

## A Modular Architecture for Energy Efficient Wireless Sensor Networks Nodes

José Catela<sup>1</sup>, Rui Rocha<sup>1</sup>, Moisés Piedade<sup>2</sup>

<sup>1</sup> Instituto de Telecomunicações  
Instituto Superior Técnico, Torre Norte, Piso 10, Av. Rovisco Pais, 1, 1049-001 Lisboa (Portugal)  
Phone/Fax number: +351 218418454 / +351 218418472  
e-mail: [jmcct@lx.it.pt](mailto:jmcct@lx.it.pt), [rui.rocha@lx.it.pt](mailto:rui.rocha@lx.it.pt)

<sup>2</sup> Instituto de Engenharia de Sistemas e Computadores  
Rua Alves Redol, 9, 1000-029 Lisboa (Portugal)  
Phone/Fax number: +351 213100300 / +351 213145843  
e-mail: [mmp@inesc-id.pt](mailto:mpp@inesc-id.pt)

**Abstract.** The current evolution of Wireless Sensor Networks (WSN) has led us towards real world problems such as the power supply of network nodes in an efficient and sustained way. The actual implementation of these networks calls for perpetually powered devices, since the constant changing of the batteries is not a realistic scenario. That is the main motivation behind the proposal of this new architecture, having the low power consumption as a key objective and an intelligent interaction with power supply modules.

The energy efficiency of this platform comes from three different factors: the use of a last generation ultra low-power microcontroller, the use of different communication modes and the intelligent interaction with a harvesting module able to perpetually supply energy to the network nodes. Hence, the modularity of this platform, allowing for independent development of its components, is a very important characteristic.

We present and discuss the major features of the proposed architecture, such as the used processor, how the modularity is achieved and how we intend to implement the energy solutions.

### Key words

Energy Management, Wireless Sensor Networks, Modularity, Embedded Systems.

### 1. Introduction

The low energy efficiency of WSN hardware platforms (usually known as motes), which reduces the possibilities of moving these networks from the lab to the field, has been one of the major concerns in our research group since the moment we started to work in the WSN area. Although an extensive work has been done in terms of software for WSN, the hardware performance and flexibility still plays a major role in this field, as does the efficiency of the power sources.

In terms of the commercially available hardware platforms, the most used solutions, and the ones we intend our device to compare with, are the MicaZ ([1]) and the Telos ([2]). The first has an Atmel ATmega128 as the main microcontroller and the second has a Texas Instruments MSP430F1611. Both were built for IEEE 802.15.4 compliant communication, using a CC2420 [3] from Texas Instruments as the main radio transceiver. However, the major weaknesses identified before cannot be tackled efficiently on both platforms even though MicaZ expands easily to different application areas using custom built sensor boards, and Telos presents better computing performance and lower power consumption.

We decided to design a new hardware platform to be built in-house for addressing some of these problems. The device, called MoteIST, encompasses relevant advantages when compared to the referred commercial solutions, as the improvement in terms of the nodes processing capabilities and its modularity, just to name a few. Indeed, not only have we improved the flexibility, both in terms of its communication interfaces and expandability to different sensor boards, but also on the possibility of using different power sources.

Apart from the reduction of power consumption achieved through the use of last generation microcontrollers, the whole platform architecture is thought to guarantee energy saving mechanisms, which are basically twofold:

- The modularity of the communication interfaces makes it possible to work with high throughput and low consumption radios, simultaneously, and also the development of wake-up radio solutions to minimize even further the power requirements by reducing listening times.

- The use of an intelligent power supply, whose main idea is to retrieve energy from the environment to supply the platform. Additionally, it contributes to the power management with information about the current status and evolution of the batteries as well as the energy sources, allowing for a better power consumption profile and the development of highly energy-efficient network algorithms.

This paper is organized as follows: after this introductory section, the primary hardware platform is presented in section 2; the description of the communication interfaces is addressed in section 3; the intelligent power supply design is presented in section 4; section 5 includes the prototyping results; the last section (the sixth) is dedicated to draw the conclusions about the developed work.

## 2. MoteIST

The development of this platform has been focused primarily on three areas: the potential of the used microcontroller, the modularity of the communication interfaces and the energy management. Figure 1 shows the block diagram of MoteIST, which complies with the referred design goals.

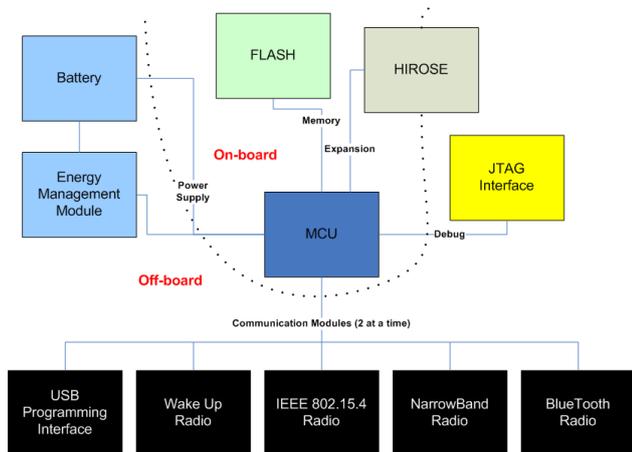


Figure 1. MoteIST block diagram

In Figure 1, the dotted line separates the main board from the ones that can be attached to it. The heart of the platform is a low-power microcontroller designed for embedded systems. It features an energy management module, which includes both hardware and software solutions for energy saving. It allows better debug functionality with a JTAG native interface, while maintaining compatibility with the previous platforms used by our group with the inclusion of a Hirose expansion socket. Data logging is achieved by the classic use of a Flash SPI memory chip. Last, but not least, it is modular in what concerns the communication modes, since USB, ZigBee, BlueTooth or Narrow-Band radio modules can be connected to its two communication sockets, interchangeably.

One aspect worthwhile to mention is the inclusion of digitally controlled analog switches allowing the microcontroller to select which blocks are powered at any given moment. This means these switches can power-off any of the expansion boards as soon as the software decides it is needed.

Having these architectural considerations in mind, it is also very important to cope with compatibility constraints related to our previous work in WSN's application area. Concerning this issue, and since we mostly work with MicaZ motes, all the software that has been developed uses the TinyOS<sup>1</sup> framework. Therefore, the choice of the microcontroller must be guided towards a device compatible with the referred system. The aforementioned mote platforms basically use ATmega and MSP430 as their controlling units. Being the only MCUs supported by TinyOS, our freedom of choice became quite restrained. A better performance and a lower power consumption led us to choose the MSP430F5438 [4] as the heart of MoteIST.

## 3. Communication Interfaces

Regarding the modularity of the communication interfaces, two small connectors (CB connectors) are attached to this mote and in its lines the access to the microcontroller serial communication units and to the general purpose input output and interrupt ports is available. Since the radio and USB modules used on embedded systems usually communicate via protocols as I<sup>2</sup>C and SPI and the MSP430 communication units are reconfigurable, this solution allows for the exchangeability of the devices.

Since most of the sensor networks modules exchange information via Narrow-Band or IEEE 802.15.4 transceivers, these will be the first radio solutions to be implemented for the proposed mote, while keeping in mind the referred modularity allows for the use of other transceivers as needed.

One possible energy saving solution related to the consumption of the radio transceivers that can be easily adapted to MoteIST is the Wake-Up Radio. The main idea behind these circuits is to keep the network nodes in the sleeping state as long as possible, sending beacons to wake them up only when there is information to be sent, as opposed to the idea of calculating when the node should check for new messages by periodic listening. Our Wake-Up Radio (presented on Figure 2) consists on an asynchronous demodulator connected to a low-power microcontroller programmed to decode the wake-up sequence. As soon as such sequence is acknowledged, this microcontroller sends an interrupt to the main board, which enables the primary radio transceiver. Depending on the network configuration, this last transceiver can be any of the indicated in Figure 1.

<sup>1</sup> <http://www.tinyos.net>

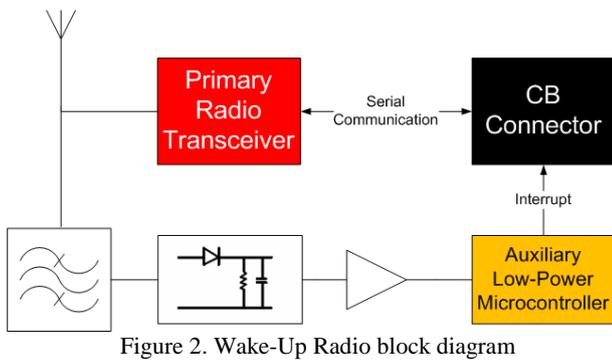


Figure 2. Wake-Up Radio block diagram

#### 4. Intelligent Power Supply

Based on the energy consumption of the wireless sensor nodes calculated on [5] and on the idea behind the design of the Prometheus device ([6]), it has been decided to design an intelligent power supply with energy harvesting capabilities which must be able to perpetually power the networks deployed by the group. From such studies and considering the used microcontroller input voltage (from 1,8 V to 3,6 V), this system is meant to work at 3,3 V and supplying a maximum of 100 mA. The main purpose of this device is to transform the electrical energy generated by a system that harvests it from an environmental source into levels of current and voltage that can be used by a WSN node.

To power the mote, a lithium battery similar to the ones used on mobile phones is used, as their voltages are around the values required by the MSP430. However, these batteries need to be charged at very steady conditions and complying with restrict algorithms. Apart from this, it is worth mentioning the fact that these batteries have a limited number of charging cycles, meaning that they can't work in the charging conditions at all times. To overcome these limitations, it is proposed that the circuit contains an ultra capacitor to retain the energy supplied by the power source. Using this device, the environmental source can be directly connected to the harvesting module without compromising the lifetime of its components.

The last feature of this module concerns the inclusion of an auxiliary low-power microcontroller with serial communication ability, having the objective of supplying information to the main board about the state of the batteries. The controller collects data related to harvesting and consumption and is able to communicate the remaining electric charge, the voltage at the batteries terminals, their aging and the number of available sources and their identification. This information is later used to adapt the power required by and supplied to the node to the available amount of energy.

The referred characteristics of this module are summarized in Figure 3:

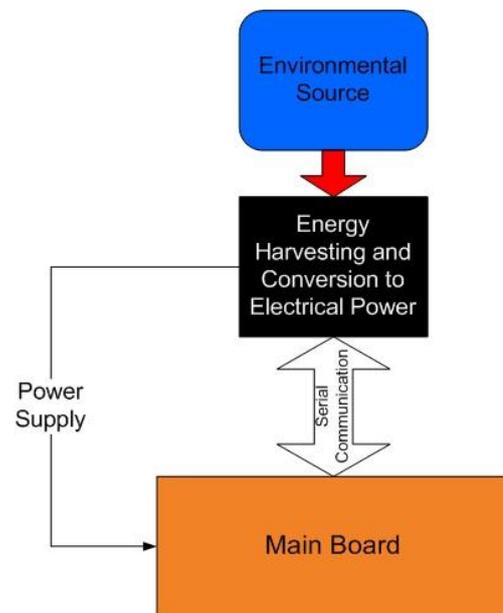


Figure 3. Intelligent Power Supply block diagram

#### 5. Prototyping Results

In order to evaluate the proposed architecture, the following prototypes have been initially implemented: the main board (Figure 4) and the ZigBee communication board (Figure 5) for IEEE 802.15.4 communication using a CC2420 radio transceiver.

The linear dimensions of these prototypes (the main board has 30 mm x 20 mm and the communication board has 116 mm x 65 mm) are twice the target ones (equivalent to MicaZ and Telos) in order to enable in-house fabrication with two-layer technology. This not only reduces the prototype production costs but also allows for better and faster hardware debug.

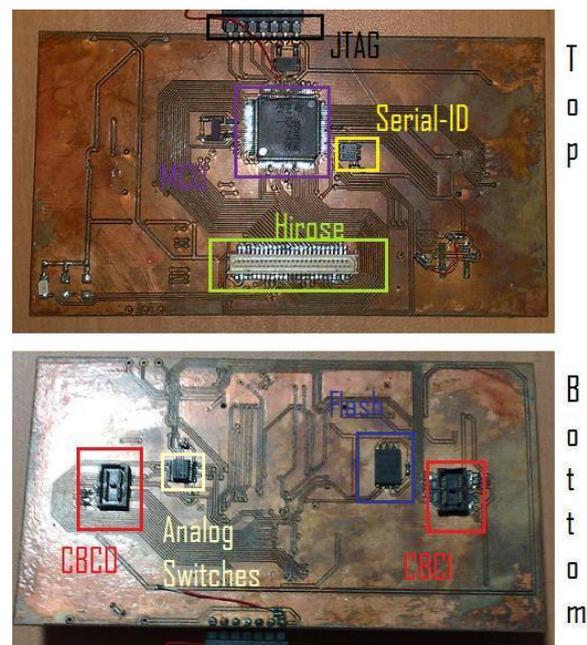


Figure 4. Main board (with MSP430F5438) prototype

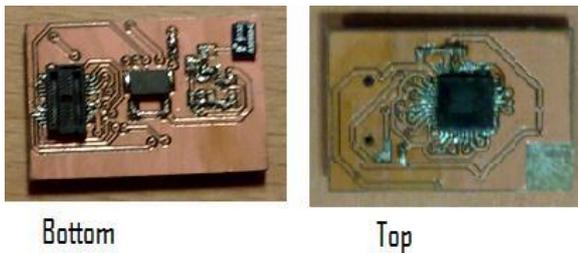


Figure 5. ZigBee communication board prototype

Although the main board presented in Figure 4 is compliant with the requirements of this project, we faced a problem with such approach. As MSP430 series 5 is relatively recent (they were commercially released in December 2008), the open source tool chain<sup>2</sup> belonging to the TinyOS framework does not support it, yet. Therefore, and in order to have a fully usable platform, it was necessary to implement a second main board prototype, using the most suitable MSP430 that still fully supports TinyOS.

To be able to choose correctly the best replacement for series 5, we have compared and analyzed the energy saving features (as well as other relevant characteristics) of all the different suitable MSP430 series, namely 1, 2 and 5. As we already knew, series 5 were the best performer, which is easily observed in Figure 6.

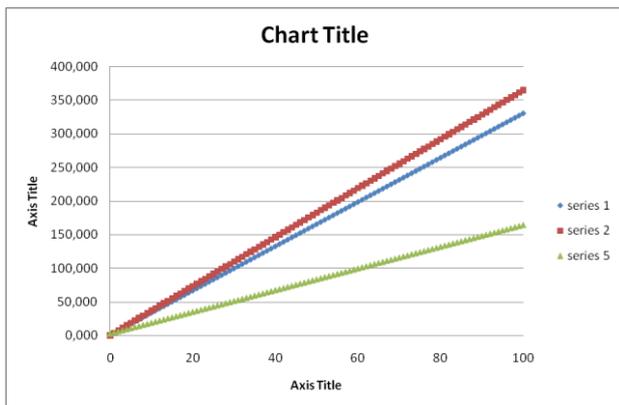


Figure 6. MSP430 series power consumption

Figure 6 displays the current consumption in  $\mu\text{A}/\text{MHz}$  as a function of the duty cycle. For low duty cycle, basically all the three series are similar. As the duty cycle increases, i.e., as the active mode becomes more relevant when compared to the total sleeping time, the low active consumption of series 5 makes all the difference. For large duty cycles ( $>60\%$ ), series 5 current consumption halves the others. On the other hand, series 1 and 2 are pretty much alike, with a slight advantage of series 1, which is less power-hungry and leads to differences that range from  $20 \mu\text{A}/\text{MHz}$  to a maximum of  $35 \mu\text{A}/\text{MHz}$  for 60% and 100%, respectively. Usually, as such large duty cycles are infrequent, the advantage of series 1 in this respect is almost negligible.

Considering this is a transitory phase while the series 5 processors are not supported by MSPGCC, the fact that

<sup>2</sup> <http://mspgcc.sourceforge.net>

series 2 is more recent, it has significantly better performance and its serial communication units are reconfigurable (what helps us meet the modularity requirements), this was naturally the selected MCU for the second version of the main board prototype.

Therefore, the following main board prototype (Figure 7), using a MSP430F2419 [7], has been implemented:

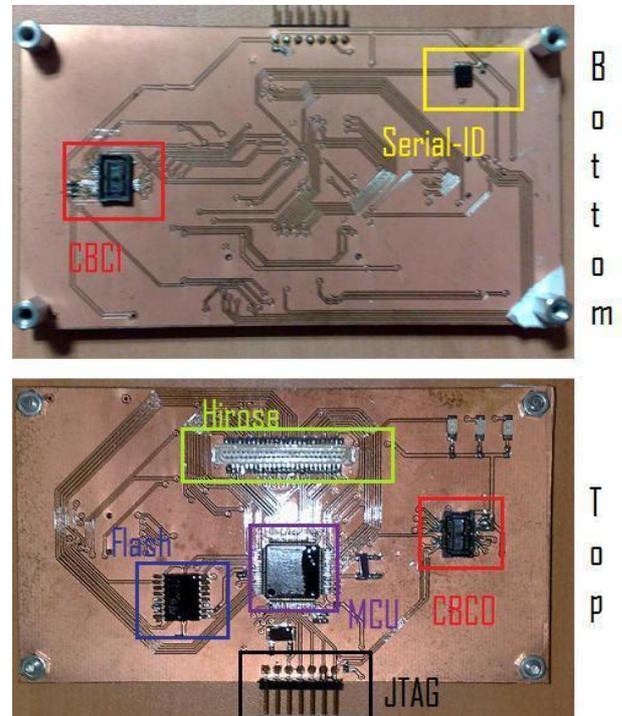


Figure 7. Main board (with MSP430F2419) prototype

Using this second prototype, it was possible to verify the TinyOS compatibility. The graph presented in Figure 8 illustrates the tested applications and how the difference between the program memory required by a MicaZ and by our platform evolves as the complexity of those applications grows. It can be seen that the MSP430 is more efficient for larger applications, which is consistent with its 16-bit architecture, in opposition to the 8-bit architecture of the Atmel processors.

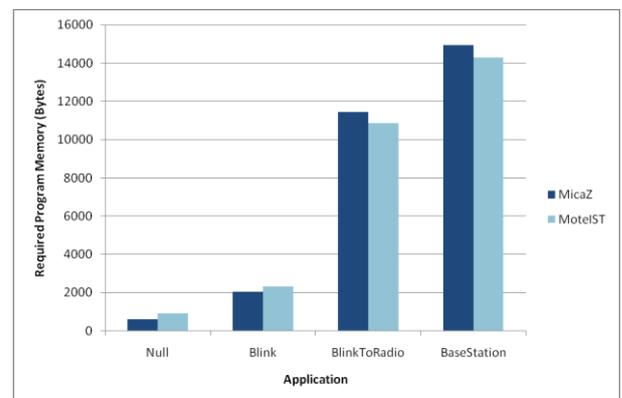


Figure 8. Memory occupied by TinyOS applications

## 6. Conclusions

With this work, we intend to address the identified WSN hardware issues: the low flexibility of the existing platforms and their reduced energy efficiency.

The results of this work are the illustrated prototypes, which prove the usability of our architecture and serve as a start for the implementation of the full platform. Although not all the blocks have been built, the ones already assembled have passed our compatibility tests. Therefore, we can conclude that the presented results are satisfactory and the architectural objectives of the platform were met.

After the implementation of MoteIST, which we intend to have ready for deployment as soon as possible, the implementation of the designed intelligent power supply will allow an increasing energy efficiency of the networks.

The work to be done regarding the power supply module consists on the study of the environmental sources available to harvest, such as solar power, vibrations, radio frequencies and thermal energy. The devices used to retrieve the energy will be integrated in the project after the prototyping and testing phases of the system blocks described in section 2 are complete.

## References

- [1] University of California, "Mica Platform", September 2002
- [2] J. Polastre, R. Szewczyk, D. Culler, "Telos: Enabling ultra-low power research" in ISPN/SPOTS, April 2005
- [3] Texas Instruments, "CC2420 Datasheet", Reference SWRS041B, March 2008
- [4] Texas Instruments, "MSP430F5438 Datasheet", Reference SLAS612, September 2008
- [5] T. Camilo, P. Melo, A. Rodrigues, L. Pedrosa, J. S. Silva, R. Neves, R. Rocha, F. Boavida, "WM<sup>2</sup>Snet deployment: an experimental approach", Book Chapter, in Aggelou, G. (Ed.): Wireless Mesh Networking with 802.16, 802.11 and ZigBEE, McGraw-Hill Communications, pp.352–362
- [6] X. Xiang, J. Polastre, D. Culler, "Perpetual Environmentally Powered Sensor Networks" in Information Processing in Sensor Networks, Fourth International Symposium on, April 2005
- [7] Texas Instruments, "MSP430F2419 Datasheet", Reference SLAS541E, June 2007