Challenges in the design of the MAC protocols for wireless sensor networks using VHDL

Enio Kaljić* & Abdulah Akšamović**
Department of Telecommunication*
Department of Automatic Control and Electronics**
Faculty of Electrical Engineering at the University of Sarajevo
Zmaja od Bosne, 71000 Sarajevo, Bosnia and Herzegovina
Email: ekaljic@etf.unsa.ba, aaksamovic@etf.unsa.ba

***Abstract*—In the design of MAC protocols for wireless sensor networks (WSN) it is necessary to fulfill some requirements such as low energy consumption, scalability, simplicity, etc. These requirements are not easy to fulfill from the viewpoint of implementation on FPGA or ASIC technologies. Therefore, in this paper we identify some challenges encountered during the design of MAC protocol for WSN. For some of these challenges, potential solutions are discussed. To illustrate the proposed solutions S-MAC protocol is chosen. VHDL design of the S-MAC protocols is experimentally verified on the Altera EP2C5 FPGA development system.**

Keywords — MAC, WSN, FPGA, VHDL, S-MAC, protocol.

I. Introduction

Wireless sensor (and actuator) networks are a special class of wireless networks that are recognizable by their simplicity and orientation to perform domain-specific tasks. The greatest impact on the quality of service of such networks generally have (wireless) medium access control protocols (MAC).

There are several requirements which are necessary to fulfill during the design of MAC protocols in wireless sensor networks. The most important factor is energy consumption since it increases the lifetime of the network. The processes that significantly affect energy consumption are: collisions, permanent listening to the media, listening to the media in IDLE state, transmission of control packets, transmission to unavailable nodes, etc. [1] In addition, important factors are scalability and low complexity.

Implementation platform can pose some restrictions on the design of MAC protocols for WSN. Today, there are two popular approaches: 1) using of microcontrollers and RF communication modules, and 2) implementation of entire system on a single chip. In the first approach, the microcontroller runs a real-time operating system or a dedicated software and communication protocols that control the RF peripherals. In the second approach, all features and functionalities WSN (processor, memory, DSP, protocols, etc.) are implemented on a single chip (ASIC - application specific integrated circuit). The advantage of ASIC in relation to the microcontroller unit (MCU) solution lies in the possibility to optimize the energy consumption by implementing only those functions that are required for the application. Today, the most popular method of ASIC prototyping, is the use of FPGA technology.

The implementation of wireless sensor node on the FPGA device is often resorted to the use of microprocessor architecture (eg. Altera Nios, Xilinx PicoBlaze, OpenRISC etc). In this case, communication protocols are usually executed on the microprocessor instantiated within the FPGA device. However, focus of this paper is the pure implementation of one of MAC protocols logical structure on the FPGA device. The logical structure is described in VHDL.

When describing the MAC protocol in VHDL, it is necessary to make the mapping from protocol design to its logical structure, respecting the previously mentioned requirements: energy efficiency, scalability and lower complexity.

The analysis will be applied to one representative of the MAC protocols in wireless sensor networks - the S-MAC protocol described in [2].

Considering the the S-MAC protocol, its basic functionalities can be observed:

- storing the received data and data ready to send,
- use of timers,
- use of a random number generators,
- checking the validity of the received data using the Cyclic Redundancy Check (CRC),
- power management of its components and the RF transceiver,
- coordination of all the above functionalities using Finite State Machine (FSM).

Realization of each of the above functionalities can be viewed as an isolated challenge.

II. Related work

This chapter will give an overview of existing solutions that can be applied to solve the identified challenges.

A. Data storage

Buffers for storing the received data and of data waiting to be sent can be realized in several ways. Different types of memory on FPGA devices (ROM, FIFO, RAM and dual-port RAM) are discussed in [3]. If the buffers are on data path from the higher layer protocol to the MAC protocol, the use of FIFO registers or dual-port RAM is recommended. ROM can be useful for storing static predefined messages or MAC addresses of wireless sensory nodes. Some MAC protocols

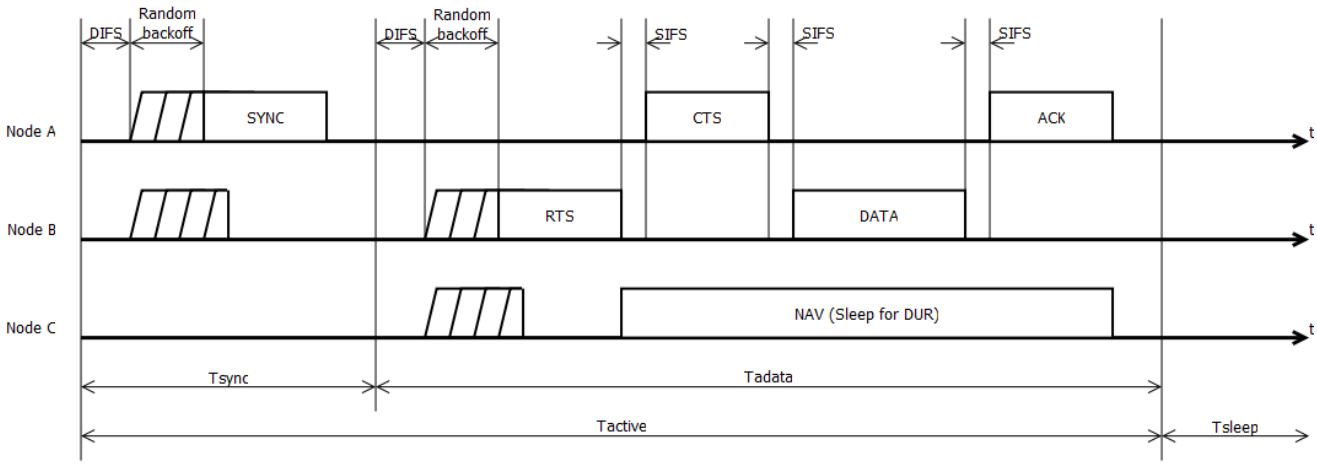

Fig. 1. Timeframe of S-MAC protocol

besides medium access control, also perform flow control. Therefore there is a need to store the message in the window. Proposal of algorithms for 2-D organization of the memory is given in [4], which can be used for the implementation of the data window.

B. Timers

Timers are implemented as simple counters. Depending on the direction of counting, the event of interest may be overflow or underflow. In both cases, the event of interest can be captured by monitoring a single bit, instead of a whole register of the counter. However, in wireless sensor networks, in order to save energy, the two clock sources are often used: low frequency crystal (when the node is in sleep mode), and a high frequency crystal (when the node is in an active state). Virtual High-resolution Time scheme that enables adjustment of the counter according to the clock frequency in order to preserve the same time resolution is proposed in [5].

C. Random number generators

In many MAC protocols back-off mechanism is used for media access in order to reduce the probability of collision. This mechanism involves the generation of a random timing point when a node will access the medium and send data. Based on the state diagrams of different MAC protocols, it can be concluded that there are two sequences of events that lead to the use of back-off mechanisms:

1) higher layer protocol delivers data to be sent to the media as soon as possible (eg. IEEE 802.15.4 CSMA/CA)
2) internal timer signals the moment when the data will be sent on the media (eg. S-MAC).

In the first case, arrival of data from the higher layer protocol is a random event. For the generation of a random back-off the Pseudo-Random Number Generator (PRNG) can be used. In [6] efficient scheme for the implementation of the PRNG is proposed. This approach uses the LUT as an independent shift register and allows the implementation of high-quality generator using a small number of logic elements.

However, if an internal timer (always with the same interval) signals the moment of drawing random back-off, there will be a random number generator synchronization on all nodes. This will disrupt the fairness of medium access, and in the worst case can lead to successive collisions. The solution to this problem is to use a True Random Number Generator (TRNG). In the paper [7] TRNG which exploits jitter of events that propagate in a self-timed ring (STR) to generate a random bit sequence is proposed. It is shown that the delay of each cell in the ring has a jitter with a Gaussian distribution. With proper selection of the sampling interval of output from STR, we can achieve that the probability that we draw a 1 or 0 also undergoes the Gaussian distribution. STR can be implemented on the an FPGA using asynchronous logic cells or FIFO registers.

D. CRC

CRC are the codes for checking errors that have occurred in the transfer of information through an unreliable medium. In the paper [8] serial implementation of some of the most popular standards (CRC-8, CRC-12, CRC-16, CRC-32 and CRC-CCITT) is proposed. Other than serial, parallel implementation of the CRC algorithm is possible as shown in [9]. A parallel implementation is faster than serial, but suffers from problems of long critical paths, which is ultimately reflected in a maximum clock frequency. Serial implementation uses less logic cells. In a wireless sensor network, serial and parallel realization give satisfactory results.

E. Power management

The largest consumer of electricity in the FPGA is SRAM which keeps the device configuration. In [10], [11], [12] the use of a new generation of flash-based FPGA devices is analyzed. It is shown that the energy consumption WSN node can be reduced below 4 mW. Unfortunately, a small number of Flash-based FPGA devices is currently available on the market.

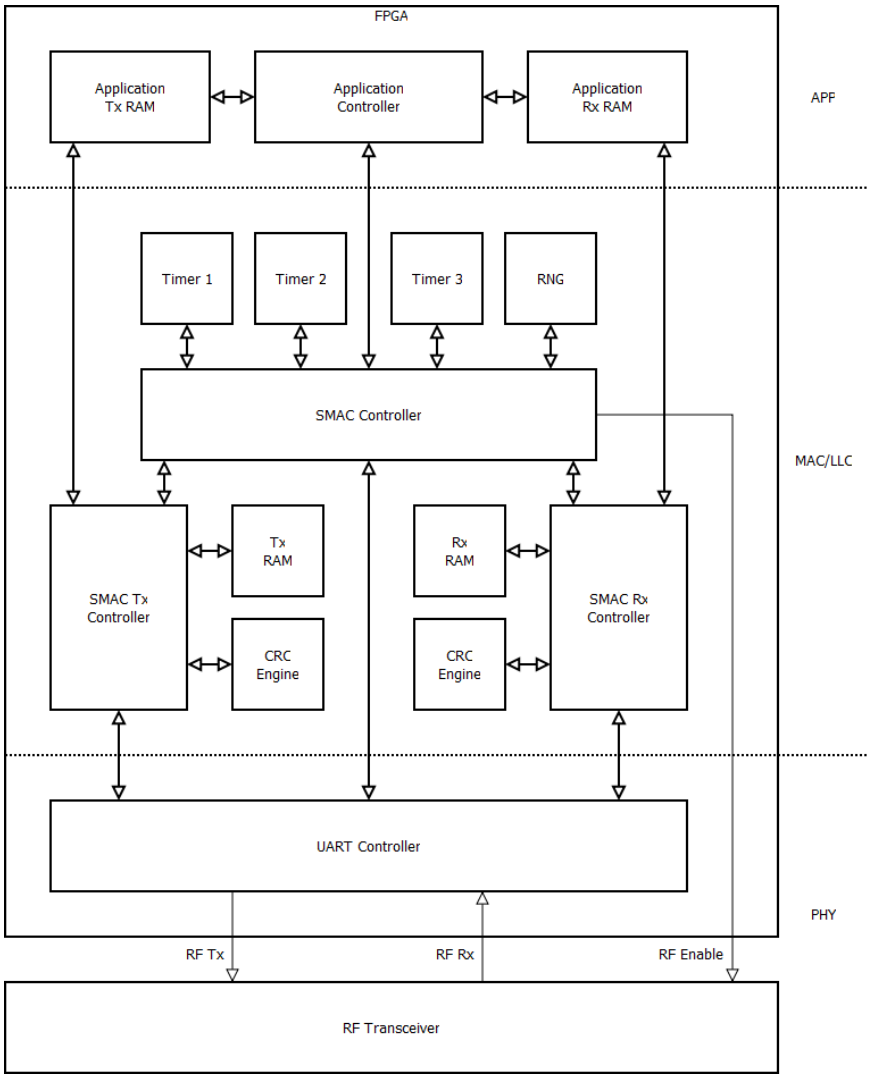

Fig. 2. S-MAC protocol architecture

There are several techniques for energy management of the SRAM-based FPGA devices. In [13] an approach that combines hardware specialization and power-gating is proposed. The solution is based on the decomposition of WSN node into small functional units (micro-task) that are active only when it is needed. Another approach is to set the constraint on the Placement and Routing when synthesizing VHDL design as demonstrated in [14].

F. FSM

For the management and coordination of all functional units of protocol an FSM is used. If we carefully perform a decomposition of logical structure of protocol, then the FSM state diagram can be easily obtained from the state diagram of the protocol. Input and output functions may be implemented using ROM as demonstrated in [15]. In [16] it is shown that it is possible to combine logic cells and ROM for the realization of the FSM's in order to save resources on the the FPGA unit.

III. S-MAC PROTOCOL

In the S-MAC protocol (Fig. 1), one working cycle is divided into two time segments: interval of activity T_{active} and sleep interval T_{sleep}. These periods alternate and have the same time of occurrence for all nodes in a virtual cluster (nodes are fully synchronized). The network can consist of multiple virtual clusters. Interval of activity is divided into two sub-intervals consisting of time slots, wherein the first is used for the exchange of synchronization messages, and the other is used for data transfer. In both sub-interval, nodes are accessing the medium using CSMA/CA mechanism with random counters, which internally runs each node if it finds a free medium. To avoid the problem of hidden nodes, the RTS/CTS control packets are used, by which a certain node informs the other nodes on the duration of its transmission. The nodes that receive the RTS packet, update their NAV counters at the time transmission of node that transmits and go to sleep. Saving of electrical energy is achieved in this

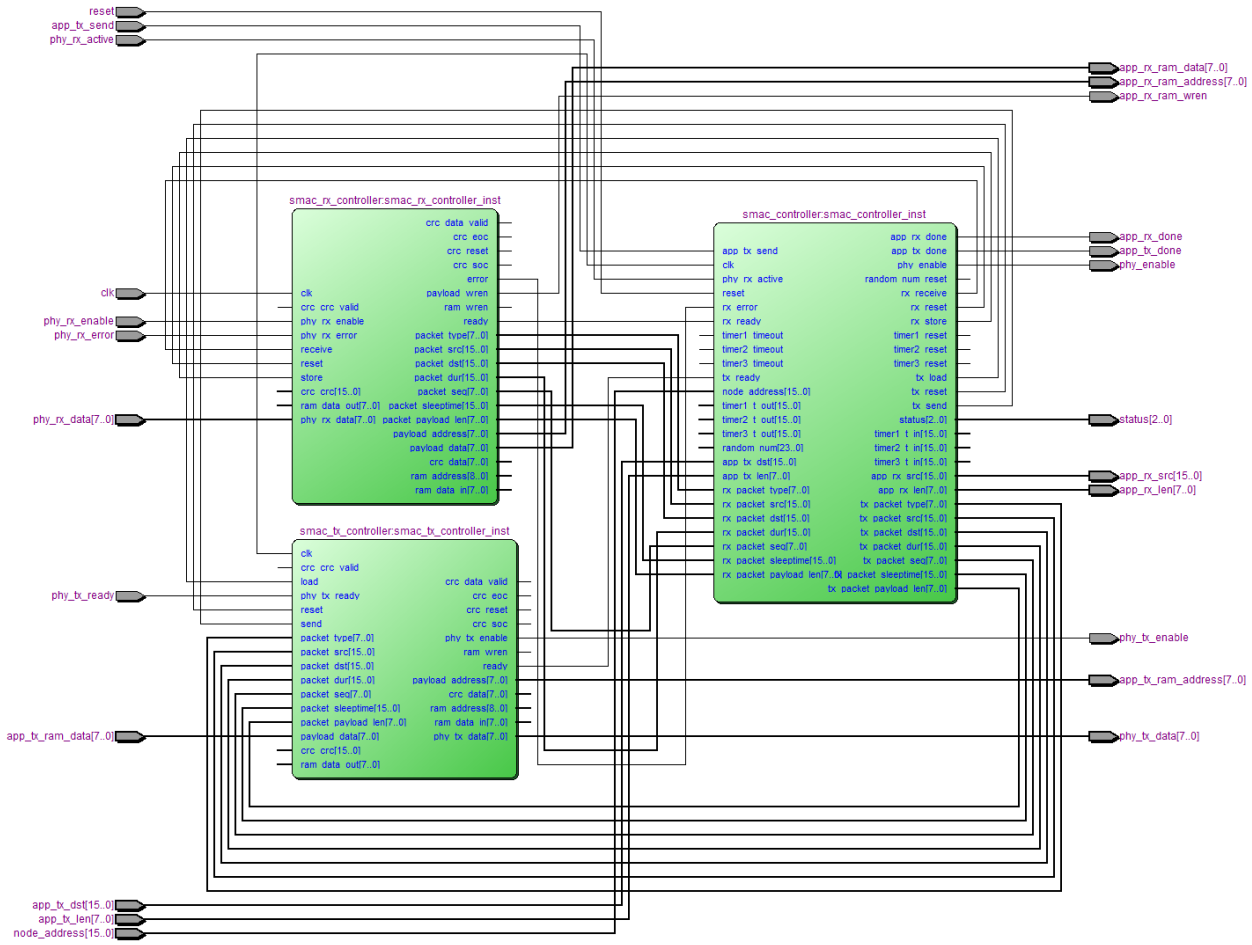

Fig. 3. S-MAC protocol components

way. The protocol uses three counters T_1, T_2 and T_3. The role of counter T_1 is to initiate the transition from a sleep state to the active state and vice versa. The role of counter T_2 is to initiate transition to the next state. Counter T_3 is used to back-off procedure which prescribes CSMA/CA mechanism.

IV. System Architecture and Design of Components

In this paper a decomposition of S-MAC protocols in several functional blocks is presented. Figure 2 shows the architecture of the system. The blocks are grouped into three layers of the OSI/ISO reference model: 1) PHY - physical layer, 2) MAC/LLC - data link layer and 3) APP - application layer. For the simplicity other layers are omitted. The physical layer consists of an RF transceiver module with UART controller. The data link layer of the S-MAC protocol consists of the following functional blocks:

- SMAC Tx Controller - control unit for sending data,
- SMAC Rx Controller - control unit for receiving data,
- Tx RAM - memory buffer for data ready to send,
- Rx RAM - memory buffer for received data,
- CRC Engine - CRC-16 parity bits generator,
- Timer 1, 2 and 3 - timers T_1, T_2 and T_3,
- RNG - random numbers generator,
- SMAC Controller - control unit of S-MAC protocol.

At the application layer is the design of application intended for testing of S-MAC protocol.

A. SMAC Tx and Rx Controller

SMAC Tx Controller (Figure 3) is a control unit for sending data, and is responsible for assembling the frame, depending on the message type, fragmentation and sending frames.

SMAC Rx Controller is a control unit for receiving data, and is responsible for frame decoding.

B. Tx and Rx RAM

Embedded memory blocks of FPGA devices can be configured to operate in several modes:

- single-port RAM,
- simple two-port RAM,
- true two-port RAM,
- shift register,
- ROM,
- FIFO buffer.

For the design of the Tx and Rx RAM components one-port RAM is used. One-port RAM does not support the

simultaneous read and write operations, which assures control unit for sending or receiving data. The data that is being written, are passing through memory to the output. If the output register is not used, the new data are available on the rising edge of clock signal in the same clock interval in which they are written. Buffers for receipt and delivery of data are in 512x8 bus configuration.

C. CRC Engine

CRC Engine component implements a standard CRC-16-CCITT which implies the use of the polynomial $x^{16} + x^{12} + x^5 + 1$.

D. Timers

Timer counts from a given number to zero. Since the main clock signal has a frequency of 50 MHz, and the timer a resolution of 10 ms, an additional component - clock divider is realized. As the clock signal spreads through the FPGA device via the global clock network with constrained delays, this component instead of generating a clock signal, generates enable signal for every 50,000 impulses of main clock.

E. RNG

RNG component is based on the STR. STR can be obtained by making the combinatorial loop, ie. on the input of the logical function we bring its output of. Tool for the synthesis of VHDL code reveal the combinatorial loops and on the output of the logical function puts latch circuit. Therefore, instead of a buffer signal should be used LCELL, which has jitter according to the Gaussian distribution.

F. S-MAC Controller

S-MAC Controller is the control unit for the S-MAC protocol. FSM is directly derived from the state diagram of the S-MAC protocol. One of the important tasks of this component is to manage the power supply of other functional blocks via dedicated enable signals.

V. Experimental verification

For experimental verification of VHDL design of S-MAC protocol, the following components are used:

- Altera FPGA EP2C5T144 development board and
- TLC1101V1-5V RF transceiver.

Figure 4 shows a wireless sensor node made up of the following components. The development system used in this paper contains of:

1) FPGA chip Altera Cyclone II EP2C5T144,
2) chip for serial configuration EPCS4 with flash memory featuring capacity of 4 Mbits,
3) quartz oscillator with a frequency of 50 MHz,
4) 3 LEDs,
5) power supply circuit.

The supply voltage is 1.8V for the FPGA device, and the maximum output current per pin is 40 mA. Supply voltages for FPGA device and 3.3V LVTTL peripherals are provided by voltage regulators AZ1084D-ADJE1 and AZ1085D-3.3E1,

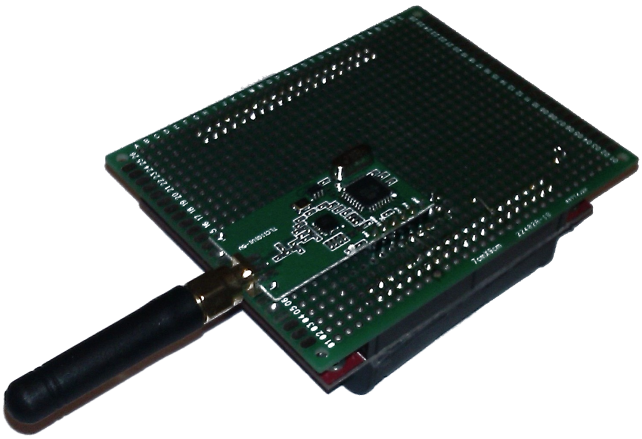

Fig. 4. FPGA based prototype of wireless sensor node

respectively. Programming is possible via JTAG or Active Serial Interface. FPGA device has the following features: 4608 logic elements, 26 M4K embedded memory blocks (totally 119808 bits), 13 embedded 18x18 multipliers, 2 PLLs and 89 input/output pins. For information exchange between wireless sensor nodes RF transceiver based on the CC1101 chip is used. RF module transmits data at a frequency of 433 MHz using FSK modulation techniques with a maximum transmitting power of 10 mW. UART communication interface is used for the communication between FPGA chip and RF module. Power supply of RF communication module is controlled from the FPGA device.

VI. Conclusion

In this paper, the challenges in the design of the MAC protocol for wireless sensor networks using VHDL are identified. For most of them there are adequate solutions, while problems related to power management still persists. The ideal solution is to use Flash-based FPGA devices, but for the well-known SRAM-based FPGA devices, some improvements can be done.

Therefore, special attention is given to the decomposition of the MAC protocol to its functional units that can be turned off when not needed. When it comes to ASIC implementations, the power supply of each component can be implemented separately (power gating), whereas in FPGA prototyping solution is in the clock gating.

Because of the time periodicity of the S-MAC protocol design of random number generator is also proven to be an interesting problem. This challenge was solved by using the TRNG based on the STR for the first time in this paper.

It is also shown that the separation of the core logic of MAC protocol in the separate functional unit facilitates the design of FSM. In this way it is possible to establish direct relationship between the state diagram of protocol and the state diagram of the FSM.

Finally, on the example of S-MAC protocol design of the functional components has been done, and experimental

verification of design is performed on a prototype of a wireless sensor node.

It is interesting to note that for the realization of the entire design the S-MAC protocol, 899 logic elements was used, which is almost half the number of logic elements needed for the implementation of minimal configuration based on the microprocessor architecture.

REFERENCES

[1] R. Abo, K. Barkoui, and H. Zayani, "A probabilistic analysis of energy efficiency of wireless sensor network duty-cycled mac protocols: Application to s-mac and eco-mac," in *Proceedings of the 6th ACM Workshop on QoS and Security for Wireless and Mobile Networks*, ser. Q2SWinet '10. New York, NY, USA: ACM, 2010, pp. 41–48.

[2] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient mac protocol for wireless sensor networks," in *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3, 2002, pp. 1567–1576 vol.3.

[3] Z. Zengrong, "Data processing technology of virtual logic analyzer based on fpga," in *Digital Manufacturing and Automation (ICDMA), 2010 International Conference on*, vol. 1, Dec 2010, pp. 458–460.

[4] P. Diniz and J. Park, "Automatic synthesis of data storage and control structures for fpga-based computing engines," in *Field-Programmable Custom Computing Machines, 2000 IEEE Symposium on*, 2000, pp. 91–100.

[5] T. Schmid, P. Dutta, and M. B. Srivastava, "High-resolution, low-power time synchronization an oxymoron no more," in *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks*, ser. IPSN '10. New York, NY, USA: ACM, 2010, pp. 151–161.

[6] D. Thomas and W. Luk, "The lut-sr family of uniform random number generators for fpga architectures," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. 21, no. 4, pp. 761–770, April 2013.

[7] A. Cherkaoui, V. Fischer, L. Fesquet, and A. Aubert, "A very high speed true random number generator with entropy assessment," in *Cryptographic Hardware and Embedded Systems - CHES 2013*, ser. Lecture Notes in Computer Science, G. Bertoni and J.-S. Coron, Eds. Springer Berlin Heidelberg, 2013, vol. 8086, pp. 179–196.

[8] T. Zhang and Q. Ding, "Design and implementation of crc based on fpga," in *Innovations in Bio-inspired Computing and Applications (IBICA), 2011 Second International Conference on*, Dec 2011, pp. 160–162.

[9] G. Campobello, G. Patane, and M. Russo, "Parallel crc realization," *Computers, IEEE Transactions on*, vol. 52, no. 10, pp. 1312–1319, Oct 2003.

[10] P. Grassi and D. Sciuto, "Energy-aware fpga-based architecture for wireless sensor networks," in *Digital System Design (DSD), 2012 15th Euromicro Conference on*, Sept 2012, pp. 866–873.

[11] P. Dutta, Y.-S. Kuo, A. Ledeczi, T. Schmid, and P. Volgyesi, "Putting the software radio on a low-calorie diet," in *Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks*, ser. Hotnets-IX. New York, NY, USA: ACM, 2010, pp. 20:1–20:6.

[12] Y.-S. Kuo, P. Pannuto, T. Schmid, and P. Dutta, "Reconfiguring the software radio to improve power, price, and portability," in *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems*, ser. SenSys '12. New York, NY, USA: ACM, 2012, pp. 267–280.

[13] M. Pasha, S. Derrien, and O. Sentieys, "A complete design-flow for the generation of ultra low-power wsn node architectures based on micro-tasking," in *Design Automation Conference (DAC), 2010 47th ACM/IEEE*, June 2010, pp. 693–698.

[14] L. Wang, M. French, A. Davoodi, and D. Agarwal, "Fpga dynamic power minimization through placement and routing constraints," *EURASIP J. Embedded Syst.*, vol. 2006, no. 1, pp. 7–7, Jan. 2006.

[15] M. Rawski, H. Selvaraj, and T. Luba, "An application of functional decomposition in rom-based fsm implementation in fpga devices," in *Digital System Design, 2003. Proceedings. Euromicro Symposium on*, Sept 2003, pp. 104–110.

[16] G. Borowik, T. Luba, and B. Falkowski, "Logic synthesis method for pattern matching circuits implementation in fpga with embedded memories," in *Design and Diagnostics of Electronic Circuits Systems, 2009. DDECS '09. 12th International Symposium on*, April 2009, pp. 230–233.