Toward SoC/SoPC Architecture in Low Power Consumption for Wireless Sensor Networks

Manel Elleuchi, M.Wassim Jmal, Mohamed ABID
CES research unit, National School of Engineers of Sfax
Digital Research Center (CRNS)
Sfax, Tunisia
e-mail : manelelleuchi@gmail.com, wassim.jmal@gmail.com, mohamed.abid@enis.rnu.tn

Abdulfattah M. Obeid, Mohammed S. Ben Saleh
National Center for Electronics, Communications and Photonics
King Abdulaziz City for Science and Technology
Riyadh, Kingdom of Saudi Arabie
e-mail : obeid@kacst.edu.sa, mbensaleh@kacst.edu.sa

***Abstract*—Maximizing sensor nodes' lifetime is an essential issue for wireless sensors networks applications. Therefore, it is necessary to control the energy consumption of the sensor node from the first stages of the design flow. Modeling nodes using system on chip is among the typical solution used to minimize energy consumption. Thus, many studies are currently focusing on proposing new designs and architectures based on system on chip technology. In this paper, we present a new solution, based on system on chip technology, aimed at improving performance. A detailed study of different parts of proposed solution is presented.**

Keywords-component; Wireless sensor networks(WSNs); Power consumption; System on Chip(SoC)

I. INTRODUCTION

Innovation in sensor devices is required due to the wide variety of wireless sensor network applications that have been emerging. With the huge number of domains, low power, low cost and highly integrated System on Chip (SoC) WSN nodes are needed. Nowadays, most of the sensor nodes are developed based on the SoC platforms in order to be very useful for WSNs applications and minimize energy consumption. In this paper, we focus on the study of different proposed hardware implementations of routing protocols in order to minimize power consumption. These implementations are executed in many architectures and platforms such as field-programmable gate array (FPGA) [15] and complex programmable logic device CPLD [10][11][12]. We present, also, the study of different proposed SoC architectures of wireless sensor network nodes. The main objective of our work is to introduce the field of wireless sensor networks, browse the existing solutions and then propose an architecture that minimizes energy consumption based on SoC/System on programmable chip(SoPC). The rest of the paper is structured as follows: The hardware implementation is presented in Section II. Section III present a study of low-power SoC architectures in WSN. Our proposed architecture (SoC/SoPC) is detailed in Section IV. We finish with a conclusion in Section V.

II. RELATED WORK

In literature, several works use reconfigurable hardware as a solution to limit energy consumption in sensors nodes when implementing routing protocols [10]-[13].

In [10] and [12], the authors introduced a platform that uses a reconfigurable device (CPLD) to enhance the processing power of the sensor nodes and reduce overall energy consumption in the common tasks, such as routing and security and header processing. In [10], the used routing protocol is XMesh for its performance when applied in real sensor networks. To implement the proposed platform, they connected a CPLD to a sensor. They used the Digilent X-Board, which connects a Xilinx CoolRunner -II CPLD with the necessary circuit to connect the reconfigurable device. The used sensor node is Crossbow's IRIS with the MDA100 sensor and data acquisition board, which includes a number of sensors and a general prototype space. Multiplication function was chosen to be implemented on reconfigurable device to improve the performance of the routing protocol and to reduce the energy consumption. According to the authors, the measured energy consumption is reduced by 71.49 %, compared to the software approach. From the results that have been found, the authors explain the interest of using CPLD as a hardware accelerator. In [12], a "Pioneering Platform" is introduced, using all the advantages of CPLDs in order to improve the processing power of the sensor nodes and, more importantly reduce the overall energy consumption in heavy demanding tasks such as the routing and processing of the header. This platform accelerates the cost estimation algorithm routing protocol XMesh by 606 times. There is also a reduction of the energy consumption measured by 97%. In addition, the proposed system in [12] can accelerate the control calculation scheme by three orders of magnitude and it consumes up to 96% less energy than the corresponding standard software implementations.

Brokalakis et al propose in [11] the use of a Turbo Code system to increase the robustness and efficiency of communication between end nodes and base stations in the single- hop topologies. In fact, the reconfigurable hardware device was used to perform the coding scheme. The approach in [11] reduces the overall energy consumption of a

 ISBN: 978-1-61208-505-0

node by more than 40 % compared with a Turbo code system implemented in software, as well as more than 70% compared with transmission systems of traditional WSNs that do not support any type of Forward Error Correction. A platform that combines WSNs with a reconfigurable device (CPLD) to reduce energy consumption Turbo coding task was proposed. The reconfigurable device is used for data processing to improve the overall energy efficiency of the WSNs infrastructure. A Turbo encoder was used and the entire coding process has been performed by CPLD. To measure the energy consumption of the platform an oscilloscope was used. The authors in [13] suggest unloading the task of data compression to a CPLD to be connected to the main node of WSN for reducing the overall energy consumption. According to [13] and from experiments, the authors show that it can reduce the energy consumption by a minimum of 46%.

The majority of the related work use CPLD in hardware implementations, but there is another architecture, which can be better than CPLD such as FPGA. In fact, FPGAs offer more logic flexibility and more sophisticated system features than CPLDs: clock management, on-chip Random Access Memory (RAM), Digital Signal Processor (DSP) functions, (multipliers), and even on-chip microprocessors and Multi-Gigabit Transceivers. These benefits and opportunities of dynamic reconfiguration, even in the end-user system, are an important advantage [14]. FPGAs are used for larger and more complex designs and have special routing resources to implement binary counters, arithmetic functions like adders, comparators and RAM. Moreover, FPGA can contain very large digital designs, while CPLD can contain only small designs. The high static (idle) power consumption prohibits use of CPLD in battery-operated equipment. Nevertheless, FPGA idle power consumption is reasonably low. IGLOO FPGAs can be considered to offer revolutionary possibilities in power, size, lead-times, operating temperature, and cost [16]. Among the FPGA implementation that we find in literature is the implementation of the GPSR routing protocol in [15]. In fact, a hardware implementation based on FPGA was used. In the experimental steps, the authors use two different development boards: a Digilent XUPV5 hosting a high-end Xilinx Virtex-5 FPGA device (XC5VLX110T) and a Digilent XUPV2P board with a low-cost Virtex-II ProFPGA (XC2VP30). In both platforms, the FPGA is connected with an Intel Board, which hosts an integrated Intel Dual-Core Atom 330 processor [15]. Via real-world experiments, the authors demonstrated that their system is 31 times faster than an existing CPU-based system, and the overall energy consumption was reduced by more than 90%. The authors explained, also, that the platform could adapt to a different routing protocol in such cases. According to the advantages of FPGA, it can be chosen as a base architecture in our proposed SoC/SoPC architecture, which will be presented in the next sections. A study on the use of SoC for routing protocol in WSNs will be detailed.

III. LOW-POWER SOC ARCHITECTURES IN WSNS

With the goal to make the wireless sensor network nodes small in size, light in weight, cheap in cost, as well as low in power consumption, projects have been carried out to develop sensor node based on SoC technology.

In [1] Wisenet was proposed. It is a platform of low power consumption developed by the Swiss Centre for Electronics and Micro technology for the implementation of wireless sensor networks based on SoC design. It is a distributed wireless sensors network, that combines detection, local signal processing and wireless communication capabilities of short-range in a compact system with low-power consumption. The WiseNET platform uses a design approach that combines a radio with WiseMAC, a control protocol in low power and a complex SoC node. The authors in [1] explained that the WiseNET solution consumes about 100 times less energy with simulation validation than other comparable solutions .

In [2], the authors proposed an approach named EasiSOC, which is a general “sensors node on chip” approach with two typical SoC architectures for different application areas. The basic functionalities, which execute simple tasks, are supported by the first architecture of the sensor node. The second architecture of sensor node supports complex functionalities [2]. The authors presented the design, implementation and simulation of hardware security coprocessor and a program protection mechanism based on (SoC) technology for WSNs. The design is mapped in FPGAs and ASICs. The authors explained that the hardware design overhead is 9.6% lower than previous designs and the design time is only 0.2% of general-purpose processors. These results were obtained with real experiments.

A SoC architecture of a sensor node for the ZigBee protocol was presented in [3]. The proposed architecture in [3] can be used as a stand-alone chip or be incorporated as a component of a larger system at a time. The complete architecture has been designed with each block coded and verified. Blocks like the CRC Unit, AES Unit and the MAC units have been synthesized and preliminary power figures have been obtained. Power and gate count of implemented units(AES UNIT, CRC UNIT, MAC UNIT) are computed. The power Figures are obtained using magama tools on 0.13 micron Technology. The proposed SoC is based on microcontroller.

The authors in [4] present the modeling and simulation of nodes composed by a system-on-chip for applications in WSNs using systemC/TLM. The SoC contains a microprocessor without interlocked pipeline stages (MIPS) processor, a memory, a bus, a timer, a transceiver and a battery. As a case study, the authors conducted a simulation of WSN in star topology and they showed the energy consumption of each node, with a discussion about the computational load. The authors proposed that their approach is flexible and can be adapted to simulate more complex systems and topologies.

In [5], the authors proposed a novel solution, which uses the reconfigurable technology RSoC. They minimized the energy consumption with the use of FPGA nodes that are equipped with "tiny rechargeable units" to recharge the batteries. This solution is designed for specific applications "border control and forest fire monitoring". Energy

 ISBN: 978-1-61208-505-0

consumption is not indicated. Table I presents a summary of the majority of the words presented here.

TABLE I. SYNTHESIS

Ref	Energy consumption	Real Experiences	Simulation
[1]	100 times less energy	-	+
[2]	Low power consumption	+	-
[3]	Low power consumption	+	-
[4]	Low power consumption	-	-
[5]	not indicated	-	-

From this study, we conclude the efficiency of using SoC architecture in WSNs. In fact, the majority of related works confirm the minimization of energy consumption using SoC solution. In this context, we propose a new efficient solution based on SoC/SoPC architecture that reduces power consumption and helps to maximize sensor lifetime, which ameliorates the WSNs operation. This solution will be detailed in the next sections.

IV. PROPOSED SOC/SOPC ARCHITECTURE

In this section, we provide the details of a study in order to propose a novel architecture. To find the most appropriate approach with WSNs characteristics and SoC, an in-depth study is essential for different parts such as the design approach, the SoC design environment, the refinement environment, the validation environment, the processor cores and the operating systems. Fig. 1 below presents the Flow of system design of SoC.

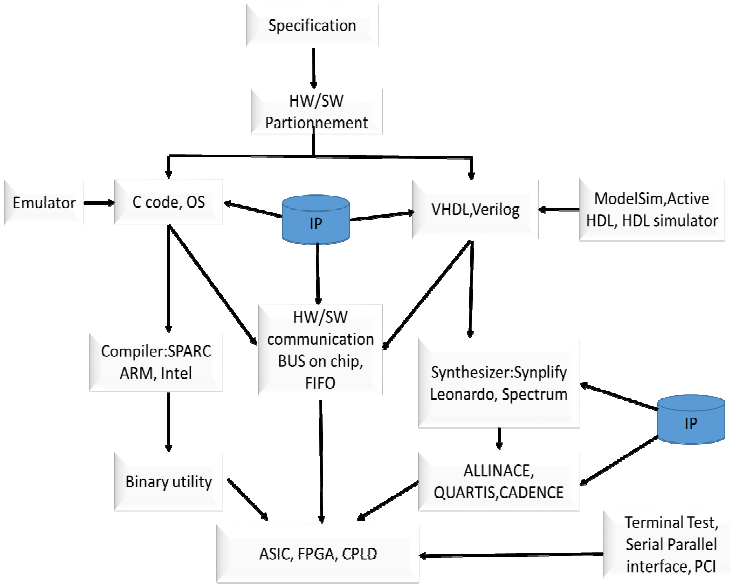

Figure.1. Flow of system design of SoC

Fig.1 presents the flow of system design of SoC, which presents the SoC design environment. Design tools for software and hardware parts form this environment. A detailed description will be done in the next sub-sections for all different parts.

A. *Study of design approaches for routing in WSNs*

The easiest solution is to use a CPU. However, this solution cannot be the only response to the constraints of complexity of operations needed in WSNs. Another solution is to use a special processor for the complex treatments of algorithms, which has many advantages compared to the classical processor (time, energy, acceleration). For the design, there exists many choices to have the appropriate processor. For the processor, one choice is the Harvard processor Digital Signal Processor (DSP), which allows parallelism due to the very long instruction word (VLIW) architecture. The second solution is the possibility to use a specific approach and choose a specific processor dedicated to common parts. This approach is the most effective in terms of treatment, but, due to its specificity, it cannot be used for other applications. This solution implies many restrictions in terms of possible applications. The reconfigurable approach consists in integrating a reconfigurable processor core (as an example FPGA). The main advantage of this approach lies in its flexibility with the possibility to synthesize other algorithms if necessary.

B. *SoC design environment*

This environment is composed of software and hardware design tools. Currently, there are specific language systems that can describe both hardware and software components in a unified manner. These languages require the presence of refinement tools to move to the prototyping level from a high level of specification. Moreover, it is possible to specify the software and hardware parts with two separate specification language.

In our case, we find interesting to adopt the second type based on a separate design, due to the availability of refinement and simulation tools. To construct a system based on SoC, we need to specify all parts of the system.

C. *Software refinement environment in SoC*

SoC need a specific compiler for the architecture of the processor (Intel, Sparc, PowerPC, etc.) for generating executable code. On the other hand, executable programs generated by compilers will be downloaded in electronic target modules (RAM, ROM, Flash memory, etc.), so there is a need to have binary tools that allow specific binary codes for a target electronic module.

In addition to the binary and compilation tools, the software environment supports an operating system that can make the link between the software and the hardware.

 ISBN: 978-1-61208-505-0

D. Hardware refinement Environment in SoC

The hardware environment refinement is based on flow circuit design. Indeed, this environment contains synthesis tools, placement tools and simulation tools. Due to that, the goal of SoC design in our solution is the implementation of complex applications in routing in terms of architecture and functions. It is necessary to have a robust design environment that can support this complexity. For this reason, the researchers are moving towards the use of commercial hardware description language (HDL) simulation tools (ModelSim, Cadence NC-Verilog) and synthesis tools (Altera's Integrated VHSIC Hardware Description Language (VHDL) / Verilog HDL Tool Leonardo Spectrum, Synplify Pro, Synopsys FPGA Express tool, etc.). However, the major problem with these tools is their refinement runtime, which is very long and can take many days for complex applications. On the other hand, the installation of these commercial tools demands efficient execution platforms in terms of operating systems and memory size. In addition to that, these refinement tools target a specific range of FPGA families, for example, there is a specific simulation tool for Altera FPGA family (ModelSim-Altera) or Xilinx (ModelSim Xilinx Edition). Therefore, a study to find hardware refinement tools with an acceptable runtime, non-specific and a consistent execution platform is necessary to have an efficient system in WSNs.

E. Validation environment in SoC

The design of our solution based on SoC requires validating the functionality of the system through all levels of abstraction in order to reduce the time of design and to avoid the propagation of error in the design flow. The validation technique is based first, on the use of emulators to validate the functionality of the system at an abstract level such as the SIS (SPARC Instruction Simulator). Each processor architecture needs a specific emulator. Second, the validation technique is based on HDL simulator. It is possible to test the functionality of HDL module due to the existing library (library for reading binary files) for RTL or logic level. The HDL simulator allows the test after synthesis phase with some libraries (Vital and UNSIM). Nevertheless, the major disadvantage of this simulator is the enormous computing time, especially for complex systems. Thirdly, the validation technique is based on communication terminals, which are communication interfaces between the prototyping cards and the workstation. As examples of communication terminals, we find XSTOOLs, JTAG Programmer and iMPACT of Xilinx. The problem with these communication terminals is the requirement to use a form of strict and well-defined file and a limited range of prototyping board that reduces the flexibility of these terminals.

F. Comparative study of processor cores

For the proposed solution, we will integrate a SoC based on one or more core processors. It is in this context that our architecture will be proposed. It is necessary to make a study about processors in order to find the most suitable for the implementation of the routing protocols. To do this, a comparative study between different synthesizable processors should be done. The architecture of the adopted processor should be studied in order to determine the adequacy of its architecture and the appropriate routing algorithm (Algorithm Architecture Adequacy(AAA)). Routing algorithms have many steps that require a huge amount of calculation and many characteristics that make very interesting the realization on dedicated architectures. The use of embedded software by processor cores implemented on programmable devices and integrated on a single chip is an inevitable trend. These processor cores can be of three types: general purpose processors, processors specific to a domain and the processors that incorporate functional units and a set of specific instructions for a specific application. The choice of the target architecture can be based on three criteria: real-time constraints, the flexibility of the system and the time to market.

In our case, we can find that the most suitable is the general purpose processor due to its flexibility and facility of integration. In fact, this type of processor can quickly and easily make treatments in WSNs and guarantee the possibility of integration of new services, as well as follow the scalability of applications in WSNs. Among processor cores, we find LEON [6], ARM [7], MicroBlaze (Xilinx), OpenRISC1200, Openfire [8] and NIOS [9] (Altera).

A comparative study of the different characteristics and parameters of processors is required to find the most suitable type for the implementation of the routing algorithms on SoC.

G. Comparative study of Operating Systems

For our proposed solution, we need to have an embedded operating system characterized by the limited availability of resources, low price, low power consumption and should be a real time operating system. These operating systems will be used for management of memory, network access time, etc. Among the embedded operating systems, there is Symbian OS, Linux, µCOS, RTEMS, etc. A comparative study of the different characteristics and parameters of operating systems will be done as future work to find the most suitable one.

The proposed SoC architecture is presented in Fig. 2 below.

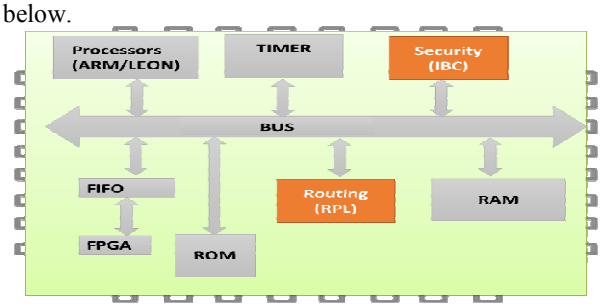

Figure 2. Proposed SoC architecture

 ISBN: 978-1-61208-505-0

The proposed solution includes processor, which can be LEON or ARM, a TIMER, a routing accelerator, a memory (ROM and RAM) and a reconfigurable part. The system components connect to the processor over a bus structure. The implementation and the validation of this proposed architecture will be done as future works.

V. CONCLUSION

The recent developments of WSNs have caused sensor nodes to encounter energy optimization problems. In fact, a sensor node must have the capability of sensing, treating and routing. Routing is the most energy consuming activity in WSNs. In this context, several works proposed many solutions to reduce power consumption and enhance the lifetime of the sensor nodes. Developing a sensor node based on SoC technology has emerged as an effective solution to minimize energy consumption, especially in the routing phase. In this paper, we presented related works according to SoC architecture of wireless sensor network node. We detailed, also, a study about the design of SoC/SoPC architecture in WSN. Following to this study, we propose a novel solution. For the proposed architecture, FPGA can be used for prototyping and testing designed system, which will be studied in future work with the implementation and the validation of the architecture.

REFERENCES

[1] C. C. Enz, A. El-Hoiydi, J.-D. Decotignie, A.-S. Porret, T. Melly, and V. Peiris "WiseNET: an ultralow-power wireless sensor network solution," Analog Circuit Design, pp. 91-122, 2004, 8 doi: 10.1007/978-1-4020-2805-2_5.

[2] Xi Huang, Ze Zhao, and Li Cui," EasiSOC: Towards Cheaper and Smaller ", Proc. First International Conference, MSN 2005, Wuhan,

[3] Ninad B. Kothari, T.S.B. Sudarshan, S. Gurunarayanan1, and Chandrasekhar3, "SOC design of a Low Power Wireless Sensor network node for Zigbee Systems", Proc. International Conference on Advanced Computing and Communications, ADCOM, pp. 462-466 2006.

[4] Heider M.G de Medeiros, J.E.G., da Costa, J.C., Beserra, and G.S., "system level power consumption modeling of a soc for wsn applications" Proc. IEEE 2nd International Conference on Networked Embedded Systems for Enterprise Applications (NESEA), pp. 1-6, 2011.

[5] Ali Elkateeb, " SOC-Based Sensor Mote Design", International Journal of Mobile Network Communications & Telematics (IJMNCT),Vol. 3, No.4, p1, August 2013.

[6] Gaisler, J. www.gaisler.com, the LEON Processor User's Manual.15-17 Nov. 2010,pp. 148 – 153.Version2.4.0 Novembre 2011.

[7] www.arm.com. February 2016.

[8] http://www.altera.com/devices/processor/nios2/ni2-index.html, January 2016.

[9] S. Craven, C. Patterson, and P. Athanas, "Configurable soft processor arrays using the openfire processor," Proc. MAPLD International Conference, pp. 250–256, 2005.

[10] G-G. Mplemenos, K. Papadopoulos, and I.Papaefstathiou, "Using Reconfigurable Hardware Devices in WSNs for Reducing the Energy Consumption of Routing and Security Tasks", Proc. IEEE Global Telecommunications Conference (GLOBECOM 2010), 6-10 Dec. 2010, 10.1109/GLOCOM.2010.5683605.

[11] A. Brokalakis and I.. Papaefstathiou, "Using hardware-based forward error correction to reduce the overall energy consumption of WSNs," Proc.IEEE Wireless Communications and Networking Conference (WCNC), pp. 2191-2196, 2012.

[12] G. G. Mplemenos, P. Christou, and I. Papaefstathiou, "Using reconfigurable hardware devices in WSNs for accelerating and reducing the power consumption of header processing tasks", Proc. IEEE International Symposium on Advanced Networks and Telecommunication Systems (ANTS), pp. 12235-12264, 2009.

[13] G. Chrysos and I. Papaefstathiou, "HeavilyReducing WSNs'Energy Consumption by employing Hardware Based Compression ", Ad-Hoc, Mobile and Wireless Networks, Vol. 5793, pp. 312-326, 2009.

[14] http://asic-soc.blogspot.com/2007/11/what-is-difference-between-fpga-and.html. March 2016.

[15] G-G. Mplemenos and I. Papaefstathiou, "Fast and Power-Efficient Hardware Implementation of a Routing scheme for WSNs" Proc. IEEE Wireless Communications and Networking Conference (WCNC), pp.1710 – 1714, 2012.

[16] http://www.microsemi.com/products/fpga-oc/fpga/igloo-nano. December 2015.

 ISBN: 978-1-61208-505-0