Wireless sensor network node with asynchronous architecture and vibration harvesting micro power generator

Yasser Ammar, Aurélien Buhrig, Marcin Marzencki, Benoît Charlot, Skandar Basrour, Karine Matou and Marc Renaudin

TIMA laboratory
46, av. Félix Viallet 38031 Grenoble, France
yasser.ammar@imag.fr

Abstract

This paper presents recent advances in the development of a microsystem designed to be part of a wireless sensor network. This microsystem is developed with two particular technologies: asynchronous circuits and ambient energy harvesting power generator. Asynchronous technologies offer several advantages allowing a global decrease in the power consumption of the node. In addition, the presence of an ambient energy scavenger allows the system to power itself, thus reducing maintenance and increasing the lifetime of the node.

1. Introduction

Wireless sensor networks are composed of a set of autonomous microsystems (or nodes, motes) scattered in a specific environment. Each node monitors physical quantities of its close environment. The measured data are stored and later sent through the self organized network to a base station. The high number of nodes produce redundancy in the measure and by consequence provides a certain amount of fault tolerance to the network. Main applications of these sensor networks concerns the monitoring of environmental physical quantities such as temperature, humidity or vibrations in difference places such as buildings and habitat, industrial or automotive environments.

Currently, most of the wireless sensor networks are composed of macro sized nodes (~10cm²) made with standard ICs[1][2]. Most of them embed non rechargeable batteries that lead to a finite lifetime of the nodes. Research is now targeting the development of ultra low power nodes with small size and MEMS sensors. The key issues for the nodes are the compactness and robustness but mainly low power consumption and low level of required maintenance.

The microsystem is developed as a platform centered on a specific asynchronous microprocessor, as shown in Figure 1. Several modules are commercial off-the-shelf (COTS) devices like the RF module or some MEMS sensors. These modules will be later plugged on this platform regarding the final application specifications. For the hardware architecture, the asynchronous technology is used for several reasons. Firstly, this one must be ultra low power and reactive to environment's events since the activity of the node can be very low and sporadic. Secondly, using this technology, the stabilization of the voltage coming from the harvesting circuit is not a problem any more. And finally delay insensitive circuits can operate at very low voltage.

In order to satisfy the specification in terms of power and lifetime a solution is to use a rechargeable battery connected to a micro power generator (µPG) that scavenge ambient energy. Ambient energy can be found in the form of electromagnetic radiation (light), heat transfer (temperature difference) or vibrations or motion (in the body of a car or a window for example). In the two first cases, the photovoltaïc and thermoelectric effects have been successfully used to scavenge energy. The best example is the watch industry where all these solutions have been used with success to power wristwatches. In our case we decide to develop a vibration energy scavenger that produces electrical energy from vibrations imposed to the body of the microsystem.

This paper details the work in progress toward the building of a wireless sensor node with these two specific technologies: asynchronous electronics and vibration energy harvesting.

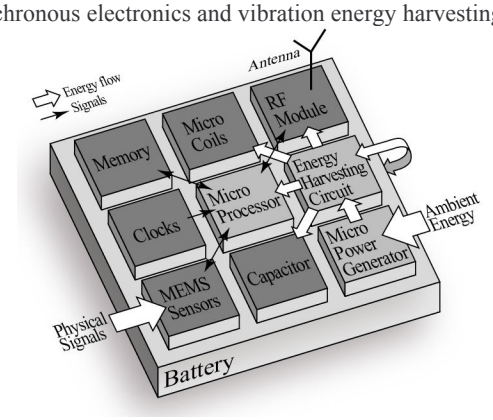

Figure 1: Schematic of the architecture of a wireless sensor node. COTS modules appear in dark grey.

2. Wireless sensor node architecture

2.1. General architecture

The microsystem platform is designed around an asynchronous ultra low power **processor** which runs dedicated software and manages several functions such as monitoring sensors, data storage and RF module driving. Around this processor, several devices take place as illustrated in Figure 1. We can notice:

- The **micro power generator**. This device harvests ambient energy and convert it into electrical energy. A solution will be described in section 3.

- The **power module** (or Energy Harvesting Circuit, EHC) has both to manage the incoming power coming from the generator and to dispatch the energy to the different modules. We will show in section 4 an example of EHC developed for the regulation of the signal coming from a vibration scavenging generator in order to charge the energy storage units.
- **Energy storage units** such as Li-ion micro batteries or super capacitors will store electrical power provided by the generator. These units are used to store the low amount of energy produced by the µPG and the EHC (~1µW) over a long period of time and to provide this stored energy over short period of time when needed for example during RF communications (~1mW).
- The **RF communication module** is composed of a module that complies with actual standard like IEEE 802.15.4.
- **MEMS sensors like** accelerometers or gyrometers, are used because of their low volume and low power consumption; they will be eventually coupled with asynchronous ADCs [11].
- Even with asynchronous technologies the system needs a time reference. A **Clock** from the watch industry is used both for its low power consumption and low volume.

3. Vibration harvesting power generator

In order to power the microsystem, a µPG has been developed to scavenge environmental vibrations[3][4]. The microsystem is placed in a vibrating field like an engine or a mechanical structure. Vibrations are transmitted to the MEMS structure via the body of the microsystem which is mechanically coupled. When excited around its resonance frequency, the µPG produces electrical power. There are different possible transduction mechanisms like electromagnetic induction[6], electrostatic [8] or piezoelectricity [7][8]. The latter has been chosen for both its power density and the ease of integration with standard microfabrication techniques.

Figure 2 shows a schematic of the device which has been manufactured with MEMS technologies. It is composed of a seismic mass made of a Silicon cube connected to the substrate by a cantilever. When excited at its resonance frequency, the system will move out of plane. During the movement, the cantilever is stressed in compression and elongation on the upper and bottom surfaces.

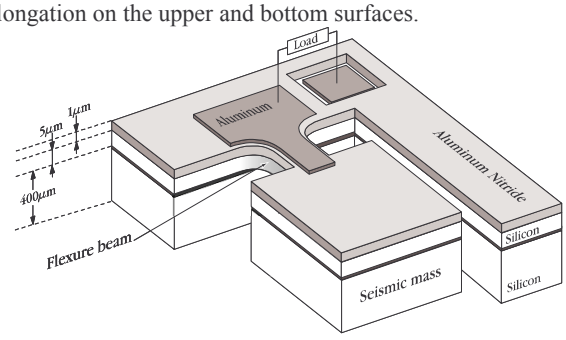

Figure 2: Schematic of the piezoelectric MEMS micro power generator.

The piezoelectric layer, Aluminium Nitride (AlN) in our case, is placed on top of the cantilever and will be stressed during flexion. By consequence, some electrical charges will appear on the surface and be collected by the metallic electrodes and sent to the electrical load or the Energy Harvesting Circuit.

The device has been fabricated with MEMS technologies in cooperation with MEMSCAP and FEMTO-ST, in France. Basically, the process is composed of Deep Reactive Ion Etching (DRIE) steps on both sides of a Silicon On Insulator (SOI) wafer. The piezoelectric layer made of AlN will be replaced in a near future by a thicker layer of PZT, which is a far better piezoelectric material. Figure 3 shows photos of the first prototype.

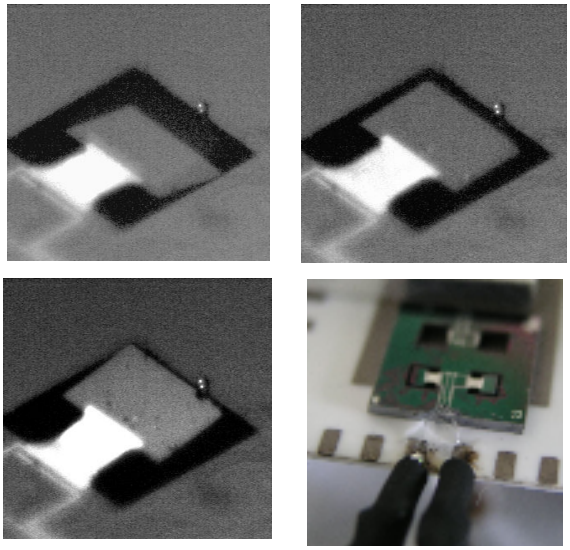

Figure 3: Microphotograph of the vibration harvesting piezoelectric micro power generator excited at the resonance frequency (204Hz). Pictures have been taken with a high speed camera working at 600 fps.

One of the main interests of the MEMS power generator is the possibility to have a large number of small devices connected in series or parallel configuration to increase current or voltage. Another point is the possibility to have several resonance frequencies instead of only one to widen the bandwidth.

4. Energy harvesting circuit

The goal of the Energy Harvesting Circuit (EHC) is to manage the transfer of energy between the µPG and the energy storage modules in order to feed the other parts of the microsystem. The global autonomy of the system is related to the efficiency of this circuit, so it must scavenge the maximum generated energy with minimum losses.

4.1. Standard configuration

The signal coming from the µPG is generally alternative (AC), so in order to charge the battery it needs to be rectified, which means the need of an AC/DC converter. In addition we use A DC/DC converter for load adaptation. The DC/DC converter is driven by a controller that generates a Pulse Width Modulated

(PWM) signal to maximize the transfer of energy between the µPG and the storage modules. The EHC shown in Figure 4 is composed of an AC/DC converter, a DC/DC converter, a controller and a storage element.

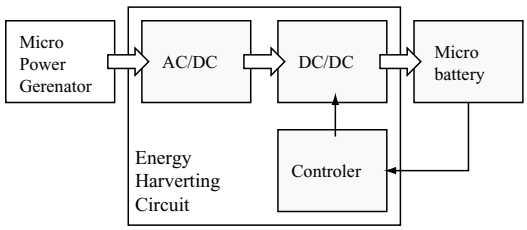

Figure 4 : Composition of the EHC.

4.2. Design optimization

The challenge in the design of AC/DC converter is the expected low voltage and low power signal given by the µPG. An ultra low power structure is needed to fit the power consumption specification of less than 0.1µW. The DC/DC converter is an important element in the power path since the use of such a circuit with a control algorithm increases the harvested energy four times with respect to direct harvesting method (without DC/DC) [5]. Figure 5 shows an example of a buck converter. The study of this type of converter leads to the existence of an optimal value of the duty cycle for which a maximum current flows to the battery. This optimal value is related to the excitation applied to the generator. We can model the piezoelectric generator as a current source in parallel with a capacitor. The value of current is proportional to the excitation applied on the generator. Figure 6 represents the relation between optimal duty cycle value and current given by the generator.

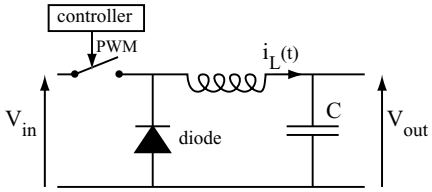

Figure 5 : Buck converter.

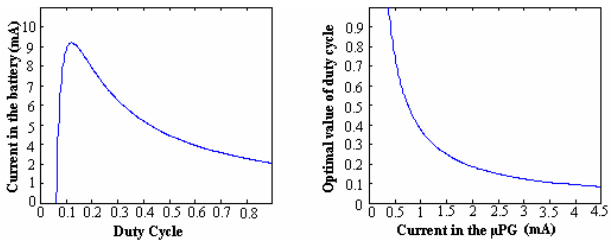

Figure 6 : influence of duty cycle on the output current.

In order to obtain maximum battery current, the duty cycle is incrementally increased or decreased by a fixed steps k.

$$D_{i+1} = D_i \pm k \,. \qquad (1)$$

So we add/subtract k whether we are in the increment/decrement part of the curve shown in Figure 6. The value of k is proposed to be fixed by Ottman et al.[5]. This leads to choose between fast or accurate algorithm. To solve this problem we propose an algorithm to adapt the circuit.

4.3. Adaptive algorithm

The proposed algorithm uses a start value of step k_0. This value is used to change the duty cycle. If this change leads to an increase of current in the battery, the value of k is doubled. Else we go back to the previous value of duty cycle and to k_0. Figure 7 represents the state machine of the algorithm.

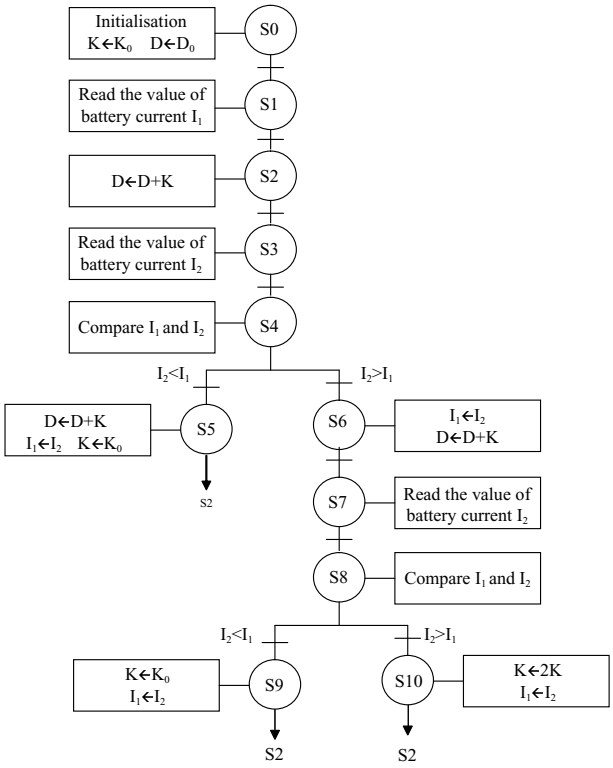

Figure 7 : State machine of the algorithm.

4.4. Experimental results

The bimorph piezoelectric generator, represented in Figure 8, was used as a macro demonstrator. This device is connected to a shaker which generates a sinusoidal excitation. The output of the generator is connected to the AC/DC converter, followed by the DC/DC buck converter and a NiCd rechargeable battery.

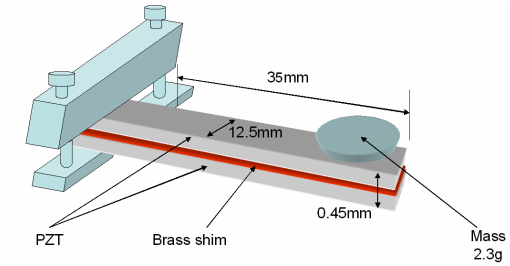

Figure 8 : Structure of the macro sized piezoelectric generator.

A Field Programmable Gate Array (FPGA) circuit from ALTERA has been used to implement the control algorithms. The schematic of the electrical circuit used is shown in Figure 9.

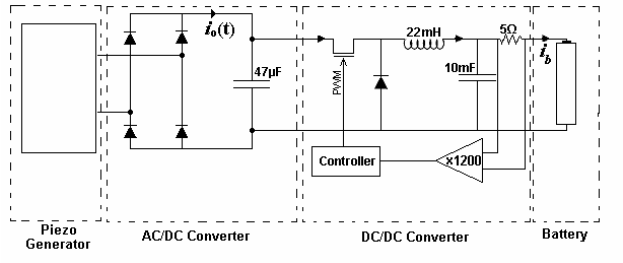

Figure 9: Schematic of the prototype.

The prototype is used to validate the process of energy scavenging, using a piezoelectric element, and to compare our algorithm with a reference one [5]. The piezo bimorph is excited with a sinusoidal acceleration with amplitude of 6.9 m/s^2 at 87Hz. The open circuit voltage is 21V. The measured capacity of the bimorph is 35nF. The frequency of the PWM signal equals to 1 kHz. We report in *Figure 10* the current flowing into the battery for the two algorithms. The initial duty cycle is fixed to 0.7%. The rate of changing of the duty cycle is 0.4%. The time between two variations of duty cycle is fixed to 10 sec. When comparing the results, we can see that our solution leads to a faster adaptation.

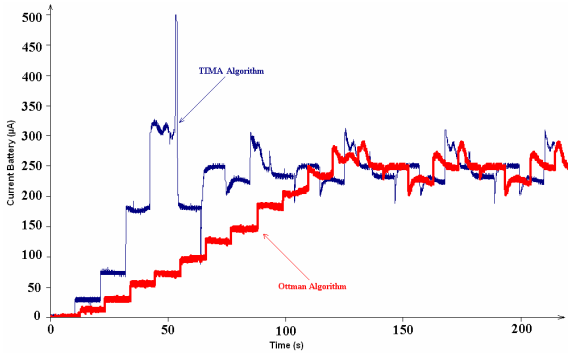

Figure 10 : I_{bat} evolution for the two algorithms : Ottman et al. [5] and the present algorithm.

The work preformed here will be completed by the design of an active AC/DC circuit able to manage voltages that are below the threshold voltage of diodes (~0.3V for Shottky diodes) which is the case of MEMS µPG. Asynchronous technologies will be investigated for the implementation of the control algorithm.

5. Asynchronous platform

5.1. Asynchronous circuit general presentation

There are several asynchronous technologies. Every one are characterized by the absence of a global clock [9]. The communications between the different asynchronous operators are performed by a request/acknowledgment protocol (handshaking protocol) implemented locally [9] as shown in Figure 11. The asynchronous technology used in this project is QDI (Quasy Delay Insensitive) [9]. A QDI circuit behaves correctly regardless of the delay of the gates and wires under the weak assumption of isochronic fork. A fork (a wire that connects a sender to several receivers) is isochronic when the delays between the sender and the receivers are about the same.

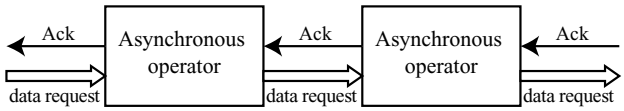

Figure 11 : Asynchronous principle

5.2. Specific advantages for wireless nodes

The asynchronous technologies allow an important power consumption reduction. Indeed, as there is no global clock, the circuit consumes energy only when it is operating, moreover only the parts involved in the computation consumes. The rest of the circuit consumes very little energy (only static leakage) and can immediately be woken-up when an event occurs on its inputs.

A way to even more increase power savings is to use dynamic voltage scaling (DVS) to control power consumption at run time. With synchronous circuits, decreasing the voltage makes the signal transition slower to establish and hence imposes a decrease of the clock frequency. Changing the clock frequency introduces delay overhead due to the synchronization of the *phase lock loop* (PLL) and extra power consumption. With asynchronous circuits, modifying the voltage does not introduce any overhead. The speed of the circuit changes on its own since the circuit behaviour is not sensitive to the rising and falling times of the signals (delay insensitivity).

In particular, this kind of circuit is able to behave correctly regardless of the voltage ripple. That's why the QDI asynchronous technology is especially suited to be used with energy harvesting circuit in which the voltage stabilisation is costly in terms of power consumption and requires large passive elements. In the same way it is possible to use the adaptive body biasing technique that controls the bulk voltage in order to increase the threshold voltage and so to reduce the static leakage [10].

5.3. Hardware architecture

In order to take advantage of such an asynchronous architecture, we want the platform to be completely data-driven and event-driven. To achieve this goal, an asynchronous analog to digital converter (AADC) is used [11]. This one takes samples only if the sensed physical characteristic changes more than a predefined quantum. The samples and the data are saved altogether. This enables a big saving of samples. Then a non uniform sampling theory can be applied [12] to the samples to take advantage of this AADC.

In this case, an event incoming from the sensor propagates through hardware layers from this AADC to the processor. The processor is an asynchronous one and reacts in an event driven way as well. Several processors have been designed such as Aspro[13], SNAP [14] or bitSNAP [15]. An ultra-low

power processor is being designed within the group to fit the requirements of wireless sensor network nodes.

As asynchronous processors are the ideal targets for DVS, we use a co-processor that manages the voltage supply of the CPU according to its need. In fact, each software application indicates its performance needs to the system that is able to know the total speed the CPU must run at to meet all application deadlines. Hence, a speed set-point is given by the CPU to the co-processor that compares this set-point to the real speed of the processor. If they are not about equal, the supply voltage is changed accordingly. First results show that it is possible to get a 45% energy saving on average with general applications using an of-the-shelf DC/DC that only has 60% average efficiency. Using an appropriate DC/DC, the saving could be much better.

In the same way, a power management module regulates the supply voltages of all hardware blocs, and manages the bulk voltage. This control is performed according to a power management policy that defines the trade-off between dynamic and static energy consumption and delays. The algorithm implemented in EHC can advantageously be implemented into hardware using asynchronous technology. The clock being used only for the PWM and sensors that requires time information.

As for the radio, we use ultra wide band signals with the impulse radio technique [17]. This method consists in emitting very short pulses and prevents from using a clock for modulating and demodulating the signals. The present clock is once again only used to quantize the time.

Figure 12 shows a general overview of the hardware architecture.

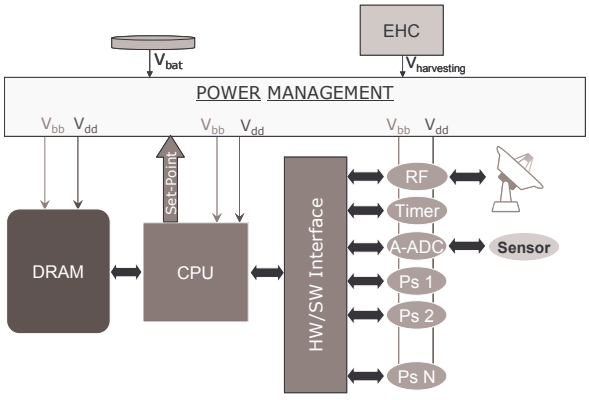

Figure 12 : Hardware architecture of a node

6. Software

6.1. Classical sensor network OS comparison and limitations

An important problem to cope with is to make code execution efficient by minimising the operating system (OS) overhead without removing traditional wireless-sensor network features. Traditional general-purpose multitasking embedded OS were originally developed for the PC platform and have been adapted to embedded systems. Those OS, even the most embedded ones, are too general-purpose to be efficient and the context switching generates expensive overhead that is tolerable for a fast and unlimited energy PC platform or a powerful embedded processor but is not acceptable for ultra low power embedded platforms.A very interesting way to solve this problem is approached with TinyOS developed at UC Berkeley [18]. TinyOS is a reactive OS that does not target a broad range of general applications but only specific tasks for sensor networks. It can be described as follow: external events that occur on the RF or sensor interface propagate upward from the lowest layers till they are handled by the upper ones in an asynchronous way between the different blocks. The component graph of PicoRadio, an ultra-low power wireless sensor node [19] implementing TinyOS, is shown in Figure 13.

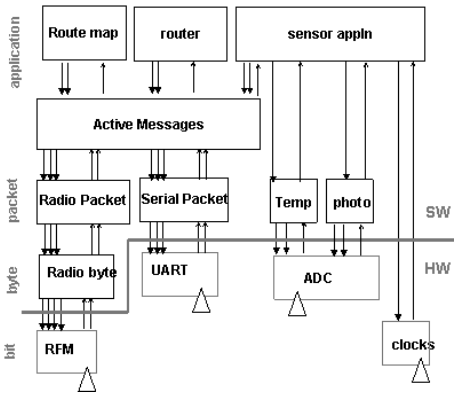

Figure 13 : Component graph of PicoRadio

A performance and power consumption comparison between TinyOS and the general-purpose operating system eCOS [20] is reported in [21]. The result is that the use of TinyOS drastically reduces power consumption and improves performances. Some features (such as virtual memory, dynamic memory allocation, etc.) that are useless for sensor networks are not implemented.

However there are some important limitations of TinyOS. Firstly, it would be advantageous for power consumption and/or performance to implement some components of the application into hardware. Those components could execute specific tasks whereas in TinyOS all tasks go into software. Secondly, there is no way to dispatch software tasks onto different resources. Moreover, some events can be lost during treatment if the event queue is full [21]. Finally, there is no global power-control mechanism; implementing dynamic voltage scaling is a good way to reduce power consumption while meeting the performance constraint but is hardly possible with TinyOS.

6.2. Approach adopted

In our approach, we want to keep the advantages of TinyOS without its drawbacks. In particular, we think that a major feature is to implement some parts of the system into hardware. For example, if a CRC or a Viterbi operator is needed to encode and decode messages, it is advantageously integrated into hardware in order to reduce the power consumption that would be engendered by the numerous

computations performed by the processor to execute equivalent software.
To achieve the integration of hardware blocs within the architecture of the software and to limit the cost of the interface, a common representation of both software and hardware is needed [22]. At the moment, the CHP [23], an asynchronous hardware description language, is used to describe the whole system. This description is then translated into Petri nets [24]. From this description, it is possible to compose each process to make a description of the whole system that can be simulated and partitioned according to the simulation results [22].

The scheduling of the software part of the global description is then performed. To reduce the OS overhead, we choose a static scheduling. This operation gathers all the software processes into one big program that includes the scheduling. In that case, there is no process' stack any more, and the amount of embedded memory is reduced a lot.

A priority mechanism using interrupts can be implemented for software applications that have hard real-time constraints such as the implementation of the MAC layer that must answer very quickly to some requests. It is preferable to have the fewer kinds of interrupts to prevent the overheads that would be generated by the different handlers.

7. Conclusions

In this paper we present recent advances in the development of a wireless sensor network node built around asynchronous electronic technologies. This technology presents several advantages that have been listed. In addition the microsystem embeds a vibration harvesting µPG to power the microsystem from ambient energy.

Acknowledgements

Part of this work has been funded by the European commission in the frame of the European research project VIBES (VIBration Energy Scavenging, IST 507911) of the 6th research framework.

References

[1] Crossbow Inc. http://www.xbow.com/
[2] DUST networks, http://www.dust-inc.com/
[3] Marzencki, M., Basrour, S., Charlot, B., Colin, M., Valbin, L. (2005) *Design and fabrication of piezoelectric micro power generators for autonomous microsystems*, in *Symposium on Design, Test, Integration and Packaging of MEMS/MOEMS*, DTIP'05, Montreux, Switzerland.
[4] Marzencki, M., Basrour, S., Charlot, B. (2005) *Design, modelling and optimisation of integrated piezoelectric micro power generator*, in *Modeling and Simulation of MEMS,* MSM'05, Anaheim, California, USA.
[5] Ottman G., Hofmann H., Bhatt A. and Lesieutre G. (2002) *Adaptive piezoelectric energy harvesting circuit for wireless remote power supply*, in *IEEE Transaction on Power Electronics*, vol. 17, no. 5.
[6] Beeby, S.P., Tudor, M.J., Koukharenko, E. (2004) *Micro machined silicon generator for harvesting power from vibrations*, in *PowerMEMS 2004*, Kyoto, Japan.
[7] Glynne-Jones, P., Beeby, S.P., White N.M. (2001) *The Modelling of a Piezoelectric Vibration Powered Generator for Microsystems,* in *Transducers'01*, Munich, Germany.
[8] Roundy, S., Kenneth Wright, P., Rabaey, J.M. (2004) *Energy scavenging for wireless sensor networks with special focus on vibrations*, Kluwer Academic Publishers, I-4020-7663-0
[9] Renaudin M. (2000) *Asynchronous Circuits and Systems : A Promising Design Alternative*, in *Microelectronic Engineering*, vol. 54, pp. 133-149.
[10] Labonne E., Sicard G., and Renaudin M. (2004) *Dynamic Voltage Scaling and Adaptive Body Biasing Study for Asynchronous Design*, http://tima.imag.fr, TIMA-RR--04/06-01-FR,.
[11] Allier E., Sicard G., Fesquet L., and Renaudin M. (2003) *A new class of Asynchronous A/D Converters Based on Time Quantization*, in *Ninth International Symposium on Asynchronous Circuits and Systems* (ASYNC'03), Vancouver.
[12] Aeschlimann F., Allier E., Fesquet L., and Renaudin M. (2004) *Asynchrounous FIR filters: Towards a new digital processing Chain,* in *Tenth International Symposium on Asynchronous Circuits and Systems* (ASYNC'04).
[13] Renaudin M., Vivet P., and Robin F. (1999) *ASPRO : an Asynchronous 16-bit RISC Microprocessor with DSP Capabilities*, in *ESSCIRC*, Duisburg, Germany.
[14] Ekanayake V., Kelly C., and Manohar R., (2004) *An Ultra-low-power Processor for Sensor Network*s, in *Eleventh International Conference on Architectural Support for Programming Languages and Operating Systems*, Boston, USA.
[15] Ekanayake V., Kelly C. and Manohar R. (2005) *BitSNAP: Dynamic Significance Compression For a Low-Energy Sensor Network Asynchronous Processor*, in *Eleventh IEEE International Symposium on Asynchronous Circuits and Systems,* New York City, USA.
[16] Rios D., Buhrig A., Renaudin M. (2005) *Asservissement de vitesse pour minimiser la puissance consommée par un processeur*, in *ftfc conference*, 18-20 may, Paris, France.
[17] Scholtz R., Win M., (1997) *Impulse Radio*, in Communication Sciences Institute, Dept of Electrical Engineering Systems, Univ. of Southern California, Los Angeles.
[18] TinyOS Group, TinyOS tutorial, http://www.tinyos.net /tinyos-1.x/doc/tutorial/index.html, 2003.
[19] Rabaey J. M., Ammer M. J., Da Silva J. L., Patel D., and Roundy S. (2000) *PicoRadio Supports Ad Hoc Ultra-Low Power Wireless Networking*, in *IEEE Computer*, vol. 33, pp. 42-48.
[20] RedHat, eCOS, http://sources.redhat.com/ecos/.
[21] Li S.F., Sutton R., and. Rabaey J.M, (2001) *Low Power Operating System for Heterogeneous Wireless Communication Systems*, in *PACT 01*, Barcelona, Spain.
[22] Buhrig A., Renaudin M., Barthel D., (2005) *Asynchronous Architecture for Sensor Network Nodes*, to be presented at *Med-Hoc-Net'05*, Île de Porquerolles, France.
[23] Martin, A. (1991) *Programming in VLSI: from communicating processes to delay-insensitive circuits*, in *Developments in concurrency and communication.* Boston, Addison-Wesley Longman Publishing Co., Inc, pp. 1-64.
[24] Renaudin M. and Yakovlev A. (2004) *From Hardware Processes to Asynchronous Circuits via Petri Nets: An Application to Arbiter Design*, in *Workshop on Token Based Computing*, Bologna, Italy.