Application of Wireless Sensor Networks to the Automobile

Jorge Tavares, Fernando J. Velez, João M. Ferro

Instituto de Telecomunicações, DEM-UBI, Calçada Fonte do Lameiro,
6201-001 Covilhã, Portugal
Phone: +351-275329919, Fax: +351-275329972, e-mail: a12504@ubi.pt,
fjv@ubi.pt, joaomferro@gmail.com

Abstract. *Some applications of Wireless sensor networks (WSNs) to the automobile are identified, and the use of MICAz motes operating at 2.4GHz is considered together with TinyOS support. These WSNs are being conceived in order to measure, process and supply to the user diverse types of information during an automobile journey. Examples are acceleration and fuel consumption, identification of incorrect tire pressure, verification of illumination, and evaluation of the vital signals of the driver. A survey on WSNs concepts is presented, and the wireless sensor network itself (transmitter/receiver/control board) is developed. Aspects of the definition of the architecture and the choice/implementation of the protocols are identified. Security aspects are also addressed.*

Keywords: wireless sensor network, mica motes, automobile, architectures, protocols

1. Introduction

Nowadays, Wireless Sensor Networks (WSNs) have several applications in industry, atmosphere monitoring, and defence, among others. Besides instrumentation concepts, WSNs involve aspects of wireless communications, and of networks architectures and protocols. Due to technological innovations in the area of wireless communications, digital electronics, and personal micro-electromechanical systems a revolution is occurring in the area of measurement with remote wireless sensors [1]. In particular, WSNs are characterised by a high amount of sensor nodes with multi-hop communication capabilities. These tiny sensors can be spread inside the environment to be monitored or close to it, with positions that are not pre-determined. Indeed, they are set randomly as wireless sensors can be dropped onto places of difficult access from helicopters or airplanes [2].

The application of wireless sensors networks to the automobile constitutes a challenge and was faced as an endeavour; we conceived a wireless sensor system capable to collect, process and supply several types of technical information (to the user) during a car race. Examples are acceleration and fuel consumption, identification of wrong tire pressure value, acknowledgment of illumination failures (turn lights, brake lights, front lights, and register plate lights), and determination of the vital signals of the driver. We chose Crossbow MICAz sensors operating at 2.4GHz (IEEE 802.15.4), and supported by TinyOS. The concepts and the wireless sensor network itself (transmitter/receiver/ interface board) are explained, and aspects of the architecture, and of the implementation of the protocols itself are established. Security aspects are also addressed, and the power consumption issues are discussed.

In Section 2, some characteristics and applications used in industry, security services, and military environment are discussed. In Section 3, routing protocols are briefly discussed, security aspects are presented, security imperfections are identified, and energy consumption issues are addressed. In Section 4 the use of TinyDB is discussed. Section 5 presents the various components, e.g., flow, tyre pressure, light, acceleration, temperature, heart beat frequency, and blood pressure sensors. Finally, conclusions are presented in Section 6.

2. Characterisitcs and Applications

At the University of California, Berkeley, and with the support of Intel, an open source operative system called TinyOS was developed for WSNs. As an application example for TinyOS, it is a worth noting that it is already being used by Crossbow to trace automobile parts in industrial environment [1]. Some other examples of the use of WSNs follow:

Safety applications - one of the applications sought for domestic use falls in the area of safety. The distribution of temperature and movement sensors, along the house allows the detection of fires and intrusions. Besides, it can supervise and control children's and elderly people’s movements in the house [1].

Industrial applications - WSNs can be designed and implemented by taking the specificities of each type of industry into account, and several applications can be identified in this framework. WSNs are capable to monitor the quality of the air, and the temperature of a building or on an oven. Besides, it controls the produced goods, the complex machinery set, and the conditions of the production system of a certain or a group of factories [1].

Military applications - Nowadays, military applications of wireless sensors are quite common, mainly because it is difficult to deploy a communication infrastructure in the theatre of operation, e.g., battle field. The installation of a centralised infrastructure, apart from being time consuming, would become a vulnerable network solution (because the destruction of the central node would totally put an end to the entire network) [1].

3. Routing, Security and Energy Consumption Aspects

There are several protocols in the context of TinyOS. As an example the TinyOS Beaconing is a protocol used in Mica Motes wireless sensor nodes of the University of Berkeley, and operates within networks with restricted hardware [3]. The protocol periodically builds the Minimum Spanning Tree starting from the Base Station. The Base Station propagates the message (beacon call) that is spread through the network with the objective of creating the routing tree. As it is a simple and general protocol, its performance is lower than the one of protocols developed for specific applications.

In terms of security, spoofing is the attempt to change or repeat the direction of information by a malicious node [4]. As a consequence, the information may enter into a loop, and the information will never arrive to the sink, as it will continuously be routed through the same set of nodes, causing wasting energy (to send and to receive data). TinySec is the TinyOS cryptography layer, and offers authenticity, integrity and confidence [4]. Actually, TinySec just offers the cryptography of symmetrical keys, and the secret key is distributed in the procedures of programming sensor nodes. The algorithm to produce messages can be any symmetrical one (that can be implemented in TinyOS).

From the essential hardware components needed, the transmitter is the largest energy consumer one. Transmitting is costly and receiving can be as costly as transmitting. Even in idle mode the transceiver wastes energy, and as so it must be put into sleep mode as much time as possible. This is done by implementing an appropriate power management scheme.

4. TINYDB

TinyDB is a query processing system for extracting information from a network of TinyOS sensors. It is only necessary to specify the data and the data rate and TinyDB will perform all the necessary operations to collect, aggregate and display the data on a PC.

In our work, TinyDB was used for reading data from external sensors attached to the sensorial board, e.g., from the flow meter, the tire pressure sensor, and the blood pressure sensor. In these cases, we used the Digi-Key H2163-ND connector. By consulting the datasheet of the sensorial board, it is possible to verify how internal sensors are connected, as well as to know how to read some information. In this work, the option was to connect the external sensors to terminals 42 and 51 (Ground), corresponding to the light sensor in the sensorial board.

5. Components

For the transmission and processing of the signals we have selected the MICAz ZigBee (MPR2400) from Crossbow. This module uses an ATMega 128L in order to collect the data from the network, and to program MICAz motes. The MIB510 interface board was connected to a computer by using the serial port, and the MICAz MTS310 sensor node was then used to connect the sensors that are not on the board via the Digi-Key H2163-ND connector, Fig. 1.

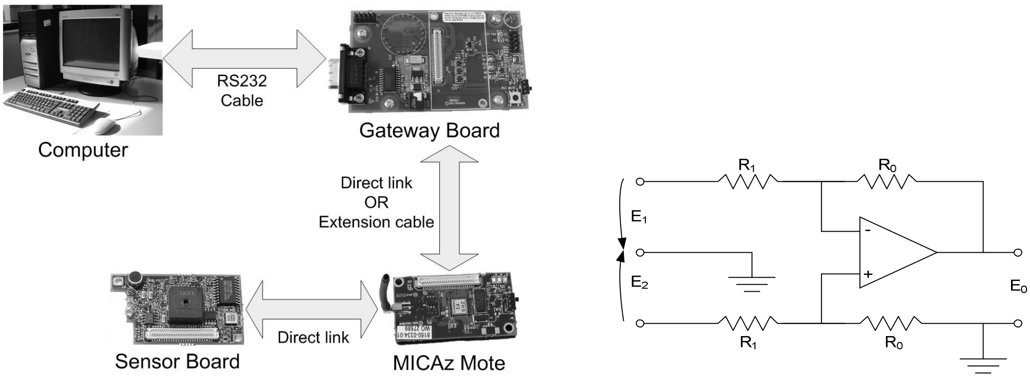

Fig. 1. MIB510 gateway board and related components, and differential amplifier for the pressure sensor.

Flow sensor - We chose a flow sensor with the technical, economical, and weight specifications closest to the requirements. The choice was the RS 508-270, which allows to measuring a wide range of flows, from 0.05 to 10 [l/min]. As the output of the flow sensor is proportional to frequency, we used the frequency to voltage converter LM2917N whose output varies 1 V for each variation of 67Hz in frequency.

Tire pressure sensor - The pressure sensor included in tire pressure reader *Sensor Monza 2 in 1*, is the one that was used. The sensor has four terminals, which indicates that its internal circuit should be a Wheatstone bridge. To have access to the output of this sensor it is thus necessary to measure the voltage in each of its four terminals, in two different cases: sensor reading the ambient pressure, and sensor under pressure (close to the limit is the ideal). To monitor the pressure value we used TinyDB, and the sensor node executes the *TinyDBApp* program. Fig. 1 also represents the scheme of the amplifier used between the MICAz and the sensor. E_0 is the output voltage amplifier circuit, and is connected to the input of MICAz; E_1 is a reference voltage of 0.2 V, which is used to calibrate the sensor; E_2 is the voltage at the output of the pressure sensor; R_1=1.5kΩ, and R_0=10kΩ.

Light Sensor - To verify the state of the automobile lights, a light sensor has been used close to each lamp. In our case, we opted for the light sensor already included into the MTS310 sensorial board, Fig. 2. To collect data from this sensor, the mote is programmed with *OscilloscopRF* application, while being placed onto the zone to be monitored. Another node running the *TosBase* application is placed on the interface board connected to the computer. Then, we can watch lively the variation of the light brightness on the photo-voltaic sensor of the mote, Fig. 2, and this allows knowing how the lamp is.

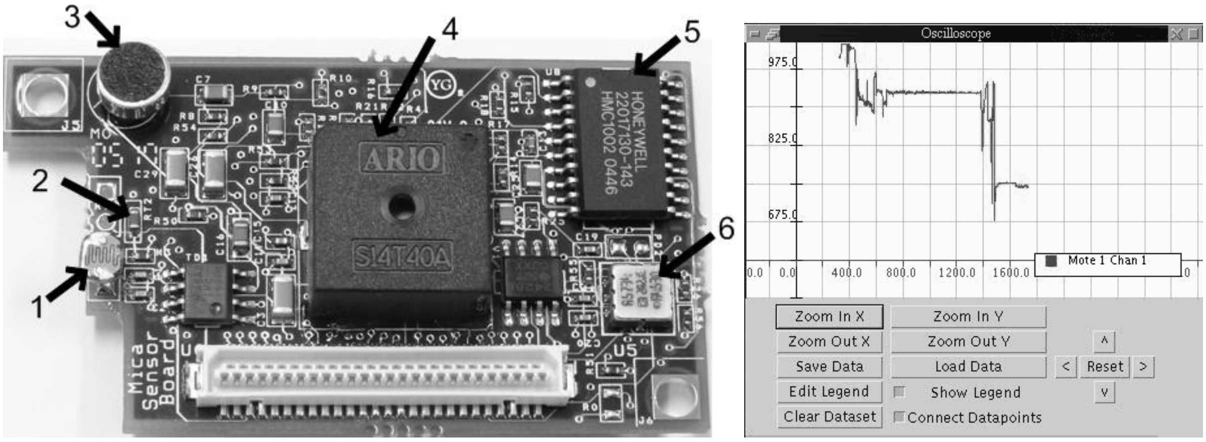

Fig. 2. The MTS310 sensor board: 1) light sensor; 2) temperature sensor; 3) microphone (not used in this application); 4) buzzer; 5) magnetometer (not used in this application); 6) accelerometer, and graphical representation of the light intensity in the mote using the *OscilloscopeRF* application.

Acceleration sensor - The acceleration sensor is incorporated into the MTS310 board of sensors. The signal acquisition is performed by using TinyDB.

Temperature sensor - The temperature sensor should be installed in contact with the driver's body in order to monitor his/her temperature. As the sensorial board already has a built-in temperature sensor, it is considered that this sensor serves our initial objectives. The data of this sensor is read by TinyDB.

Heart frequency and arterial pressure - The equipment selected to measure the heart frequency and the blood pressure (diastolic and systolic) is the prototype for testing and monitoring the biomedical signals developed in the Department of Computer Science of University of Beira Interior, by Prof. Pedro Araújo and his student Pedro Ussman, Fig. 3. This prototype uses an inner tube that is placed under pressure. During its emptying process the pressure is measured through a sensor. This signal is amplified, Fig. 3, and sent to a computer that runs the software to interpret the signal while extracting the blood pressure and the heart beating frequency values. The monitor of arterial pressure is formed by a pump that insufflates air into the inner tube, by a Metrodyn MPS-1001 pressure sensor, by an inductor, and escape valves for the inner tube (fast and slow ones). The connections from the circuit amplifier to the MICAz are the following:

- White wire - analogue signal of the absolute pressure, and it is connected to the terminal 42 of the MICAz;
- Green wire - analogue output signal of the relative pressure, and it is connected to the terminal 42 of another MICAz;
- Yellow/blue wire - digital output signal to turn the pump on/off, and it is connected directly to a relay that is commanded by terminal 9 of MICAz;
- White/blue wire - digital output signal that closes/opens the valve, and connects to the relay that is commanded by the terminal 10 of MICAz.

These processes were controlled by running the *SimpleLedCmd* application, followed by the execution of the following MICAz command

java net.tinyos.tools.BcastInject <parameter>

where the parameter can have the following values: led_on - closes the valve; led_off - opens the valve; led_A_on - activates the pump; led_A_off - turns the pump off.

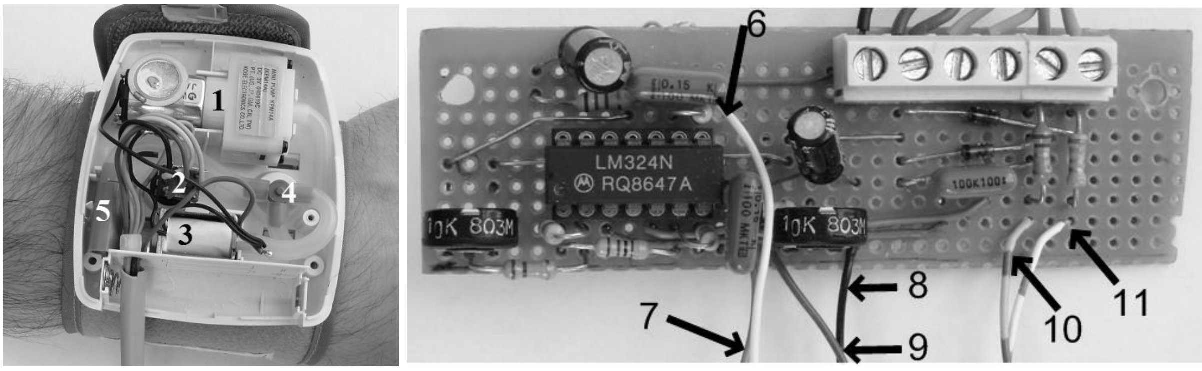

Fig. 3. Arterial pressure monitoring prototype placed in the wrist: 1) pump; 2) pressure sensor; 3) fast escape valve; 4) brace air input; 5) slow escape valve, and circuit for signal amplification: 6) absolute pressure output; 7) relative pressure output; 8) ground; 9) power supply; 10) pump control; 11) valve control.

6. Conclusions

This work addressed the conception of a WSN capable of measuring, process and supply to the user diverse types of information during a car race. Examples are acceleration and fuel consumption, identification of incorrect tire pressure, verification of illumination, and evaluation of the vital signals of the driver. Besides a survey on the concepts, the wireless sensor network itself (transmitter/ receiver/control board) was developed, and aspects of the architecture and protocols were addressed. Security aspects were also identified, and the difficulties and solutions were discussed. Competition cars in a controlled environment is a good scenario (for experimental work) but the evolutions in this field promise a lot in the automobile industry, e.g., for cooperation among cars for road safety purposes.

7. References

[1] http://www.gta.ufrj.br/~rezende/cursos/eel879/trabalhos/rssf1/caracteristicas.htm, Jul. 2006.

[2] Teixeira, Ingrid, Roteamento com Balanceamento de Consumo de Energia para Redes de Sensores Sem Fio, Universidade Federal do Rio de Janeiro, Brasil, Apr. 2005 (http://www.gta.ufrj.br/ftp/gta/TechReports/Ingrid 05/tese.pdf).

[3] Luz, Giulian Dalton, Roteamento em Redes de Sensores, Instituto de Matemática e Estatística, Universidade de São Paulo, Brasil, Nov. 2004 (http://grenoble.ime.usp .br/movel/roteamentosensores.pdf).

[4] Araújo, Rodrigo Cavalcanti, Um Estudo do Impacto do Uso de Criptografia em redes de Sensores Sem Fio (RSSFs), Centro de Informática, Universidade Federal de Pernambuco, Brasil, Sep. 2004 (http://hantuanne.com.br/faculdade/7semestre/redes/criptografia.doc).