliant with the IEEE1451 standard, we add plug and play capabilities to analog transducers, while granting at the same time a high level of customization. As an additional contribution of the work, we developed a cosimulation tool which simplifies the physical connection with the hardware devices and provides support for complex systems. Finally, a wireless body area network for fall detection and health monitoring, based on wireless node prototypes realized according to the proposed architecture, is presented as an application scenario.

1. Introduction

The application of Wireless Sensor Network (WSN) technology in different scenarios rapidly increased in the past few years [1–3]. The recent interest in this topic can be attributed to several factors:

(i) The growing availability on the market of small and inexpensive sensors and devices easy to embed.

(ii) The worldwide diffusion of networking technologies, such as Wi-Fi, Ethernet, and Bluetooth which makes the communication between devices easier.

(iii) The presence of small computing devices (such as smartphones, tablets, and netbooks) in daily living.

As reported in [4, 5], one of its innovative deployments relates to biomedical sensor networks to monitor human vital signals. The most important change in demographic situation in the European Union is the transition towards a much older population. In this context a technology that promises to bring elderly people health care to a higher level of personalization is the Wireless Body Area Network (WBAN) [6, 7]. WBAN and Wireless Body Sensor Network (WBSN) are terms used to describe the application of wearable computing devices in Ambient Assisted Living (AAL). Patient monitoring systems can be used to collect patient physical status related data at home, and in some cases outdoors, to facilitate disease management, diagnosis, prediction, and follow-up [8, 9].

A Body Sensor Network (BSN) consists of a number of smart sensors with limited computing, storage, communication, and energy resources. These sensors are placed around the human body in order to collect various vital parameters related to the patient's health status. In this domain, due to the unacceptability of wired technologies over the human body, the wireless approach is the only solution; see, for example, [10–12]. However, WBSN technology still poses many challenges.

WSNs are composed of sensor nodes that autonomously operate by gathering sensors information and combining both communication and computational capabilities in a small form factor. These nodes, establishing a wireless link, collaborate with each other to execute application tasks. The main obstacles to the spread diffusion of this technology are mainly represented by communication issues (in terms of reliability and latency), power supply issues, and flexibility [13, 14]. Indeed, most of the existing commercial node architectures provide little flexibility, configurability, and the absence of interoperability among them. Daughter boards provide sensing capabilities, but the processing and communication modules are fixed and cannot be often extended. These limitations constrain the cross-usability of the same node in different applications and the use of different branded nodes in the same application.

In this paper we face the flexibility and customization problem in Wireless Sensor Networks, and in particular in Wireless Body Area Networks, presenting a novel architecture of an open hardware wireless modular sensor node. By separating the connection and sensing functions into two separate boards, the new architecture adds plug and play capabilities to analog transducers, while providing at the same time a higher level of customization for the whole network. The node hardware designer can thus exploit the modular architecture to implement different features, such as occupancy reduction, improved energy management, and increased power transmission, while always remaining compliant with the IEEE1451 standard. An additional contribution of the work regards the development of a cosimulation tool which simplifies the physical connection with the hardware devices and provides support for complex systems. We finally present prototypes of the wireless nodes and use them to build a wireless body area network for fall detection and health monitoring.

The paper is organized as follows. In Section 2 we review the state of the art in WSNs. The innovative design of the proposed node and a brief introduction to the IEEE standards are discussed in Section 3, while the hardware, chosen for the prototype, is reported in Section 4. In Section 5, a brief description of the developed cosimulation tool is reported. The proposed WBAN application is described in Section 6 where the modular node has been configured to monitor different vital parameters. Some remarks conclude the paper.

2. WSNs State of the Art

WSNs are generally composed of a large number of nodes which operate in a specific configuration. Typically, the sensor nodes are autonomous and spatially distributed and cooperate to monitor and to gather environmental conditions. Data processing can be done either in a centralized/decentralized mode or by sending data to a sink which sends them to other networks (e.g., through a gateway). Project, design, prototyping, and utilization of a WSN include a wide range of application-specific constraints. Even if WSNs are application dependent, it is possible to classify them in relation to common features:

(1) self-organization capabilities;

(2) short-range communication and/or star/multihop routing;

(3) centralized or decentralized cooperation of sensor nodes;

(4) capability to modify topology in runtime;

(5) constrains in energy consumption, transmission range, memory, computing power, and security.

The sensor nodes of a WSN, typically, are made up of three basic building blocks: sensing unit, computational unit, and communication unit. In order for a WSN to operate properly, the sensor nodes require an operating system, a routing protocol, and eventually a simulator. In the following, we provide an overview of the state of the art of operating systems, routing protocols, and simulators for WSNs, together with a short list of commercially available wireless sensor nodes.

2.1. Operating Systems. The most critical problems which can affect a WSN are the absence of hardware nodes standardization (the architectures provide little flexibility and configurability), energy consumption (typically provided by a small capacity battery), quality of service (communication delay, packets loss, and the out-of-order packets), scalability, distributed reconfiguration, programmability, and memory, which often allows only a few kilobytes of storage. In this context the Operating System (OS) becomes the manager for allocating the limited resources in a correctly and controlled manner. The main OSs for WSNs are presented in the following.

2.1.1. TinyOS. TinyOS is open source, flexible, based on component and module, and designed specifically for wireless sensor networks [15]. TinyOS supports concurrent programs with very low memory requirements and it includes many libraries to manage network protocols, distributed services, transducer drivers, and data gathering tools. The runtime of this OS is based on a monolithic architecture class and it uses the component model to compose a static image that runs on the node. From version 2.1, TinyOS provides support for multithreading, called TOS Threads, which uses a cooperative threading approach.

2.1.2. Contiki. This OS, developed by the Swedish Institute of Computer Science, is a lightweight, open-source OS written in C for WSNs [16]. It is a highly portable OS and it is based on an event-driven kernel. Contiki provides multitasking, so called Protothreads, that can be used at the individual process level. Contiki project includes many features to support an application-specific scenario, like multitasking kernel, preemptive multithreading, Protothreads, TCP/IP protocol, IPv6 protocol, a simple web server, a light telnet client, and so forth. Although Contiki supports dynamic memory management, it does not provide any support for real-time applications.

2.1.3. MANTIS. The MultimodAl system for NeTworks of In Situ Wireless Sensors (MANTIS) [17] is an operating system for WSNs based on multithreaded approach. MANTIS is a lightweight and energy efficient OS which includes kernel, scheduler, and network stack and, in addition, it is portable across multiple platforms, that is, PDA or a PC.

2.1.4. Nano-RK. Nano-RK is a real-time OS for WSNs based on multitasking [18]. The design goals for Nano-RK are the multihop networking, efficient power management to extend WSN lifetime, light applications with limited resources, and priority-based scheduling.

2.1.5. LiteOS. LiteOS is a Unix-like OS developed by the University of Illinois at Urbana-Champaign to support WSNs programming [19]. LiteOS provides a familiar programming environment based on Unix, threads, and C. It follows a hybrid programming model that allows both event-driven and thread-driven programming.

2.2. Routing Protocols. Routing is the key process for data transmission within a WSN. It consists in determining a path between the source node and the sink (destination) node. The routing protocols can be mainly classified into different ways as follows.

2.2.1. Path Establishment Based Routing Protocols. The routing paths can be calculated in three different ways: proactive, reactive, or hybrid. The proactive protocols develop all the possible routes before they are needed and then create a routing table in each node. Reactive protocols use a dynamic research techniques based on the request to send message. The hybrid routing strategies, instead, use clustering techniques to stabilize and scale the network and thus are generally applicable to networks of larger size and contain both strategies.

2.2.2. Network-Based Routing Protocols. Protocols included into this family are further classified into three classes in relation to their functionalities.

Flat-Based Routing. Flat-based routing is used when a WSN is composed of a large amount of sensor nodes and the gateway sends a request to a group of specific motes in a bounded area and waits for response.

Hierarchical-Based Routing. Hierarchical-based routing is used when within a WSN scalability and efficient communications are needed. This protocol is based on an energy efficient method in which nodes with high batteries are randomly selected for data analysis and sending data, while nodes with low battery are used for sensing and send data to the master. This increases the network scalability, lifetime, and energy usage.

Location-Based Routing. In this routing architecture, motes are distributed randomly, with a geographic position known in a specific region and the distance between nodes is calculated using the signal strength received from those nodes.

2.2.3. Operation-Based Routing Protocols. WSNs applications as well as routing protocols can be classified according to their operations.

Multipath Routing Protocols. These protocols provide multiple path selection in order to decrease delay and increase network performance, but they consume a great amount of energy.

Query-Based Routing Protocols. This type of protocols is based on distributed data queries using high level languages.

Negotiation-Based Routing Protocols. These protocols use intelligent algorithm for communication based on network available resources.

2.2.4. Next-Hop Selection-Based Routing Protocols. These protocols determine the next hop on the route. The decision can be based on

(i) the query content;

(ii) probabilistic approach and random selection of the next-hop neighbor;

(iii) the known position of the neighbors and the destination.

Alternatively, each mote in the network can decide individually whether to forward a message or not.

2.3. Commercial Solutions. Due to a wide range of application areas, a great number of solutions for wireless sensor nodes have been designed and commercialized. Nowadays there are mainly two kinds of nodes used in WSNs. The first one is the normal sensor node deployed to sense the phenomena and gather data, while the second is the gateway node that interfaces sensor networks to the external world. Commercial solutions, research prototypes of wireless sensor nodes, and their main features are illustrated in Table 1, and some samples are depicted in Figures 1 and 2. About sensor nodes for WBANs, different commercial solutions exist and some of them are shown in Figure 3.

2.4. WSN Simulators. In order to realize a real scenario or a test bench which provides realistic results, the physical architecture and the hardware development require a lot of resources, and the WSN programming and debug become extremely complex. In this context, wireless sensor network simulation becomes a very important and essential tool which provides good results in a cost effective way. The WSN simulators can be divided into different categories in relation to their features and applications:

(i) code level simulators;

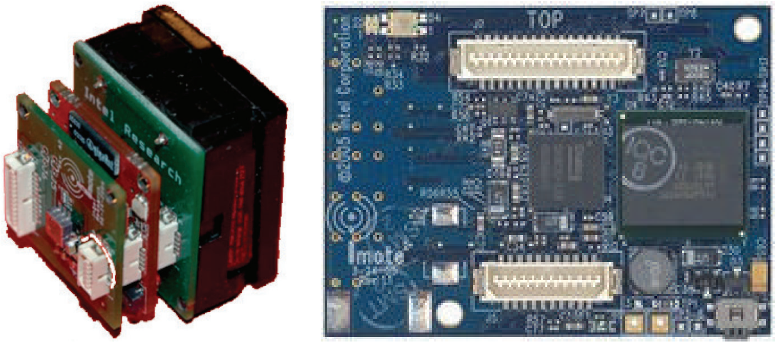
(ii) topology control simulators;

(iii) environment and wireless medium simulators.

Due to the ability to increase the real WSN prototyping, the Cross Levels Simulators, like COOJA, have become an important class of simulators. This kind of simulators operates at three abstraction levels: the network level, the operating system level, and the machine code instruction set level. Although they are open source, flexible, and extensible at all levels, the test interface, the external connection at a physical

Table 1: Commercial nodes.

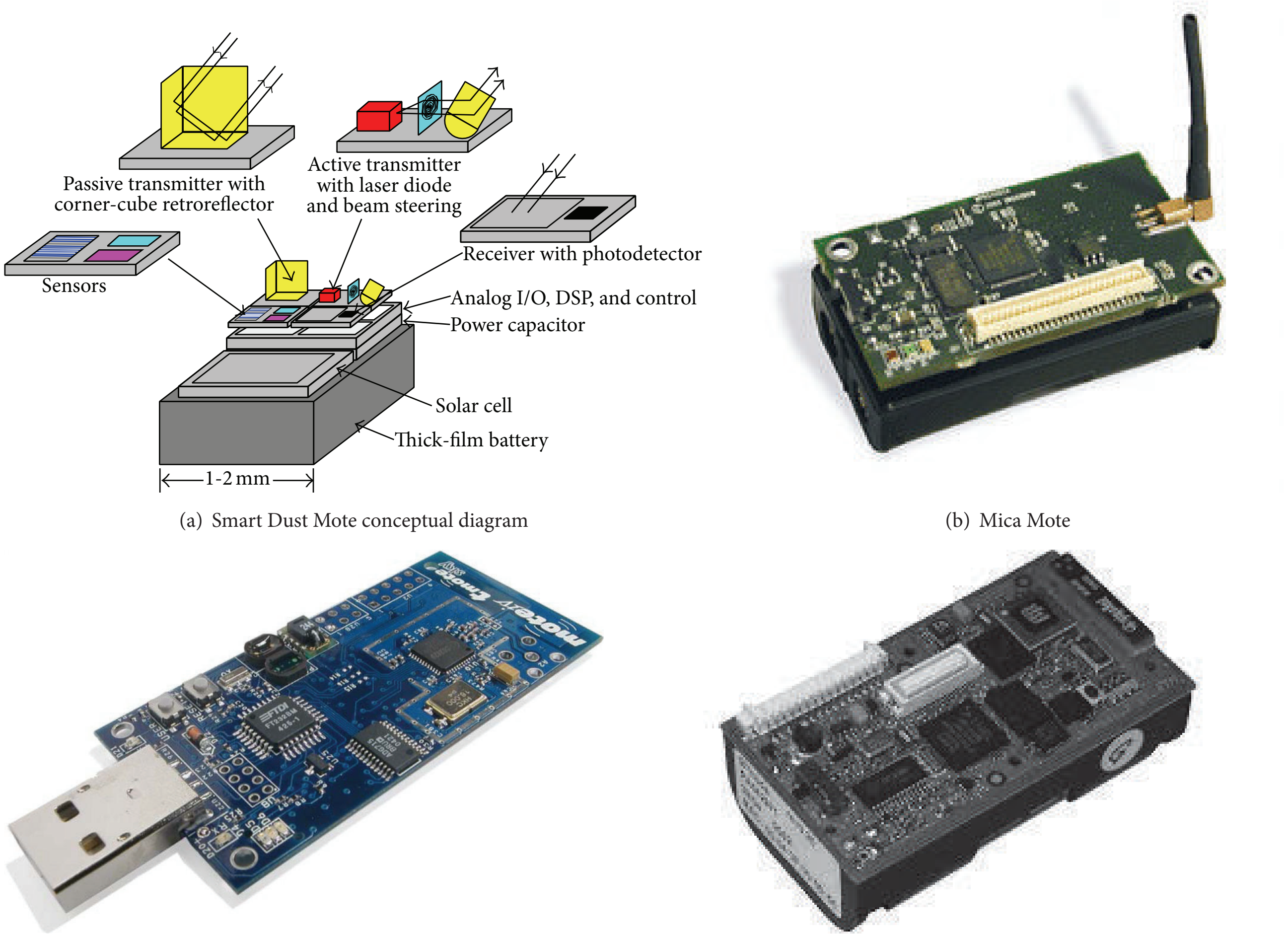
Node Name	Micro	Transceiver	Remarks	Maker
COOKIES [20]	ADUC841, MSP430	ETRX2 TELEGESIS, and ZigBit 868/915	Platform with hardware reconfigurability (Spartan 3FPGA based or Actel Igloo)	Centro de Electronica Industrial, Universidad Politécnica de Madrid
BTnode [21]	Atmel ATmega 128L (8 MHz @ 8 MIPS)	Chipcon CC1000 (433–915 MHz) and Bluetooth (2.4 GHz)	BTnut and TinyOS support	Computer Engineering and Networks Laboratory, ETH Zurich
EPIC mote [22]	TI MSP430	250 kbit/s 2.4 GHz IEEE802.15.4 Chipcon Wireless Transceiver	TinyOS	University of California at Berkeley
Eyes [23]	MSP430F149	TR1001	PeerOS Support	TU Berlin, Germany
FlatMesh FM1 [24]	16 MHz	802.15.4 compliant	Commercial system, for digital sensors. Second generation with built-in tilt sensor	Senceive Ltd., London, UK
IMote [25]	ARM core 12 MHz	Bluetooth with the range of 30 m	TinyOS Support	Intel Research, USA
IMote 2.0 [25]	Marvell PXA271 ARM 11–400 MHz	TI CC2420 802.15.4/ ZigBee compliant radio	Microsoft.NET Micro, Linux, and TinyOS Support	Intel Research, USA
T-Mote Sky [26]	TI MSP430	250 kbit/s 2.4 GHz IEEE802.15.4 Chipcon Wireless Transceiver	Contiki, TinyOS, SOS, and MantisOS Support	Advantic Sistemas y Servicios S.L., Madrid, Spain
Waspmote [27]	Atmel ATmega 1281	ZigBee/802.15.4/DigiMesh/RF, 2.4 GHz/868/900 MHz	GPRS, Bluetooth, and GPS modules, sensor boards	Libelium Comunicaciones Distribuidas S.L., Zaragoza, Spain
INDriya [28]	Atmel ATmega 128L	IEEE802.15.4 compliant XBee radios	Comprehensive sensor mote with ambient light, temperature, accelerometer, JPEG camera, PIR, sound sensor, TinyOS, TinyOS compliant, and IPv6 network supportive stacks for internetworking	Indrion Technologies Ltd., Bangalore, India
Iris Mote [29]	ATmega 1281	Atmel AT86RF230 802.15.4/ ZigBee compliant radio	Mote Runner, TinyOS, and MoteWorks Support	MEMSIC Inc., Andover, MA, USA
Mica2 [30]	ATMEGA 128L	Chipcon 868/916 MHz	TinyOS, SOS, and MantisOS Support	University of California at Berkeley, USA
MicaZ [30]	ATMEGA 128	TI CC2420 802.15.4/ ZigBee compliant radio	TinyOS, SOS, MantisOS, and Nano-RK Support	University of California at Berkeley, USA
Mulle [31]	Renesas M16C	Atmel AT86RF230 802.15.4/Bluetooth 2.0	Contiki, TinyOS, lwIP: TCP/IP and Bluetooth Profiles: LAP, DUN, PAN, and SPP Support	Eishtec AB, Sweden
PowWow [32]	MSP430F1612	TI CC2420 802.15.4/ ZigBee compliant radio	Open source; open hardware; and research platform	INRIA CAIRN, Université de Rennes
Preon32 [33]	ARM Cortex M3	Atmel AT86RF231 (2.4 GHz)	Virtual machine, Contiki and 6LoWPan supported	Virtenio GmbH, Berlin, Germany
Shimmer [34]	MSP430F1611	802.15.4 Shimmer SR7 (TI CC2420)	TinyOS Support. Built-In 3-Axis Accel., Tilt/Vib Sensor. Full range of expansion modules	Shimmer Sensing, Dublin, Ireland
TelosB [26]	TI MSP430	250 kbit/s 2.4 GHz IEEE802.15.4 Chipcon Wireless Transceiver	Contiki, TinyOS, SOS, and MantisOS Support	Advantic Sistemas y Servicios S.L., Madrid, Spain



(a) Intel Mote

(b) Intel Mote 2

Figure 1: Intel family motes.



(a) Smart Dust Mote conceptual diagram

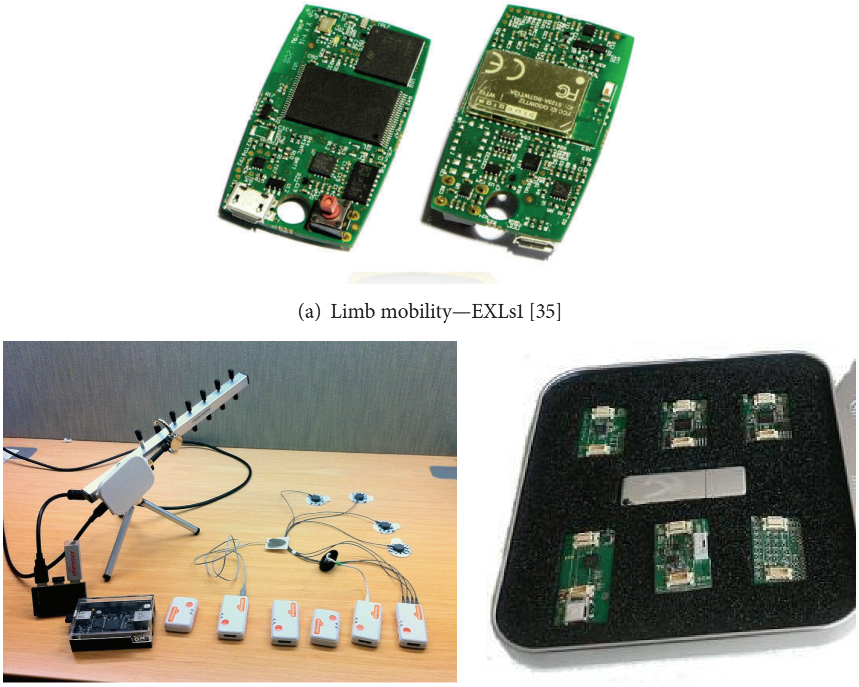
(b) Mica Mote

(c) Tmote Sky Mote

(d) BTnode Mote

Figure 2: Samples of commercial sensor nodes.

level and the direct interaction with the process control via the WSN are very poor [38]. In recent years, to solve these problems, a few numbers of cosimulators have been developed which integrate WSN simulators and MATLAB/Simulink tools. The Simulink tool provides a wide range of library and simulation model blocks but does not provide an adequate physical connection with the hardware devices used in a Cyber-Physical System (CPS), and it is not possible to simulate complex systems like WBAN or IEEE1451 standard architecture.



(a) Limb mobility—EXLs1 [35]

(b) SuWBAN Platform [36]

(c) BSN Development Kit [37]

Figure 3: Samples of WBAN sensor nodes.

The main *simulators* for WSNs are the following.

(1) Avrora. Avrora is an emulator and a code level simulator [39]. It is used to emulate the sensor hardware or to process the program code as it would be on a real hardware device. Avrora is a command-line framework compatible with MEMSIC Mica2 and MicaZ sensor platforms.

(2) TOSSIM. It is an emulator for WSNs running TinyOS [40]. The simulation environment permits creating a common topology which runs exactly the same TinyOS applications. Unfortunately, TOSSIM simulator supports only TinyOS Operating System and the MICAZ hardware platform [44].

(3) COOJA. COOJA Simulator, by Swedish Institute of Computer Science, is an open-source simulator for the Contiki sensor node operating system [41]. The simulator operates at three abstraction levels, the code level, the topology control level, and the environment and wireless medium level.

(4) Atarraya. Atarraya is a simulator for topology construction and topology maintenance in WSNs [42].

The main *cosimulators* for WSNs are the following.

(5) WCPS. In [43] an open-source simulation environment for wireless control systems, namely, Wireless Cyber-Physical Simulator (WCPS), has been proposed. This solution integrates Simulink and the TOSSIM wireless sensor simulator [40]. WCPS has been used to manage the physical systems and the wireless sensor-actuator networks used for control.

(6) NCSWT. The Networked Control Systems Wind Tunnel (NCSWT), a new integrated modeling and simulation tool for the evaluation of networked control systems, has been proposed in [45]. NCSWT integrates MATLAB/Simulink and ns-2 simulator using the High Level Architecture for an accurate time synchronization and data communication in heterogenous simulations [45].

(7) PiccSIM. In [46] the authors proposed a new Platform for Integrated Communications and Control design, Simulation, Implementation and Modeling (PiccSIM) composed of Simulink and ns-2. The communication between the two frameworks, for sending and receiving sensor data, time synchronization, and node position, is network-based with UDP packets.

(8) GISOO. A new virtual testbed for simulation of wireless cyberphysical systems that integrates two state-of-the art simulators, Simulink and COOJA, namely, GISOO, has been developed by the Swedish KTH Royal Institute of Technology [47]. The main base is the cross-level simulator COOJA, that permits to manage

(i) the code level simulators which emulate the sensor hardware, the process and, simultaneously, permit executing the program code directly on a real device;

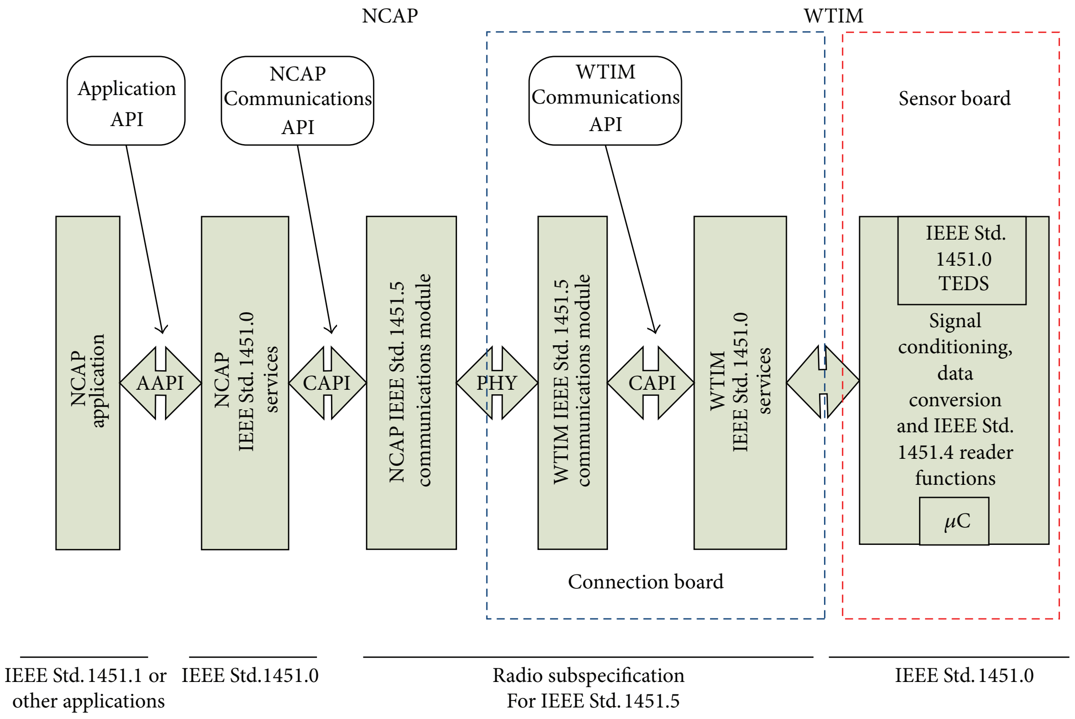

NCAP
WTIM
Application API
NCAP Communications API
WTIM Communications API
Sensor board
NCAP application
AAPI
NCAP IEEE Std. 1451.0 services
CAPI
NCAP IEEE Std. 1451.5 communications module
PHY
WTIM IEEE Std. 1451.5 communications module
CAPI
WTIM IEEE Std. 1451.0 services
IEEE Std. 1451.0 TEDS
Signal conditioning, data conversion and IEEE Std. 1451.4 reader functions
μC
Connection board
IEEE Std. 1451.1 or other applications
IEEE Std. 1451.0
Radio subspecification For IEEE Std. 1451.5
IEEE Std. 1451.0

Figure 4: New network architecture and IEEE1451 standard division.

(ii) the topology control level simulators which are used to study topology construction mechanisms;

(iii) the environment and wireless medium level simulators which offer the opportunity to simulate physical phenomena [48].

In addition to the COOJA features, GISOO integrates the Simulink advantages, thus allowing to extend the three levels with a new one for the cyberphysical modeling [49].

3. Design of the Proposed Wireless Sensor Node

It is recognized that standardization of wireless nodes will have a great impact on WSNs market expansion. In effect, it will help to decrease the cost of the system industrialization, reducing at the same time the development cycle. Among the existing and emerging standards for WSN, IEEE1451 has been used for the design of the proposed node. IEEE1451 is a family of standards introduced to add plug and play capabilities to smart transducers.

As defined in IEEE standard 1451.5 [50], a Wireless Transducer Interface Module (WTIM) is a device connected to transducers and, via Dot5AR protocol, to the Network Capable Application Processors (NCAPs). A WTIM differs from the standard TIM (Transducer Interface Module), as defined in IEEE Std 1451.0-2007, only for the wireless communication to the NCAP and provides two different functions. On one side, it allows the connection with the NCAP node while, on the other, makes possible the sensors interfacing.

The main design novelty presented in this paper is the separation of the WTIM into two independent boards to perform in a separate way the connection with the NCAP and the sensor interfacing functions, while always respecting IEEE1451 standards, as shown in Figure 4.

The connection board performs only actions involved in the wireless connection process with the NCAP node, maintaining in memory only the wireless related PHY TEDS (Transducer Electronic Data Sheets) and communication module commands. About the aspect of the transducers related commands, it acts as a gateway for the sensor board.

The sensor board has another microcontroller to perform the remaining functions: transducers interfacing, signal acquisition, and conditioning. In this board TEDS are stored, and all the information coming from the network, through the communication board, is processed. Since the IEEE1451 standards do not provide a specific hardware communication protocol between the two boards, the Universal Asynchronous Receiver-Transmitter (UART) protocol with a 3.3 V line has been adopted.

4. Hardware of the Proposed Wireless Sensor Node Prototype

The proposed hardware demo-board, called Argosd II, mainly developed for biometric data acquisition, is based on Argosd I, the previous version of the wireless sensor node which is briefly recalled in the following subsection.

4.1. Argosd I. Argos I is a Sky Mote based node, hardware designed by UC Berkeley [51] and produced by Crossbow [52]. Argosd I has the same Sky microcontroller and radio chip, but with a different architecture, and it was developed

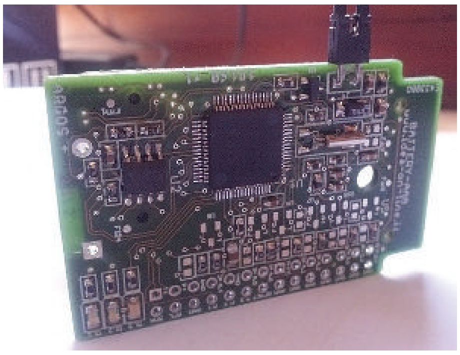


Figure 5: Argosd I Platfom.

for reducing the amount of components on board, the node size, and the power consumption, to increase the flexibility, and to reduce the current loss due to leakage that happens when the chip is pulling electrical current, even when powered down.

The core element of Argosd I is a Texas Instruments MSP430 MCU [53] which has been widely used in wireless sensors networks [54]. The main advantages of the MCU are the extremely low power during periods of Sleep Mode and the massive use in the WSN nodes. The MSP430 microcontroller has a 16-bit RISC CPU, connected via a data bus (MDB), and a 16-bit address bus (MAB) with memories (RAM and Flash/ROM) and peripheral I/O. In addition, it has a 8-MBit Serial Flash Memory Chip, the M25P80 by STMicroelectronics, with advanced write protection mechanisms and access via a high speed SPI-compatible bus. Moreover, an electronic registration number with external power supply, the low-cost DS2411 silicon serial number by Maxim Integrated, has been used in order to provide an absolutely unique identity that can be determined with a minimal electronic interface and associated with the network address (see Figures 5 and 6).

The low power techniques used to design the Argosd I platform are detailed in the following.

(i) *Reduction of the components on board and the utilization of low power chips.*

(ii) *Switching off External Circuits Duty Cycle*. All the low power modes are ineffective in the reduction of power consumption if the system is unable to control the power used by external circuits to the microcontroller. It is therefore crucial to analyse what physical modes or states are required and to partition the electronics in order to shut down unneeded circuitry. In the standard WSN platforms the flash, the sensors (temperature, battery voltage, humidity, etc.), the unique ID, and its bias circuit are energized at all times. To get the minimum current draw, Argosd I shuts down these circuits when they are not required.

(iii) *Utilization of High-Value Pull-Up Resistors*. It is more power efficient to use larger pull-up resistors on I/O pins, such as MCLR, I^2C signals, and switches, and also resistor dividers.

(iv) *Reduction of Operating Voltage*. Reducing the operating voltage of the device (Vdd) is a useful step in the reduction of the overall power consumption. When running, power consumption is mainly influenced by the clock speed. When sleeping, the most significant factor is leakage in the transistors. At lower voltages, less charge is required to switch the system clocks and transistors leak less current.

4.2. Argosd II. Starting from these considerations and following the IEEE1451 standard guidelines, the Argosd I has been divided into two parts.

(i) *The Wireless Transducer Interface Module (WTIM)*. The WTIM, namely Argosd II, which is used to manage the network policy, IEEE1451 commands, and the radio chip.

(ii) *The Transducer Electronic Data Sheets (TEDS)*. This module is used to manage the intelligent transducers. It contains the critical information needed by an instrument or measurement system to identify, to characterize, to interface, and to properly use the signal from analog sensors. It provides physical units to use (e.g., pressure in Pascal, temperature in K), sensors accuracy and their resolution, calibration information, and so forth.

The first prototype has been designed with the aim of testing the new architecture and the internal bus communication, and the developed 4-layer Printed Circuit Board (PCB) has a size of 35 mm × 60 mm. However a reduction in the size of the board to 20 mm × 20 mm is possible in order to optimize the usability and the free space of the PCB. The wireless module interface is a modular board with a single serial bus used to communicate and to exchange data with the TEDS. Its modular nature lends itself to the development of numerous TEDS for use in different application scenarios. The TEDS can be attached in an innovative plug and play way and includes communication, processing, and sensing. The differences in the radio chip and the sensors within the new developed architecture are highlighted in Figure 6. The hardware design principles are

(i) separation between intelligence transmission and intelligence transduction, in compliance with the IEEE1451 standard;

(ii) optimal energy management in both of the connected devices, TEDS and WTIM, as a result of the use of the MOSFET power switches;

(iii) increasing power transmission to reduce the influence of unintentional disturbance sources caused by coexistent radio systems, nonideal operation of communication devices and, in industrial environments, electric machines, welders, and so forth;

(iv) space reduction.

According to the standard, the first developed module must only execute the actions to process the wireless communication with the NCAP devices and, therefore, it is composed using a limited number of components. In particular,

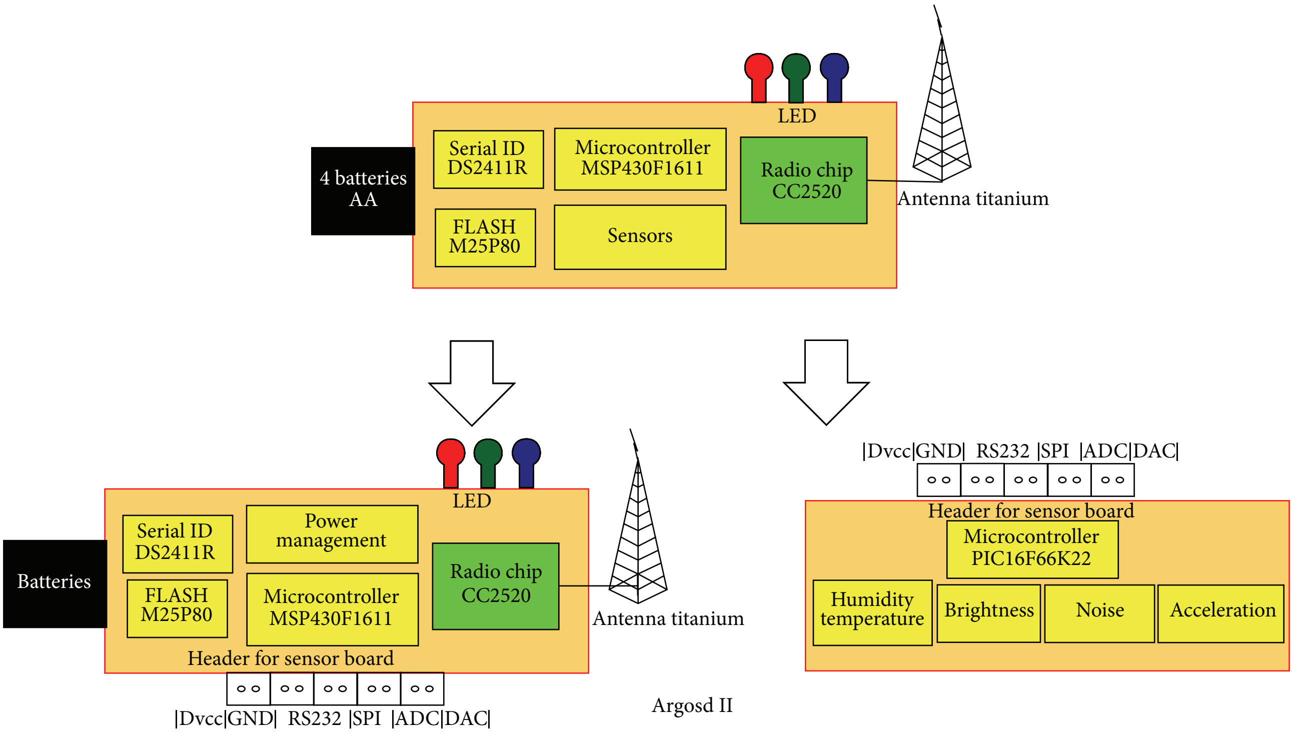


Figure 6: Connection board architecture.

as highlighted in Figure 7, the architecture comprises a radio chip, a processing unit, and other components like the unique serial number chip, the external flash memory, the three-state LEDs, the temperature sensor, and the battery charge sensor.

The Argosd II printed circuit board, shown in Figure 8, contains

(i) an ultralow power MSP430 16-bit RISC microcontroller by Texas Instruments, with five low power modes to achieve extended battery life in portable measurement applications;

(ii) a ChipCon CC2520 IEEE802.15.4 compliant radio transceiver with a programmable Output Power ranging from −20 to 5 dB and two power mode states;

(iii) an external flash memory due to the limited Random Access Memory (RAM) by MCUs;

(iv) simple wire antenna.

The Argosd II board does not include sensors but has only a serial bus interface and the power supply port to connect the intelligent transducers. A MOSFET is used to manage the power supply port in order to turn off/on the TEDS, controlling its power consumption, as shown in Figure 9. The batteries charge state is monitored by the internal ADC port with a specific AVcc reference.

5. Cosimulator for CPSs and WBAN

As already described in Section 2, in recent years a few numbers of cosimulators have been developed, which integrate WSN simulators and MATLAB/Simulink tools. However, they do not typically provide an adequate support to physical hardware connections and complex WSN architectures simulation. For this reason we decided to develop a custom cosimulator, which integrates the LabVIEW, a system-design platform and development environment for a visual programming language from National Instruments, and COOJA, a cross-level wireless sensor network simulator. The developed software module, called "GILOO," a Graphical Integration of LabVIEW and COOJA, enables to simultaneously develop and debug the control policy in a simulated or realistic scenario using or the virtual environment or the hardware modules, such as the National Instruments Data Acquisition, the FPGA platform for biometric data, and the CompactRio. Therefore, GILOO can be defined as an extension of COOJA, by which it becomes a four-level simulator, where the last level is the application connected directly to the physical reality [55].

The proposed architecture, depicted in Figure 10, integrates from one side the GISOO plugin implemented in COOJA (which monitors any call made by the native Analog-to-Digital Converter (ADC), Digital-to-Analog Converter (DAC), and serial port functions in the real wireless nodes), and from the other side it integrates the virtual instruments (VIs) (the building blocks of programs written in LabVIEW). Within a practical WBAN application, using Argosd II and two TEDs, the system can be decomposed into three subparts (see Figure 11):

(i) a physical process with sensors (e.g., a fall detection board and a blood pressure board, as in the example detailed in Section 6);

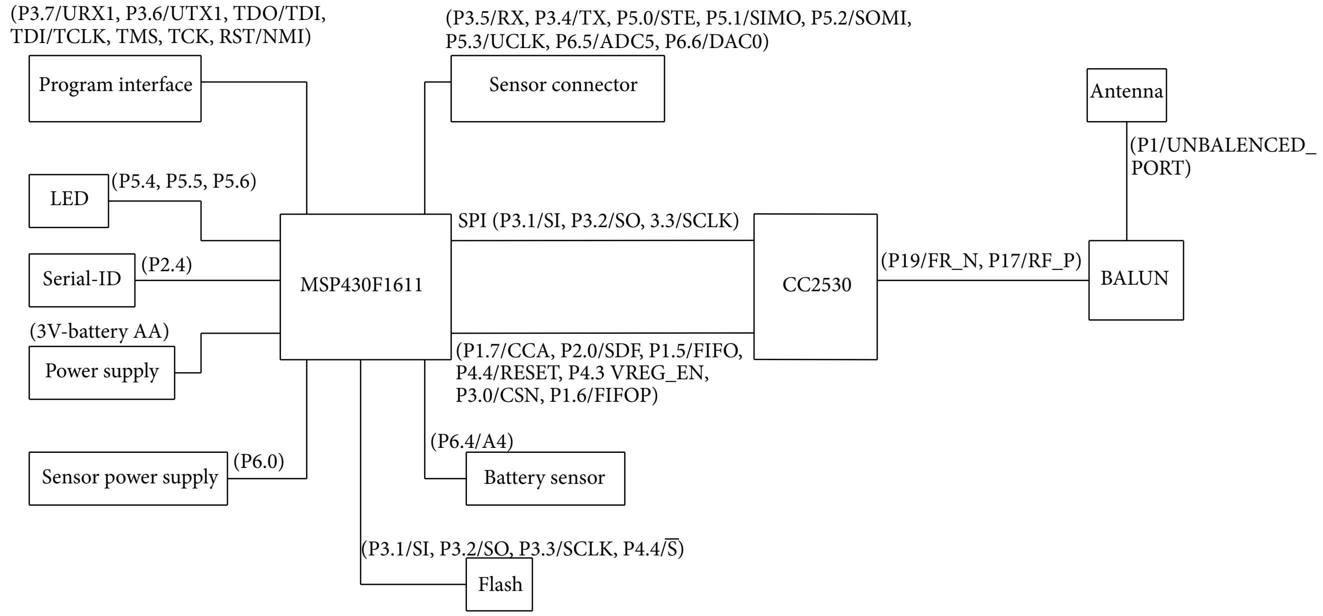

(P3.7/URX1, P3.6/UTX1, TDO/TDI,
TDI/TCLK, TMS, TCK, RST/NMI)
Program interface
(P3.5/RX, P3.4/TX, P5.0/STE, P5.1/SIMO, P5.2/SOMI,
P5.3/UCLK, P6.5/ADC5, P6.6/DAC0)
Sensor connector
Antenna
(P1/UNBALENCED_
PORT)
LED
(P5.4, P5.5, P5.6)
SPI (P3.1/SI, P3.2/SO, 3.3/SCLK)
Serial-ID
(P2.4)
MSP430F1611
CC2530
(P19/FR_N, P17/RF_P)
BALUN
(3V-battery AA)
Power supply
(P1.7/CCA, P2.0/SDF, P1.5/FIFO,
P4.4/RESET, P4.3 VREG_EN,
P3.0/CSN, P1.6/FIFOP)
(P6.4/A4)
Sensor power supply
(P6.0)
Battery sensor
(P3.1/SI, P3.2/SO, P3.3/SCLK, P4.4/$\overline{S}$)
Flash

Figure 7: Argosd II blocks schema.

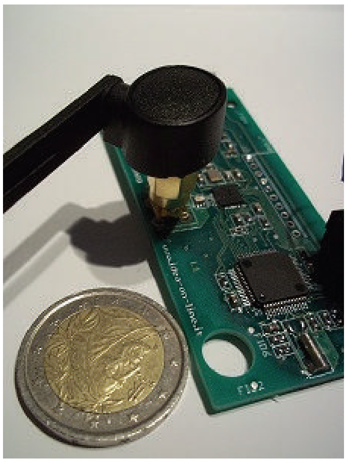


Figure 8: Argosd II printed circuit board.

(ii) a LabVIEW program that on the one hand interacts with the physical system through the communication modules of the National Instrument and that on the other communicates with COOJA through the LabVIEW block GILOO;

(iii) a wireless sensor network simulated in COOJA with an ad hoc communication protocol based on the IEEE802.15.4 standard and formed by a set of actors-sensors which acquire data from the physical world and a set of actors-actuators which interact with the control devices in LabVIEW.

COOJA simulates a wireless sensor network and, through the GILOO module, it interacts with the variables and the control unit in the LabVIEW program. The nodes used for sensory acquisition will read the field data and the nodes, dedicated to the control implementation, and act directly on the physical devices. The GILOO-LabVIEW library contains blocks to handle both the data communication and the time synchronization while the correct format of the serial messages and the relative bytes conversion have been implemented with a subVi routine.

6. Application Scenario: WBAN for Fall Detection and Health Monitoring

The proposed open hardware wireless modular node has been adopted to develop a body area network for fall detection and health monitoring. In the proposed scenario, modular nodes and a custom programmer are used to create a Low Power Area Networks (LowPans), as described in [50], adopting a Contiki operating system. Two main hardware categories are present:

(i) *wireless sensor nodes*, which acquire analog and digital inputs (depending on the sensors) and send measured values to the edge router every second using the 6LowPan standard;

(ii) *wireless edge router*, which opens the virtual channel to send data from WSN to the server and provides the communication between LowPans and Internet, implementing all the required features.

We will focus our attention on the sensor boards, designed starting from our modular node Argosd II, and realized to be integrated in a wearable system.

6.1. Fall Detection Board. Falling is one of the leading causes of serious health problem or injury-related deaths in the elderly so it is extremely important to detect or estimate when a potential fall can happen; see, for example, [56, 57]. A fall

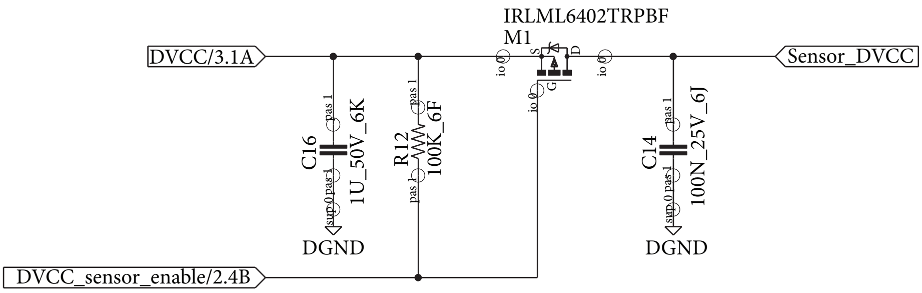


Figure 9: TEDS power management.

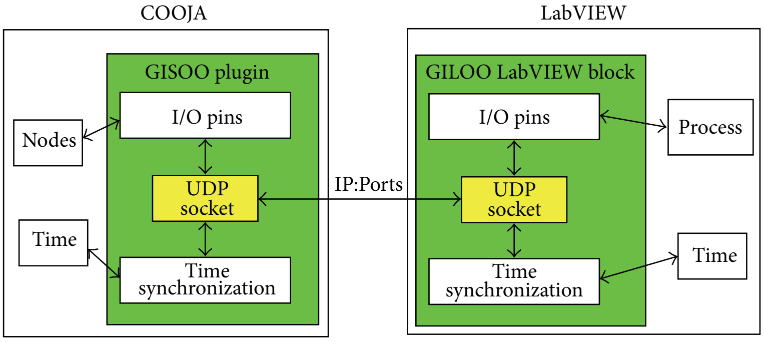


Figure 10: GILOO architecture.

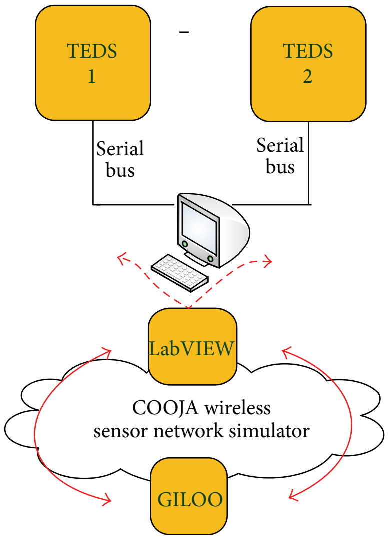


Figure 11: GILOO—WBAN architecture.

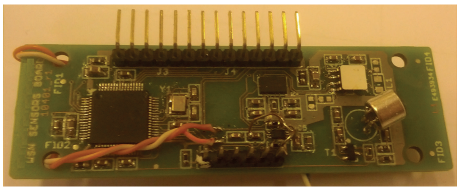


Figure 12: Fall detection board.

can occur not only when a person is standing, but also while sitting on a chair or lying on a bed during sleep.

In the design of our Fall Detection Board, as shown in Figure 12, we included 3D accelerometers and environmental noise detectors.

(i) Motion Detector: Accelerometer. The key feature of the fall detection is the ability to detect a change in the patient position and the high accelerations. In order to acquire this information, the board is equipped with an acceleration sensor model MMA7455L which is a digital output (I^2C and SPI) capacitive accelerometer (shown in Figure 13). The main features are built-in signal conditioning with a low pass filter, temperature compensation, self-test, and capability to detect 0 g. The power consumption, one of the most important features of the sensor, is 400 μA during the operation mode and 10 μA in standby mode.

(ii) Acoustic Fall Detection: Noise Sensor. Most of the wearable devices used for this purpose are versatile and effective in indoor environments, but they often have maintenance problems like power management, high dimensions, and potential inconvenience for carrying them all the time during daily living activities. Detecting a fall with acoustic sensors is practical, reliable, and inexpensive and does not cause privacy issues [58]. This sensor is used in a more complex fall detection system where a motion detector sensor is integrated with an acoustic sensor for learning new sounds and, thus, studying the correlation between the fall and noise. In order to acquire this information, the board is also equipped with

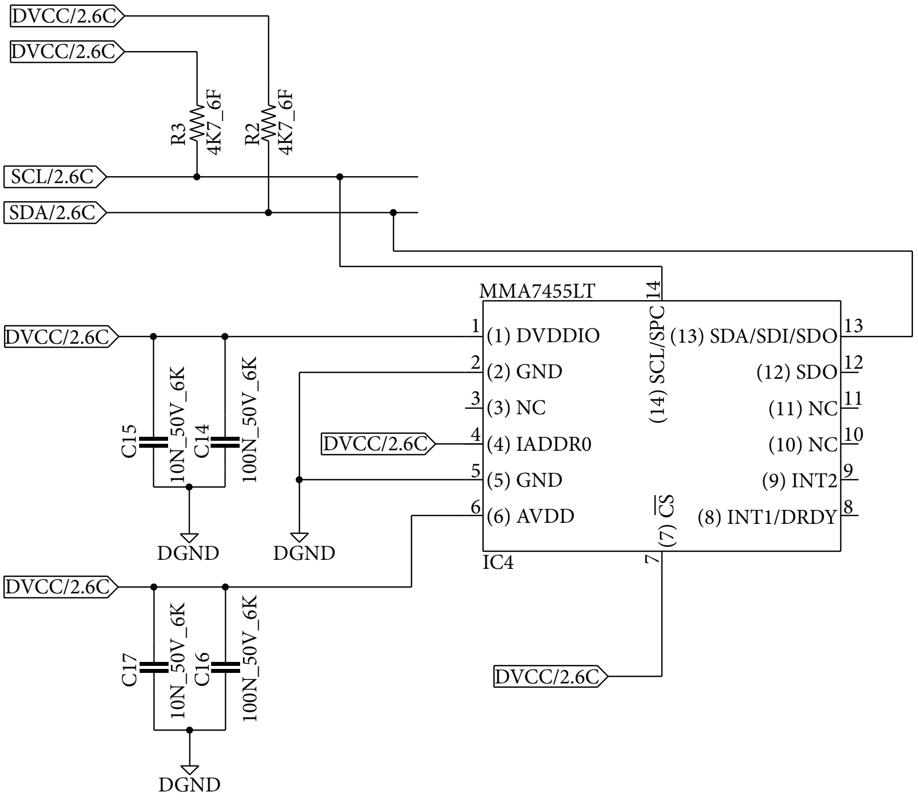


Figure 13: Accelerometer schema.

a noise sensor model CMC-5042PF-AC which is an omnidirectional noise sensor with a sensitivity of −42 dB (shown in Figure 14).

6.2. Fall Detection Algorithm. The board presented in the previous section was used to detect fall. We select a sample frequency of 200 Hz for acceleration data and 2000 Hz for acoustic data. A sliding window of 1 s is applied to both signals. Window step is 0.4 s. Before starting feature extraction, acceleration is high pass filtered through a second-order, zero-lag Butterworth filter with 0.5 cutoff frequency, in order to remove gravitational component. On each window we compute four features, two from the acceleration signal and two from the acoustic signal. In detail, acceleration is used to compute the following features.

(i) Mean Acceleration Magnitude (MAM). Consider

$$\text{MAM} = \frac{\sum_{i=1}^{N} \|a(i)\|}{N}$$
$$\|a(i)\| = \sqrt{a_x^2 + a_y^2 + a_z^2}, \tag{1}$$

where N is the number of acceleration samples in the window and $\|a(i)\|$ is the magnitude of the ith acceleration vector [59].

(ii) Reference Velocity (RV) [59]. Consider

$$\text{RV} = \int_{t\in W} \|a(t)\| \, \mathrm{d}t. \tag{2}$$

Acceleration features are defined similarly to the ones proposed by Huang and Chan [59], but our approach does not require any particular sensor placement or alignment, since it takes into account the magnitude of acceleration vector rather than its components.

From the acoustic signal we extract the following:

(i) the energy in the 0–200 Hz frequency band [60] is

$$E_{0-200} = \int_0^{200} A(f) \, \mathrm{d}f, \tag{3}$$

where $A(f)$ is the power spectrum of the acceleration;

(ii) the ratio between E_{0-200} and the energy in the 200–500 Hz frequency band is

$$ER = \frac{\int_0^{200} A(f) \, \mathrm{d}f}{\int_{200}^{1000} A(f) \, \mathrm{d}f}. \tag{4}$$

Features are sent as input of a fuzzy inference system which computes the warning level of the fall event. The goal of the integration of acoustic and motion features is the drastic reduction of false positives.

6.3. Fuzzy Logic Approach. Fuzzy rule-based systems (FRBS) have been successfully employed for system identification, control, and modeling in many areas [61, 62]. The approach considered in this work is the linguistic fuzzy modeling

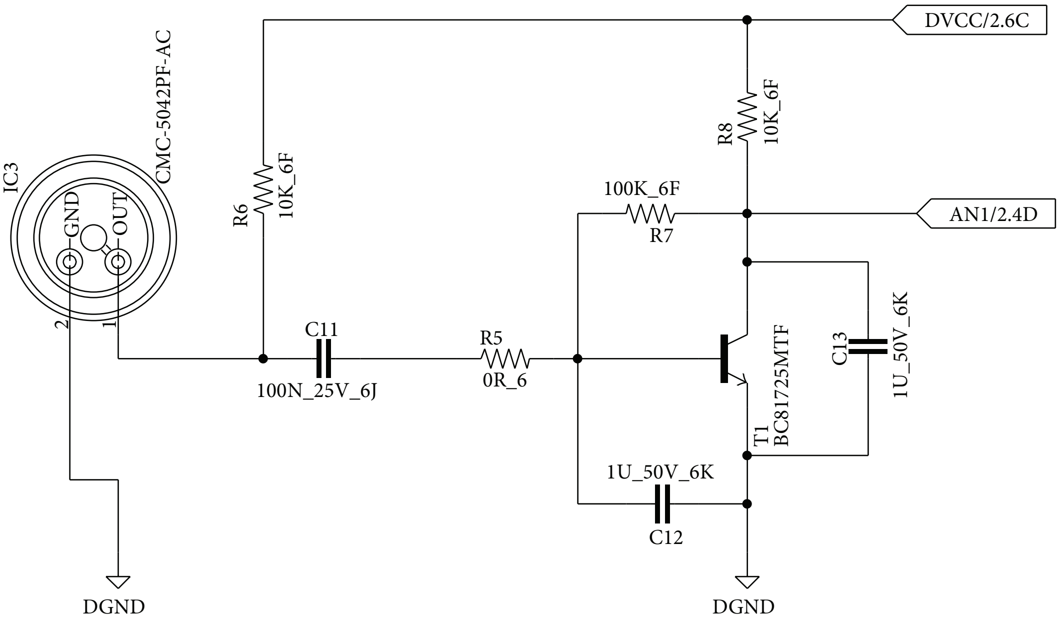


Figure 14: Noise sensor schema.

(LFM) with Mamdani rule structure due to its capability to model human knowledge in an explicit way. The membership functions of the variables involved in both of the fuzzy systems presented consist of triangular asymmetric and trapezoidal functions. The trapezoidal fuzzy set A in the universe of discourse $U \in \mathbb{R}$ with the membership function μ_A is parameterized by four real scalar parameters: (a, b, c, d) with $a < b \leq c < d$. This representation can be interpreted as a mathematical membership function as described in [63]

$$\mu_A(x) = \begin{cases} 0, & x < a \\ \dfrac{x-a}{b-a}, & a < x < b \\ 1, & b < x < c \\ \dfrac{d-x}{d-c}, & c < x < d \\ 0, & x > d. \end{cases} \quad (5)$$

When $b = c$, the triangular function can be considered as a particular case of the trapezoidal one (a sample of the fuzzification of the variables is shown in Figures 15 and 16).

Input values of acceleration features (MAM and RV) and noise ones are normalized over their thresholds, which are computed as mean plus one standard deviation from 10 s of normal activities. The fuzzy system is composed of four inputs and one output (the warning level) and the values of the fuzzy sets are reported in Table 2.

In particular when the output is more than 0.75, we classify the event as a "falling event." The fuzzy inference engine is composed of 36 rules chosen by examining the signals involved in the falling events.

6.4. Experimental Evaluation. The proposed fall detection board was tested on 8 healthy subjects emulating falls. Each subject emulated 30 falls in his home while performing daily living activities (ADL). Subjects were asked to annotate the time of their falls in order to compare detected falls with real ones. On the same time they performed 30 daily living activities which can be similar to a falling event (e.g., sitting

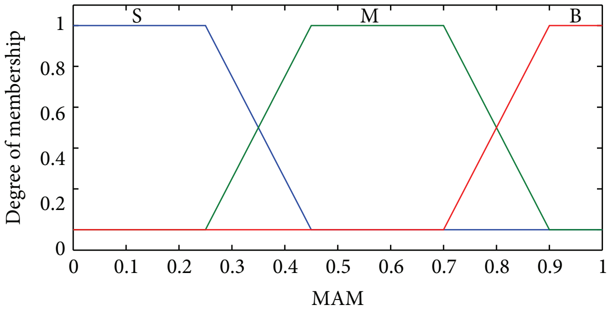


Figure 15: Fuzzy sets for the input variable mean acceleration magnitude.

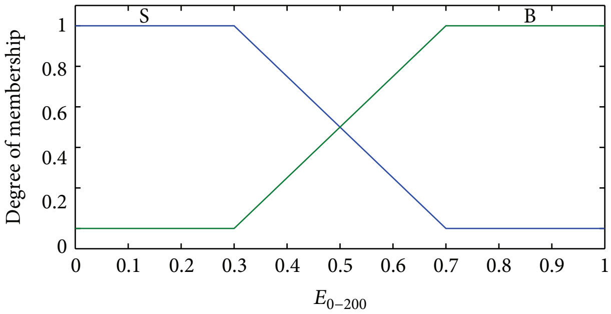


Figure 16: Fuzzy sets for the input variable energy 0–200 Hz.

Table 2: Fuzzy sets of the input and output variables: linguistic terms and their corresponding trapezoidal fuzzy sets parameters.

Input variables	Linguistic terms	Fuzzy sets (a, b, c, d)
E_{0-200}	Small	0, 0, 0.3, 0.7
	Big	0.3, 0.7, $+\infty$, $+\infty$
MAM	Small	0, 0, 0.25, 0.45
	Medium	0.25, 0.45, 0.7, 0.9
	Big	0.7, 0.9, $+\infty$, $+\infty$
RV	Small	0, 0, 0.3, 0.7
	Big	0.3, 0.7, $+\infty$, $+\infty$
ER	Small	0, 0, 0.25, 0.45
	Medium	0.25, 0.45, 0.7, 0.9
	Big	0.7, 0.9, $+\infty$, $+\infty$
Output variable	Linguistic terms	Fuzzy sets (a, b, c, d)
Warning level	Zero	0, 0, 0.1, 0.2
	Low	0.1, 0.2, 0.3, 0.4
	Medium	0.3, 0.4, 0.6, 0.7
	High	0.6, 0.7, 0.8, 0.9
	Very high	0.8, 0.9, 1, 1

rapidly). A number of 240 falling events and 240 ADL were recorded.

The system was able to detect as “true positive” 221 falling events with a success percentage of 92%. On the same time 6% of ADL were wrongly recognized as “falls.”

7. Conclusion

This paper presents an open hardware modular design of a wireless sensor node, which can be used for a wide range of applications and in particular for wireless body area networks. Following the IEEE1451 standard, the node has been designed by developing two main boards, related to the connection and the sensor interfaces, respectively. The main purpose of this design is to standardize the communication for the entire sensor network, thus giving the chance to use a wide range of sensors as plug and play devices, while granting at the same time a high level of customization. An additional contribution of the work is the development of a cosimulation tool which simplifies both hardware connection and software simulation. As an application scenario of the proposed modular node, we present a wireless body area network for fall detection and health monitoring.

In order to compare the proposed node to other similar commercial solutions, a performance analysis is currently under investigation. In the future, the proposed prototypes will be improved and integrated in a more complex wireless body area network, in order to provide a continuous health monitoring for ambient assisted living applications in smart homes.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] M. Grisostomi, L. Ciabattoni, M. Prist, G. Ippoliti, and S. Longhi, “Application of a wireless sensor networks and Web2Py architecture for factory line production monitoring,” in *Proceedings of the IEEE 11th International Multi-Conference on Systems, Signals and Devices (SSD ’14)*, pp. 1–6, Barcelona, Spain, February 2014.

[2] P. Wang, Y. Yan, G. Y. Tian, O. Bouzid, and Z. Ding, “Investigation of wireless sensor networks for structural health monitoring,” *Journal of Sensors*, vol. 2012, Article ID 156329, 7 pages, 2012.

[3] M. Kuorilehto, M. Hännikäinen, and T. D. Hämäläinen, “A survey of application distribution in wireless sensor networks,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2005, no. 5, pp. 774–788, 2005.

[4] V. Vaidehi, M. Vardhini, H. Yogeshwaran, G. Inbasagar, R. Bhargavi, and C. Hemalatha, “Agent based health monitoring of elderly people in indoor environments using wireless sensor networks,” *Procedia Computer Science*, vol. 19, pp. 64–71, 2013, The 4th International Conference on Ambient Systems, Networks and Technologies (ANT 2013), the 3rd International Conference on Sustainable Energy Information Technology (SEIT-2013).

[5] U. Anliker, J. A. Ward, P. Lukowicz et al., “AMON: a wearable multiparameter medical monitoring and alert system,” *IEEE Transactions on Information Technology in Biomedicine*, vol. 8, no. 4, pp. 415–427, 2004.

[6] E. Montón, J. F. Hernandez, J. M. Blasco et al., “Body area network for wireless patient monitoring,” *IET Communications*, vol. 2, no. 2, pp. 215–222, 2008.

[7] B. Alghamdi and H. Fouchal, “A mobile wireless body area network platform,” *Journal of Computational Science*, vol. 5, no. 4, pp. 664–674, 2014.

[8] A. C. W. Wong, D. McDonagh, O. Omeni, C. Nunn, M. Hernandez-Silveira, and A. Burdett, “Sensium: an ultra-low-power wireless body sensor network platform: design and application challenges,” in *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 6576–6579, Minneapolis, Minn, USA, September 2009.

[9] A.-W. Wong, D. McDonagh, G. Kathiresan et al., “A 1V, micropower system-on-chip for vital-sign monitoring in wireless body sensor networks,” in *IEEE International Solid-State Circuits Conference, Digest of Technical Papers (ISSCC ’08)*, pp. 138–602, San Francisco, Calif, USA, February 2008.

[10] S. J. Lee, C. (Andrew) Jung, K. Choi, and S. Kim, “Design of wireless nanosensor networks for intrabody application,” *International Journal of Distributed Sensor Networks*, vol. 2015, Article ID 176761, 12 pages, 2015.

[11] R. Katiyar, V. K. Pathak, and K. Arya, “A study on existing gait biometrics approaches and challenges,” *International Journal of Computer Science*, vol. 10, no. 1, pp. 135–144, 2013.

[12] D. Naranjo-Hern, L. M. Roa, J. Reina-Tosina, M. A. Estudillo-Valderrama, and G. Barbarov, “Low-power platform and communications for the development of wireless body sensor networks,” *International Journal of Distributed Sensor Networks*, vol. 2015, Article ID 431798, 13 pages, 2015.

[13] H. A. Nguyen, A. Förster, D. Puccinelli, and S. Giordano, “Sensor node lifetime: an experimental study,” in *Proceedings of the 9th IEEE International Conference on Pervasive Computing and Communications Workshops*, pp. 202–207, IEEE, Seattle, Wash, USA, March 2011.

[14] G. Anastasi, M. Conti, M. Di Francesco, and A. Passarella, “Energy conservation in wireless sensor networks: a survey,” *Ad Hoc Networks*, vol. 7, no. 3, pp. 537–568, 2009.

[15] P. Levis, S. Madden, J. Polastre et al., “Tinyos: an operating system for sensor networks,” in *Ambient Intelligence*, W. Weber, J. Rabaey, and E. Aarts, Eds., pp. 115–148, Springer, Berlin, Germany, 2005.

[16] Contiki, January 2015, http://www.contiki-os.org/.

[17] MANTIS, “Documentation Support for MANTIS,” 2015, http://mantisos.org/index/tiki-index.php%3Fpage=HomePage.html.

[18] NANO-RK, “NANO-RK operating system,” 2015, http://www.nano-rk.org.

[19] LiteOS, “LiteOS operating system,” 2015, http://www.liteos.net.

[20] Centro de Electronica Industrial and Universidad Politecnica de Madrid, “Cookies hardware platform,” 2014, https://cookieswsn.wordpress.com/.

[21] ETH Zurich, *Btnode Hardware Platform*, ETH Zurich, 2014, http://www.btnode.ethz.ch/.

[22] C. B. D. Kuncoro, “Miniature and low-power wireless sensor node platform: state of the art and current trends,” *IPTEK Journal of Proceedings Series*, vol. 1, no. 1, 2015.

[23] TU Berlin, *Energy Efficient Sensor Networks (EYES)*, 2014, http://www2.tkn.tu-berlin.de/research/eyes/.

[24] Senceive Ltd, “FlatMesh node,” 2014, http://www.senceive.com/index.php/flatmesh/.

[25] Intel Research Group, “Intel Mote generation 1 and 2 nodes,” 2014, http://tinyos.stanford.edu/tinyos-wiki/index.php/Imote2.

[26] Advantic Sistemas y Servicios S L, “TelosB mote module,” 2015, http://www.advanticsys.com/shop/mtmcm5000msp-p-14.html.

[27] Libelium Comunicaciones Distribuidas SL, “Waspmote overview,” 2015, http://www.libelium.com/products/waspmote.

[28] Indrion Technologies Ltd, *Indriya Development Platform (IDP)*, Indrion Technologies Ltd, 2014, http://indrion.co.in/developmentkit.php.

[29] MEMSIC inc, “IRIS sensor node family,” 2014, http://www.memsic.com/wireless-sensor-networks/.

[30] UC Berkeley, *Mica Family Nodes*, UC Berkeley, 2014, http://www.sensorsmag.com/networking-communications/mica-the-commercialization-microsensor-motes-1070.

[31] A. B. Eistec, “Mulle wireless sensor node,” 2014, http://www.eistec.se/mulle/wsn.

[32] INRIA CAIRN Project-Team, “PowWow: Power Optimized Hardware and Software FrameWork for Wireless Motes. A Hardware/Software Platform developed by INRIA CAIRN project-team,” 2015, http://powwow.gforge.inria.fr/.

[33] Virtenio GmbH, “PREON—Smart Wireless Devices,” 2015, http://www.virtenio.com/en/.

[34] Shimmer Innovation Campus, *Shimmer Sensor Nodes*, Shimmer Innovation Campus, 2015, http://www.shimmersensing.com/.

[35] Wireless Inertial Measurement Unit (IMU), http://www.exel-microel.com/eng_electronic_medical-wearable-technology-exl-s1_module.html.

[36] K. Heurtefeux, E. Ben Hamida, and H. Menouar, “Design and implementation of a sustainable wireless BAN platform for remote monitoring of workers health care in harsh environments,” in *Proceedings of the 6th International Conference on New Technologies, Mobility and Security (NTMS ’14)*, pp. 1–5, IEEE, Dubai, United Arab Emirates, April 2014.

[37] B. Lo, S. Thiemjarus, A. Panousopoulou, and G.-Z. Yang, “Bioinspired design for body sensor networks [life sciences],” *IEEE Signal Processing Magazine*, vol. 30, no. 1, pp. 165–170, 2013.

[38] M. Prist, S. Longhi, A. Monteriu, F. Giuggioloni, and A. Freddi, “An integrated simulation environment for wireless sensor networks,” in *Proceedings of the 16th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM ’15)*, pp. 1–3, IEEE, Boston, Mass, USA, June 2015.

[39] B. L. Titzer, D. K. Lee, and J. Palsberg, “Avrora: scalable sensor network simulation with precise timing,” in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN ’05)*, pp. 477–482, IEEE, Los Angeles, Calif, USA, April 2005.

[40] P. Levis, N. Lee, M. Welsh, and D. Culler, “TOSSIM: accurate and scalable simulation of entire tinyos applications,” in *Proceedings of the First International Conference on Embedded Networked Sensor Systems (SenSys ’03)*, pp. 126–137, ACM, New York, NY, USA, November 2003.

[41] F. Österlind, A. Dunkels, J. Eriksson, N. Finne, and T. Voigt, “Cross-level sensor network simulation with COOJA,” in *Proceedings of the 31st Annual IEEE Conference on Local Computer Networks (LCN ’06)*, pp. 641–648, Tampa, Fla, USA, November 2006.

[42] P. M. Wightman and M. A. Labrador, “Atarraya: a simulation tool to teach and research topology control algorithms for wireless sensor networks,” in *2nd International Conference on Simulation Tools and Techniques, Simutools ’09*, pp. 26:1–26:10, Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering (ICST), Brussels, Belgium, 2009.

[43] B. Li, Z. Sun, K. Mechitov et al., “Realistic case studies of wireless structural control,” in *Proceedings of the ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS ’13)*, pp. 179–188, Philadelphia, Pa, USA, April 2013.

[44] TOSSIM Simulator lessons, http://tinyos.stanford.edu/tinyos-wiki/index.php/TOSSIM.

[45] E. Eyisi, J. Bai, D. Riley et al., “NCSWT: an integrated modeling and simulation tool for networked control systems,” *Simulation Modelling Practice and Theory*, vol. 27, pp. 90–111, 2012.

[46] S. Nethi, M. Pohjola, L. Eriksson, and R. Jäntti, “Platform for emulating networked control systems in laboratory environments,” in *Proceedings of the IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WOWMOM ’07)*, Espoo, Finland, June 2007.

[47] W. Li, X. Zhang, and H. Li, “Co-simulation platforms for co-design of networked control systems: an overview,” *Control Engineering Practice*, vol. 23, pp. 44–56, 2014.

[48] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “Wireless sensor networks: a survey,” *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.

[49] KTH—Automatic Control Lab, “KTH—GISOO: a virtual testbed for wireless cyber-physical systems,” https://code.google.com/p/kth-gisoo/.

[50] I. Instrumentation and M. Society, “IEEE standard for a smart transducer interface for sensors and actuators wireless communication protocols and transducer electronic data sheet (TEDS) formats,” IEEE Std 1451.5-2007, 2007.

[51] Berkeley Wireless Embedded Systems Project, http://www.memsic.com/userfiles/files/Datasheets/WSN/telosb_datasheet.pdf.

[52] Crossbow Technology Inc, http://www.xbow.com/.

[53] Texas Instruments, “MSP430 16-Bit Ultra-Low Power MCUs,” http://www.ti.com/lit/ug/slau049f/slau049f.pdf.

[54] E. Lattanzi and A. Bogliolo, "VirtualSense: a java-based open platform for ultra-low-power wireless sensor nodes," *International Journal of Distributed Sensor Networks*, vol. 2012, Article ID 154737, 16 pages, 2012.

[55] M. Prist, A. Freddi, S. Longhi, and A. Monteriu, "An integrated simulation module for wireless cyber-physical system," in *Proceedings of the 15th International Conference on Environment and Electrical Engineering (EEEIC 15)*, pp. 1397–1402, IEEE, Rome, Italy, June 2015.

[56] F. Wu, H. Zhao, Y. Zhao, and H. Zhong, "Development of a wearable-sensor-based fall detection system," *International Journal of Telemedicine and Applications*, vol. 2015, Article ID 576364, 11 pages, 2015.

[57] J. M. Sim, Y. Lee, and O. Kwon, "Acoustic sensor based recognition of human activity in everyday life for smart home services," *International Journal of Distributed Sensor Networks*, vol. 2015, Article ID 679123, 11 pages, 2015.

[58] Y. Li, K. C. Ho, and M. Popescu, "A microphone array system for automatic fall detection," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 5, pp. 1291–1301, 2012.

[59] C. N. Huang and C. T. Chan, "A zigbee-based location-aware fall detection system for improving elderly telecare," *International Journal of Environmental Research and Public Health*, vol. 11, no. 4, pp. 4233–4248, 2014.

[60] M. Popescu and S. Coupland, "A fuzzy logic system for acoustic fall detection," in *Proceedings of the AAAI Symposium on AI in Eldercare*, pp. 78–83, Washington, DC, USA, November 2008.

[61] L. Pepa, L. Ciabattoni, F. Verdini, M. Capecci, and M. G. Ceravolo, "Smartphone based fuzzy Logic freezing of gait detection in Parkinson's disease," in *Proceedings of the 10th International Conference on Mechatronic and Embedded Systems and Applications (MESA '14)*, pp. 1–6, IEEE, Senigallia, Italy, September 2014.

[62] L. Ciabattoni, M. Grisostomi, G. Ippoliti, and S. Longhi, "Fuzzy logic home energy consumption modeling for residential photovoltaic plant sizing in the new Italian scenario," *Energy*, vol. 74, pp. 359–367, 2014.

[63] L. Ciabattoni, F. Ferracuti, M. Grisostomi, G. Ippoliti, and S. Longhi, "Fuzzy logic based economical analysis of photovoltaic energy management," *Neurocomputing*, vol. 170, pp. 296–305, 2015.

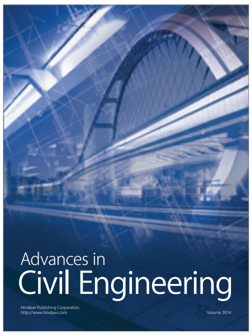
Journal of
Engineering

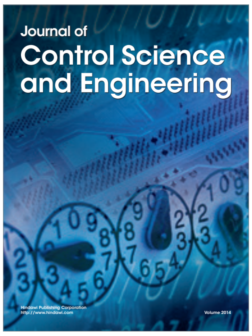
The Scientific
World Journal

International Journal of
Rotating
Machinery

Journal of
Sensors

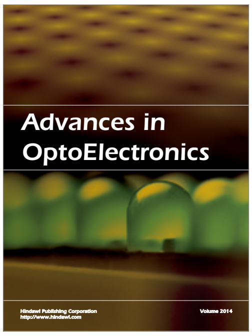
International Journal of
Distributed
Sensor Networks


Advances in
Civil Engineering


Journal of
Control Science
and Engineering

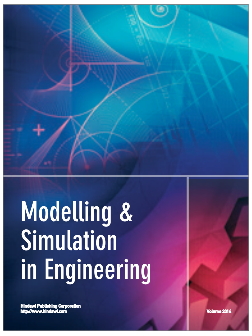
Journal of
Robotics

Journal of
Electrical and Computer
Engineering

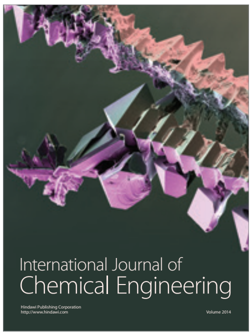

Advances in
OptoElectronics

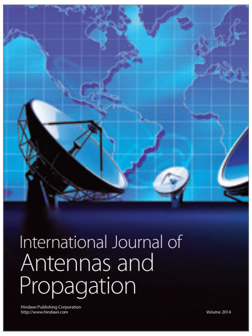
VLSI Design

International Journal of
Navigation and
Observation

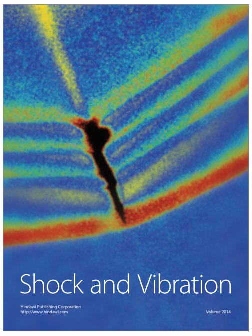

Modelling &
Simulation
in Engineering

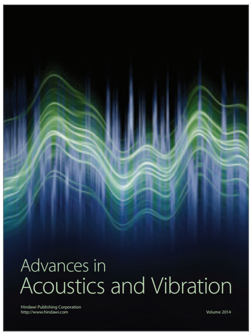
International Journal of
Aerospace
Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and
Propagation

Active and Passive
Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration