Heiko Hinkelmann, Andreas Reinhardt, Sameer Varyani, Manfred Glesner: A Reconfigurable Prototyping Platform for Smart Sensor Networks. In: Proceedings of the 4th Southern Conference on Programmable Logic (SPL), pp. 125-130, 2008.

A RECONFIGURABLE PROTOTYPING PLATFORM FOR SMART SENSOR NETWORKS

Heiko Hinkelmann, Andreas Reinhardt, Sameer Varyani, Manfred Glesner

Institute of Microelectronic Systems, Technische Universität Darmstadt
Karlstrasse 15, 64283 Darmstadt, Germany
email: hinkelmann@mes.tu-darmstadt.de

ABSTRACT

In this paper, a new concept for a very flexible and modular prototype platform for rapid prototyping of wireless sensor networks is presented. We propose to use a FPGA with high gate count as core of the platform. The FPGA is utilized to attain 3 major goals for the prototype platform: to emulate arbitrary mote architectures even including smart motes with high system complexity, to realize flexible interfaces to sensors and radio transceivers, and to embed versatile debugging and system monitoring functionality in the mote prototypes. The presented prototype platform is suitable to realize complete sensor networks based on different mote architectures, different wireless communication schemes, and arbitrary application domains. The design concepts and implementation aspects of the platform are presented and discussed in detail.

1. INTRODUCTION

For a large number of wireless sensor network applications, functionality is desired that exceeds simple data collection and involves the completion of more sophisticated tasks within the network, e.g., specific sensor signal processing, localization, or object and event tracking. Such *smart sensor networks* therefore characteristically require data processing to be performed locally on the network nodes (*motes*). Because the motes also have to meet stringent energy constraints, mote architectures for smart sensor networks typically comprise dedicated hardware components to achieve low energy consumption for data processing [1, 2, 3], so that the design of smart mote systems may get rather complex.

A design flow for smart motes typically includes the design and verification of synthesizable HDL models of a mote system. Generic prototype platforms then allow rapid prototyping of complete sensor networks using the mote models and allow to test them under realistic conditions before chip manufacturing is started. However, smart motes have some unique requirements that make it difficult to create suitable generic prototype platforms: The prototypes should be very flexible, small, autonomous (possibly mobile in some applications), provide wireless communication, and support a wide range of possible sensors for different application domains. As a main contribution of this paper, several concepts will be discussed for meeting these requirements by using reconfigurable devices for the design of suitable prototype platforms for smart sensor networks. The use of a single FPGA with high gate count is proposed to emulate the complete digital subsystem of a mote and provide a highly flexible solution for sensor interfacing and wireless communication protocol implementation. Furthermore, a concept for using the FPGA configurability to embed debugging and system monitoring functionality into the motes is presented, which allows easy access to detailed information about all motes in a network during run-time.

The rest of this paper is organized as follows. Chapter 2 discusses several concepts for the construction of a generic prototype platform based on FPGAs. Implementation details will then be presented in chapter 3, and a practical application example for prototyping a mote architecture is given in chapter 4. Chapter 5 concludes the paper with a final evaluation of the proposed prototype concepts and an outlook on future research possibilities in this domain.

2. CONCEPTS FOR A GENERIC PROTOTYPE PLATFORM

2.1. Layered Design Approach

The prototype platform concept comprises four separate layers for *sensing*, *communication*, *processing*, and *power supply*, as illustrated in Fig. 1. This layered design approach has proven in the past to be well suited for motes and has been applied similarly in [4, 5, 6]. The concept allows to implement the main functionalities of a mote separately and possibly replace certain layer implementations quickly by alternative designs, e.g. to equip the platform with different sensor types. The signals from the communication and sensing layer are directly mapped to generic FPGA pins at the processing layer, so that the interfacing can be realized very flexibly. This allows to exchange a layer implementation independently of other layers, thus making this modular platform very generic and easily reusable.

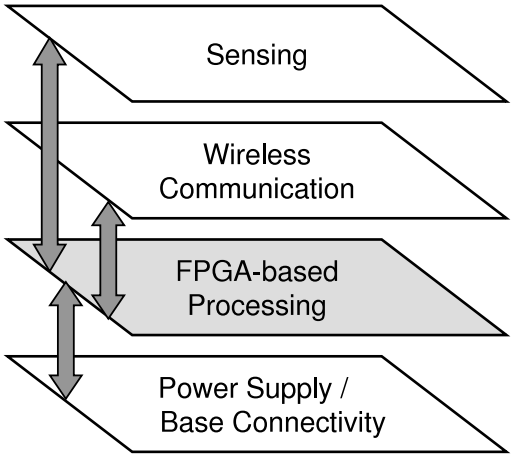

Fig. 1. Layer Model of the Prototype Platform

2.1.1. The Processing Layer

The core of the processing layer will be a FPGA with high gate count, which is intended to emulate the digital part of the prototyped mote system. While simple mote systems typically consist of a small processor and some interface logic only, smart motes often contain additional system components for enabling efficient data processing within the motes, e.g. coprocessors, reconfigurable units [3], or ASIC extensions [1]. The prototyping of smart motes, which is the primary objective of our platform, therefore requires the use of a FPGA with a considerable amount of logic resources. Two additional major objectives will be met by using a large FPGA: it enables very flexible interfacing, and provides advanced debugging and system monitoring capabilities for the prototypes. The concept to map processing, interfacing and system monitoring completely to a single FPGA is a major difference to many previous FPGA-based mote platforms [4, 5, 6], where FPGAs are often used in combination with a microcontroller.

2.1.2. The Communication Layer

For the prototyping and testing of different wireless communication schemes, it is essential to allow the implementation of various protocols on the platform and keep the prototype hardware flexible. Therefore, the use of a light-weight radio transceiver chip providing a physical layer implementation only is proposed. Thus, it becomes possible to implement the higher layers, particularly the MAC layer, within the FPGA and specify them freely. Designers can exploit this freedom to test different communication protocols or to prototype systems that comprise specific hardware components for data processing at the MAC layer (e.g., involving error correction coding or encryption cores), which is of particular interest for smart mote systems. In case the use of a specific radio transceiver is preferred over a flexible solution (because its application to the end product is already known), the layered prototype design easily allows to replace the flexible implementation by a more specific one using the desired transceiver chip.

2.1.3. The Sensing Layer

Besides wireless communication, sensing capabilities are the second specific feature that a mote prototype platform needs to provide. This task is accomplished by the sensing layer. The layer enables the use of arbitrary sensor types with the prototype platform and allows to replace them easily by providing a simple plug-in mechanism. A slot-based concept is employed to connect small sensor modules to the platform. The slots are kept universal and provide the connection to generic FPGA pins, thus enabling to use the FPGA for implementing any digital interface to the sensors. Similar approaches to use field-programmable devices as flexible sensor interfaces have been proposed in [4, 7].

To use a specific sensor, a small PCB module is created for this sensor type which can be plugged into a slot and routes some of the generic signals to the sensor pins. Besides the sensor, the module may also contain few other components, e.g. for A/D conversion if the sensor has analog outputs or for generating sensor-specific supply or reference voltages (whereas common supply voltages of 3.3V and 5V will be provided at the slots through dedicated pins). This modular approach allows to equip the prototype platform with arbitrary sensor types and thus to use it flexibly for a wide range of wireless sensor network applications. Two slots will be provided by our prototype platform to allow simultaneous use of multiple sensors. Additionally, each module may comprise several sensors.

2.1.4. The Power Supply Layer

A drawback of many regular FPGA boards used for prototyping is that they need to be mains-operated. This clearly restricts their suitability for prototyping sensor networks with more than a few nodes, since it always requires to deploy a power network first. On the other hand, FPGAs tend to have a much higher power consumption than average motes. Consequently, powering the FPGA board by batteries will result in short life-times and will require to recharge the batteries frequently. To overcome this problem and achieve a suitable compromise, we therefore propose the usage of two different instances for the power supply layer of our prototype platform – one mains-operated and the other battery powered. The objective of the version supplied by batteries is to provide high autonomy for the mote prototypes. It enables to realize mote mobility and hence a

whole class of applications that involve localization or tracking of moving motes, as well as the general possibility to deploy the prototypes on moving objects or in environments without power infrastructure. Furthermore, it is well-suited for presentations on exhibitions or conferences. As an alternative instance of the power supply layer, we propose the use of a mains-operated base-board. It can be used to conduct long-term tests with the prototypes that exceed the duration of the battery life-time. Also, the base-board will be designed to enable additional debugging features and equipped with an ethernet interface, as will be discussed in section 2.2.

2.2. Debugging and System Monitoring Support

Advanced monitoring capabilities are an essential requirement when the prototypes shall be used for functional verification of sensor networks. Two major problems exist: Internal behaviour of the motes is usually very difficult to observe from outside, and secondly, simultaneous access to many motes in a network requires high efforts and is typically not feasible. Using the wireless link to access the motes may lead to severe link reliability problems, particularly when parts of the wireless communication hardware or protocols have been matter of changes. To account for this problem, a secondary network can be deployed which is not part of the original mote architecture but only used for prototyping [8, 9, 10]. The secondary network operates independently of the wireless link and therefore does not compromise the application under test. In our prototype concept, a wired solution (ethernet) will be used as secondary network provided with the mains-operated base boards. This allows remote access to all motes in a network by a PC simultaneously during run-time, and provides sufficient data rates and reliability to get detailed information about the motes. Particularly when the wireless communication requires debugging, this solution is of great benefit.

The problem of observability of internal mote behaviour can be solved by exploiting the reconfigurability of the prototype. Since the complete mote architecture is emulated in the FPGA and synthesized from a HDL model, it becomes possible to integrate additional monitoring components into the HDL code and map them to the FPGA together with the mote system. Thus, any internal information can be accessed, including direct signals from the sensors and the radio since their interfaces are also realized in the FPGA. This clearly allows versatile debugging options ranging from passive monitoring not influencing the application to actively transferring debug messages and application data. The information can be transmitted over the ethernet link to a PC with according software for sensor network monitoring and debugging. A bi-directional communication then also allows to start, stop, or reset motes remotely during debugging, thus simplifying to conduct tests with large networks.

3. PROTOTYPE IMPLEMENTATION

A prototype platform has been implemented based on the discussed concepts. It is shown in Fig. 2 and Fig. 3 as schematic and photography respectively. The following sections provide detailed information on its implementation and how the concepts have been realized practically.

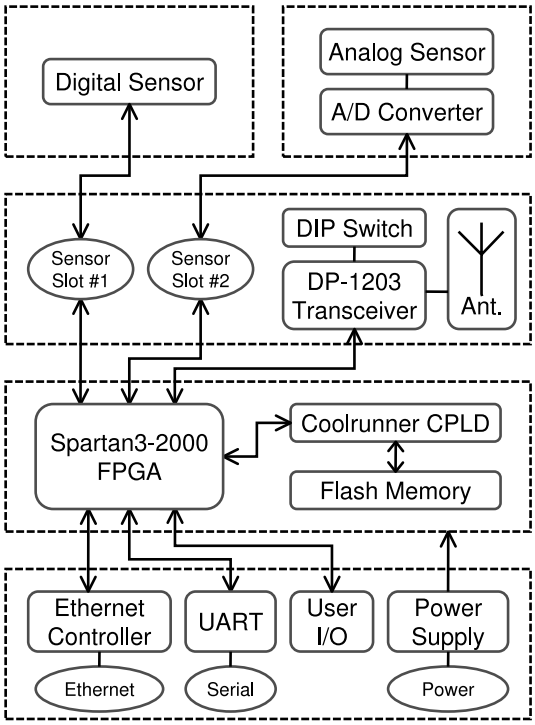

Fig. 2. Schematic view of the prototype platform

3.1. The Processing Layer Implementation

The processing layer of the prototype has been realized by a commercial FPGA board, the Zefant XS3-2000 [11, 12]. It has only the size of a credit card (55mm×85mm) and contains a Xilinx Spartan3-2000 FPGA, a CPLD, and 128 MBit flash memory. Featuring the combination of a small form factor and a high-capacity FPGA makes the Zefant board well-suited for the prototyping of smart motes. In particular, the gate count of 2000k should suffice for the realization of even very complex mote systems on the FPGA, as will be demonstrated for an example mote system in section 4. Similar variants of the Zefant XS3 board are available with lower gate counts of 400k, 1000k, and 1500k, which can be considered as alternative implementations for the processing layer, e.g., to reduce costs.

The non-volatile flash memory on board can be used to store configurations for the FPGA and initial data for the

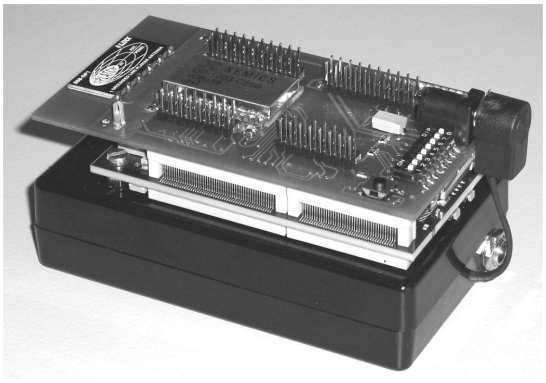

Fig. 3. Photo of the prototype platform, showing the PCB for wireless communication and sensor slots on top, the FPGA board in the middle, and a battery box.

prototyped system (e.g., software for embedded soft-core processors), so the prototypes will be ready at power-up and, most important, can be used without connection to a PC. The connectivity of the FPGA to the upper and lower platform layers (depicted in Fig. 2) is provided by fine-pitched Hirose connectors on both sides of the board, providing a total number of over 250 generic I/O signals to the other layers despite the board's small size.

3.2. The Communication Layer Implementation

A printed circuit board (PCB) has been designed in-house for realizing the wireless communication. It comprises a XEMICS DP-1203 radio transceiver, a planar antenna, LEDs, and two sensor module slots. Since a flexible solution was desired for the radio, the DP-1203 transceiver chip, providing high programmability and implementing the physical layer only, has been chosen. It is a low-power transceiver in the 868 MHz ISM band and features data rates between 1.2 and 152.3 kbps, which is in the typical range of current sensor networks. Its power consumption varies between 0.6 uW in sleep mode and up to 200 mW when sending at maximum output power. The chip can work with arbitrary MAC protocols implemented in the FPGA and provides a *received signal strength indication* (RSSI) function to support the detection of ongoing transmissions of other motes, as used in most CSMA based protocols. Since a MAC address needs to be specified within the FPGA as well, an 8-bit DIP switch has been added to set up a unique value for each mote and thus distinguish between up to 256 motes without the need to use different configuration bit files.

To obtain alternative implementations of the communication layer, only the sensor module slots should be provided in the same manner, while the transceiver chip and antenna may be chosen freely. This is of interest when specific sensor network standards like the IEEE 802.15.4 / Zigbee are intended to be used, for which specific transceiver chips with built-in protocol stacks exist.

3.3. The Sensing Layer Implementation

Two slots are available on the communication PCB for the plug-in of different sensor modules, as can be seen in Fig. 4. Each slot provides 3.3V and 5V supply voltages and 28 generic I/O pins connected directly to the FPGA. This allows to connect a broad range of possible sensor types to the prototype platform, although due to the background of wireless sensor networks, small and cheap sensors with low power consumption are preferred. As sensors are available for measuring light, temperature, pressure, acceleration, sound, magnetic fields, and many other values, we expect that the prototype platform can be used universally. Even GPS modules may be attached. One strength of this modular character of the prototype platform is that it allows to equip several motes with different sensor types and thus build heterogeneous sensor networks. Particularly, this is a prerequisite for smart sensor networks, which often perform data fusion of different physical values.

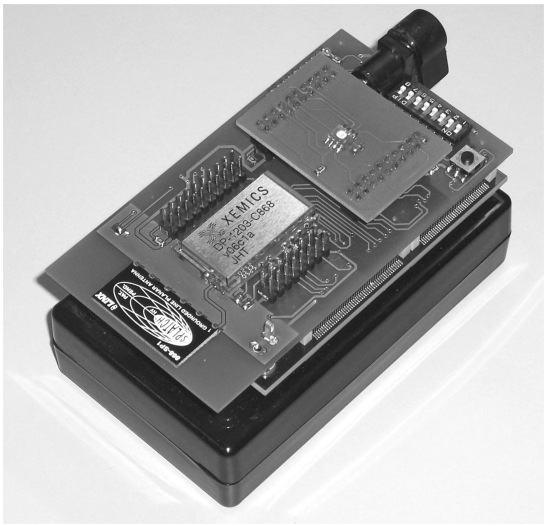

Fig. 4. The prototype platform with a light sensor module attached to one slot

3.4. The Power Supply Layer Implementation

As outlined before, two versions exist for the power supply, one battery-powered and the other a mains-operated base board. The battery-powered version uses four AA rechargeable Ni-MH batteries with a capacity of 2700 mAh each. They are connected in series to deliver a supply voltage between 4V and 6V required by the FPGA board, which converts it to the operating voltages of the Spartan3 device and

most other components. Experimental results show that the power consumption of the prototype varies between 0.7 W and 1.1 W for a clock frequency of 11 MHz (measured for the example system in section 4), depending on the usage of the transceiver and the LEDs. Thus, under realistic conditions, the batteries can supply the prototype for more than 10 hours in practical experiments. This is longer than usually required for presentations and most application experiments. For longer experiments, the base board version should be used for power supply.

Besides power supply, the main objective of the base board is to provide a connection to a PC, either directly or via Ethernet. The base board therefore comprises a JTAG interface for configuring the FPGA, a UART interface for loading the flash memory, and an Ethernet interface. The Ethernet interface is controlled by the FPGA and can serve as user access point for gathering data from the sensor network, to transmit debugging information from the motes to a central PC, or to control the motes remotely during debugging, as outlined in section 2.2.

4. PRACTICAL APPLICATION OF THE PROTOTYPE PLATFORM

The presented prototype platform has been developed in the scope of a research project on smart sensor networks and applied for the prototyping of a new mote architecture incorporating coarse-grained reconfigurable hardware components [3]. This architecture will serve as example to demonstrate how the platform can be used to prototype a complex mote architecture and how it can be suitably integrated into the design flow. As depicted in Fig. 5, the heart of this mote system is a hybrid core combining a 32-bit LEON2 RISC processor core with a domain-specific, coarse-grained reconfigurable function unit (RFU), which is integrated directly into the processor's data path. The RFU improves the energy-efficiency for data processing by up to two orders of magnitude [3]. The system furthermore comprises on-chip memories, peripheral components, and sensor and transceiver interfaces. It is intended to be fabricated as a system-on-chip in 130nm standard cell technology. Fig. 5 also illustrates the possible integration of a debugging unit into this system, which is only done for prototyping (see section 2.2). The debugging unit can gather relevant data by bus snooping, allows to send debug messages from the processor via ethernet, and can remotely start, stop or reset the mote system.

The complete system is designed as VHDL model and can be synthesized either in the target standard cell technology or in the FPGA prototype. Software or configuration data can be hard-coded in the design or loaded via a UART interface to the mote's internal memories. In the prototype, this can be done from the flash memory. While the VHDL

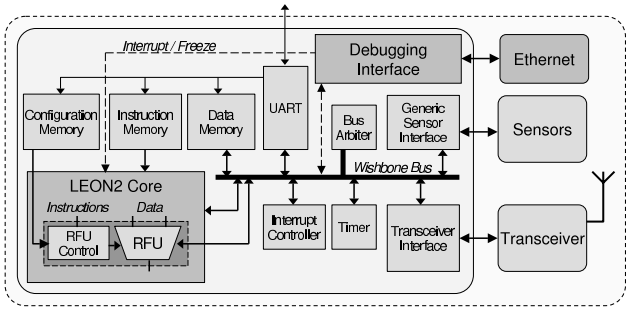

Fig. 5. Example Mote Architecture

model can be simulated to test the hardware functionality in principle, simulations of complete networks are not feasible and would be very time-consuming. The prototype platform therefore represents a significant improvement of the possibilities for analyzing and verifying the functionality of sensor networks incorporating our new mote architectures under realistic conditions. Also, the platform allows rapid prototyping of various applications as well as of hardware design alternatives during the refinement of the mote architecture, for which detailed data can be obtained via system monitoring. This data has been applied successfully to improve our mote design even further.

5. CONCLUSION

A new concept for a prototype platform for smart sensor networks based on the use of a single FPGA only has been presented. The configurability of the FPGA is used to attain 3 major goals: to emulate arbitrary mote architectures even including smart motes with high system complexity, to enable flexible interfaces to sensors and transceivers, and to add debugging and system monitoring functionality to a mote system without interfering with its operation.

The use of a large FPGA with high gate count has turned out to be mandatory for the prototyping of smart motes, which has been demonstrated with an example architecture of a smart mote consuming 60% of the logic resources of a Spartan3-2000 FPGA. This still leaves enough space for adding debugging functionality, which accounts for additional 15% of the logic resources in our system. The problem of power consumption, which is evident for such large FPGAs, could be handled with a feasible compromise by providing the possibility to select between two different power supply versions on demand. In the future, the lifetime of the battery-powered version might be increased even further by the use of upcoming low-power FPGAs (e.g., Actel's IGLOO family) and FPGA power management techniques allowing to switch between active and low-power modes

during run-time.

The prototypes represent a significant improvement of the possibilities for analyzing and verifying the functionality of smart sensor networks in addition to computer aided simulations. They allow rapid prototyping of complete networks and thus to test new mote hardware, communication protocols, and many other system features under realistic conditions before costly chip manufacturing is started. The ability to embed versatile debugging and system monitoring features in the prototype motes moreover enables to gather detailed information on the mote behaviour during run-time and improves system observability radically, even allowing easy access to internal signals of the motes. The availability of such detailed information can help to find and eliminate bugs in the mote hardware, as well as to generate statistics on characteristic system functions and hardware components, which may be used for the refinement of high-level models of the motes or the sensor network system.

Particularly in the domain of smart sensor networks, the prototyping and analysis of heterogeneous sensor networks is of great interest. Our prototype platform allows to implement arbitrary mote architectures on the FPGA, so that motes with different architectures can be realized easily in the same network by still using only one type of prototype platform. Since the sensor modules of each mote can also be exchanged easily, heterogeneity can be perfectly provided for sensing capabilities as well as for system architectures.

Using FPGAs in end products for wireless sensor nodes is a fascinating option and promising research domain [4, 13]. Currently, the high power consumption of FPGAs with high gate count is a critical handicap, but recent developments and trends in the FPGA market lead to the expectation that this problem will be solved in the future. An FPGA-based smart mote can clearly benefit from the same concepts that were proposed for our prototypes, like enabling flexible interfaces or realizing different system architectures with the same mote platform. Hardware functionality may be even altered on demand during run-time by using partial dynamic reconfiguration [14]. Also, remote reconfiguration over the radio link provides new interesting options [4, 13]. Though this feature was not included in our prototype concepts initially, the presented hardware platform allows for remote reconfiguration, which may be an interesting starting point for future research.

6. ACKNOWLEDGEMENTS

This work is part of a research project funded by the German Research Foundation (DFG) under grant no. GL144/25-2.

7. REFERENCES

[1] D. Dietterle, J.-P. Ebert, G. Wagenknecht, and R. Kraemer, "A Wireless Communication Platform for Long-Term Health Monitoring," in *4th IEEE Conference on Pervasive Computing and Communications Workshops*, 2006.

[2] J. Rabaey, M. Ammer, J. da Silva Jr., D. Patel, and S. Roundy, "PicoRadio supports ad hoc ultra-low power wireless networking," *Computer Magazine*, vol. 33, pp. 42–48, 2000.

[3] H. Hinkelmann, P. Zipf, and M. Glesner, "A Domain-Specific Dynamically Reconfigurable Hardware Platform for Wireless Sensor Networks," in *Int. Conf. on Field-Programmable Technology*, 2007.

[4] J. Portilla, T. Riesgo, and A. de Castro, "A Reconfigurable FPGA-Based Architecture for Modular Nodes in Wireless Sensor Networks," in *Proc. 3rd Southern Conference on Programmable Logic*, 2007, pp. 203–206.

[5] B. O'Flynn, S. Bellis, K. Delaney, J. Barton, S. C. O'Mathuna, A. M. Barroso, J. Benson, U. Roedig, and C. Sreenan, "The Development of a Novel Minaturized Modular Platform for Wireless Sensor Networks," in *Proc. 4th Int. Symp. on Information Processing in Sensor Networks*, 2005, pp. 370–375.

[6] D. Lymberopoulos, N. Priyantha, and F. Zhao, "mPlatform: A Reconfigurable Architecture and Efficient Data Sharing Mechanism for Modular Sensor Nodes," in *Proc. 6th Int. Conf. on Information Processing in Sensor Networks*, 2007, pp. 128–137.

[7] D. P. Morales, A. García, A. J. Palma, and A. Martínez-Olmos, "Merging FPGA and FPAA Reconfiguration Capabilities for IEEE 1451.4 Compliant Smart Sensor Applications," in *Proc. 3rd Southern Conference on Programmable Logic*, 2007, pp. 217–220.

[8] J. Beutel, O. Kasten, F. Mattern, K. Römer, F. Siegemund, and L. Thiele, "Prototyping Wireless Sensor Network Applications with BTnodes," in *Proc. 1st European Workshop on Wireless Sensor Networks*, 2004.

[9] J. Beutel, M. Dyer, M. Hinz, L. Meier, and M. Ringwald, "Next-Generation Prototyping of Sensor Networks," in *Proc. 2nd Int. Conf. on Embedded Networked Sensor Systems*, 2004, pp. 291–292.

[10] G. Werner-Allen, P. Swieskowski, and M. Welsh, "MoteLab: A Wireless Sensor Network Testbed," in *Proc. of the 4th Int. Symp. on Information Processing in Sensor Networks*, 2005.

[11] S. Schirrmann, "Zefant XS3 FPGA Module Users Manual, version 1.7," available online at http://www.simple-solutions.de, 2007.

[12] Trenz Electronic, http://www.trenz-electronic.de.

[13] D. Efstathiou, K. Kazakos, and A. Dollas, "Parrotfish: Task Distribution in a Low Cost Autonomous ad hoc Sensor Network through Dynamic Runtime Reconfiguration," in *Proc. 14th Annual IEEE Symp. on Field-Programmable Custom Computing Machines*, 2006.

[14] M. Majer, J. Teich, A. Ahmadinia, and C. Bobda, "The Erlangen Slot Machine: A Dynamically Reconfigurable FPGA-Based Computer," *Journal of VLSI Signal Processing Systems*, vol. 47, pp. 15–31, 2007.