System on a Programmable Chip Design of a Wireless Transceiver Prototype for Smart Grid Applications

Dan Shan[1], Weidong Xiang[1], Paul Richardson[1], and Hua Qian[2]

[1] ECE Department, University of Michigan, Dearborn, MI 48128, USA
[2] Shanghai Institute of Microsystem and Information Technology, Shanghai, China
{danshan,xwd,richarpc}@umich.edu,hua.qian@shrcwc.org

Abstract. When compared to digital signal processors (DSPs), filed programmable gate arrays (FPGAs) have more computation power where functions are realized by hardware modules operating in parallel, instead of by instructions executing in sequence in DSPs. In this contribution, a large scale low duty smart utility network (SUN) radio upon the IEEE 802.15.4g standard (draft) for smart grid applications is implemented. Unlike the current wireless sensor network (WSN) using Zigbee radios, which is mainly used in a local scale, say less than tens of meters, SUN focuses on large scale WSN, which is able to access to the infrastructure at a distance up to kilometers. At the moment of composing this article, the IEEE 802.15.4g standard is still under developed. However, the narrow band, less than 1MHz, orthogonal frequency multiplexing division (OFDM) technology is identified to be adopted to realize robust link in large scale outdoor environments. Funded by Argonne National Laboratory (ANL), Department of Energy (DoE) in 2011 and then collaborated with Shanghai Research Center for Wireless Communications, a SUN radio prototype based on an Altera FPGA chip is firstly implemented and evaluated. The system design, baseband signal processing, medium access control (MAC) protocols and interfaces to computers are detailed in the article. An embedded processor within the FPGA chip is used to realize MAC protocol simplifying the system design. Moreover, such a FPGA based prototype serves as a universal *system on programmable chip* for a board range wireless communications. Experiments results demonstrate satisfied processing latency and bit error rate (BER) for smart grid applications.

Keywords: SUN, IEEE 802.15.4g, FPGA, Smart Grid.

1 Introduction

By the aid of an intelligent sensing and control system, smart grid possesses enhanced resolutions of energy accessibility, flexibility, manageability, and reliability than traditional power grid current in use. However, smart grid requires intensive effort including research, design and development of smart equipment, communications sbusystems, data management center, security and applications

H. Qian and K. Kang (Eds.): WICON 2013, LNICST 121, pp. 85–95, 2013.

softwares [1]. Like the Internet, smart gird is a network of networks, subsystems and subnets of which have different configuration and ownership and need to interconnect each other. Such a realization highly relies on the effective network and media access control (MAC) protocols. Moreover, as an important nation wide infrastructure, smart gird is expected to accommodate with diverse applications, which will be added on or defined in the decades to come. The applications have distinguished requirements on network topology, transmission range, data rate, delay, bit error rate (BER) and package error rates (PER) under various channel environments. Unfortunately, the current wireless sensor network (WSN) based on local Zigbee radios cannot offer the claimed feature [2]. Instead, a geographically diverse and large scale low duty WSN should be presented to provide seamless connectivity among the equipments in smart gird.

In 2008, IEEE 802.15 smart utility networks (SUN) Task Group 4g was founded to focus on the amendment to Zigbee and provide a global standard named IEEE 802.15.4g, one of the main applications of which is smart grid [3]. The new standard upgrades data rate from 250kb/s to 800kb/s through adopting orthogonal frequency multiplexing division (OFDM), an advanced modulation scheme that has been widely applied. OFDM outperforms traditional modulation schemes over deep fading channels. It is believed that the IEEE 802.15.4g standard adopting a mature and market success technology will take over smart grid communications networks. Since the standard is still under development, few effort has been reported on the the hardware prototyping and evaluation of SUN radio, which motivates our work.

Wireless systems are normally prototyped upon digital signal processor (DSP) or/and field programmable gate array (FPGA). In this effort, we select FPGA due to its more powerful computation and versatile processing ability. A *programmable system on chip* design strategy is adopted to realize the SUN radio including baseband processing, MAC protocol and data interface to computers, all upon the IEEE 802.15.4g standard (draft). The prototype can serve as a hardware platform to evaluate the performance of SUN as well as the benchmark for next step application specific integrated circuit (ASIC) design.

An overview on the hardware solutions to the prototyping on wireless communications systems are discussed in Section 2. Section 3 details the design of SUN radio including both the hardware and software while Section 4 presents corresponding experimental results. Finally conclusions are withdrawn.

2 Hardware Solutions to Prototyping on Wireless Transceivers

Due to the high computation load of the digital signal processing algorithms including fast Fourier transformation (FFT), viterbi decoding, timing synchronization and channel estimation, wireless communications systems are normally built upon ASIC. However, ASIC is more suitable for radios that have massive product. At its early development and prototyping stage, a general wireless prototype is a more feasible solution.

2.1 Computers Based Wireless Prototyping Solution

Matlab is a convenient tool to realize baseband signal processing. The built-in functions such as fast Fourier transformation (FFT) and matrix operations make easy the implementation. We normally start with Matlab simulation of a wireless communications system and then implement it upon a prototype. A typical computer based prototyping solution to wireless systems is shown in Fig. 1. Through the peripheral component interconnect (PCI) interfaces, data can be exchanged between the analog-to-digital (ADC) and digital-to-analog (DAC) boards and computers. For example, the digitizer board from the *Signatec* Company [4] and the arbitrary waveform generator (AWG) from the *Strategictest* Company [5] both have drivers and application programming interface (API) to Matlab. The transmitter and receiver locates within computers resulting in a compact solution. Normally, PCI supports high-speed data transfer between wireless systems and hard disks of computers.

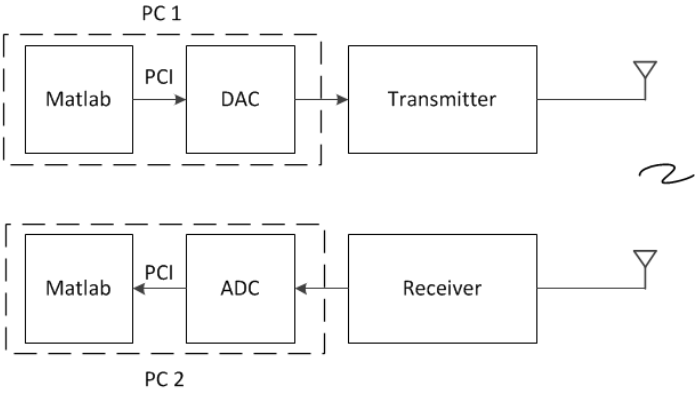

Fig. 1. The computer based wireless prototyping solution

There exists several disadvantages in this solution. The foremost is the limited computation power and comparably slow operation speed. It may take up to tens or hundreds ms for Matlab functions to complete the processing for one package. Such solution normally is not suitable for high-speed transceiver running in a real-time mode. Another issue is its lump size consisting of at least two desktop computers with PCI interfaces, inappropriate for portable applications.

2.2 DSP Based Solution

The architecture for DSP solution is shown in Fig. 2. The advantage of the solution lies on that there are a lot of open source codes and free libraries for typical baseband processing on various DSPs, including FFT and synchronization. *GNU Radio* is one of such toolkits [6]. Since the processing is completed by software,

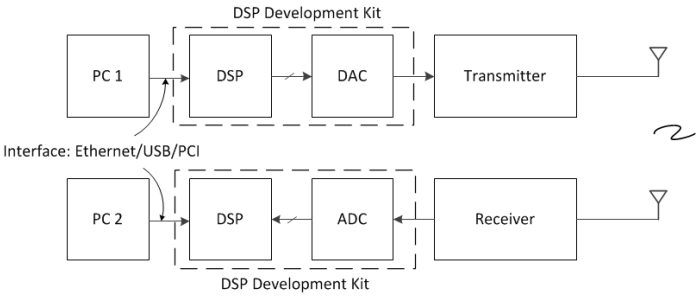

Fig. 2. The DSP based wireless prototyping solution

this solution is also called *software defined radio*. The implementation complexity for this solution is similar to the computer based solution. The data interfaces need to be further developed based on the *board support packages* provided by DSP manufactures.

A disadvantage of this solution is that the interface between DSPs and ADCs and DACs requires additional work since most high-speed ADCs and DACs do not contain on-chip buffers. Taking the ADC as an example, the output data at the I/O pins last only one clock cycle (for example, 100ns at a sampling rate of 10MSPS). A DSP running an operating systems or enabling interrupts may miss of one or more samples of data resulting in seriously degradation in performance. Therefore, an extra buffer is inserted between DSPs and ADCs. A lot of DSP development kits insert a FPGA or a complex programmable logic device (CPLD) to deal with such an issue. Another problem is that the computation is still not powerful enough for most medium or high-speed applications, such as Wi-Fi and ultra-wide band (UWB).

2.3 FPGA Solution

Baseband processing in wireless communications systems normally requires intensive computation to complete up to hundreds of multiplications, additions and others during one sampling cycle. DSP and FPGA are the two most commonly used programmable devices that meet the requirements of computation. Furthermore, the computation capability can be estimated by the number of 16-bit multiplications that can be executed within one second. A high-speed DSP is TMS320C6474 from TI has a computation power of 28,800 MIPS (million instructions per second) at a unit price of $170. At the comparable price, a FPGA chip of EP3C55 from Altera can be bought, which contains 156 multipliers when operating at a clock of 250MHz, or 55k logic elements (LEs) at a clock of 437.5MHz. Every 266 LEs can also be synthesized to form a 16-bit multiplier. The total number of multiplications that this FPGA chip can accommodate with within a second is therefore $156 * 250M + (55000/266) * 437.5M = 129,460M$, which has several times of computation power than DSP. With higher prices,

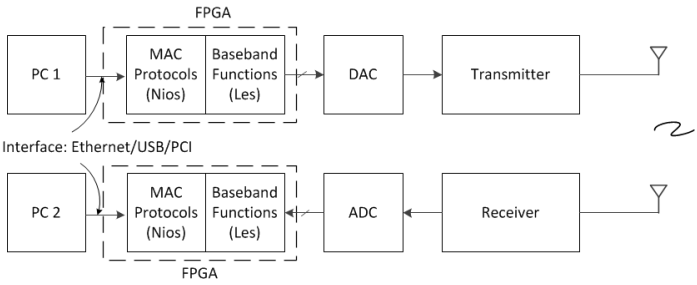

Fig. 3. The FPGA based wireless prototyping solution

FPGA chips with a triple number of LEs operating at a double clock can be purchased, while there seems no faster DSP can be found on the market.

The FPGA based architecture of wireless communications prototype is shown in Fig. 3. An obvious drawback of FPGA based solution lies on that the high requirements on the programmers's skills and training as well as the long development period when compared to that upon DSP. However, in recent years the advance of the development software and tools largely assuages such an issue. Another drawback is that the coding and programming of signal processing algorithms on FPGA using VHDL is more difficult than that over DSP using C language. This is the price that we have to pay for the benefit gained from the FPGA based solution. Meanwhile, it normally takes more synthesis time for FPGA than compilation time when DSP is adopted to implement similar modules.

In addition, the high-speed user data interface, such as USB, PCI and Ethernet, are non-trivia for FPGA when they are implemented through using VHDL codes upon LEs than C codes over DSP. Solutions to such issue will be further discussed in the followings.

2.4 DSP+FPGA Solution

As we mentioned above, DSP is good at data interfacing while FPGA at parallel computing and interfacing to ADC and DAC modules, the combination of DSP and FPGA seems to be a *good* approach, such as the development kits from Sundance [7] and Lyrtech [8]. However, the foremost factor that prevent such a solution from being widely employed lies in the complexity of hardware, across development environment and soaring high price more than $10K. Moreover, both platforms use third party *Diamond* operating system, even worsening the situation.

2.5 FPGA + Embedded Processer Solution

We can conclude that FPGA is the feasible solution and dominate the wireless communications prototyping but require significant experience and sufficient

backgrounds. On the other hand, the implementation of MAC protocol over LEs is tedious and onerous, although doable. To this end, recent FPGA chip fortunately embeds a micro-processor, which is called *system on a programmable chip* (SoPC). Such a solution outperform the solution of DSP plus FPGA in many aspects and therefore prevails. The FPGA board used in this project costs $1,200/unit. More detail on design and implementation will be given in followings.

3 Prototyping on SUN Radio: An Example

A SUN radio prototype upon the IEEE 802.15.4 standard is built up according to the FPGA based solution. The Cyclone III development kits and data conversion HSMC boards containing dual channel ADC and DAC, both from the *Terasic* company are selected. The FPGA kit integrates a EP3C120F780C FPGA chip, 256MB dual-channel DDR2 SDRAM, 8MB SRAM, 64MB flash memory, an Ethernet transceiver chip, and a USB port for debugging. The data conversion board has a dual channel DAC (DAC5672) and two ADC chips (AD9254), both having a 14-bit resolution and an operation clock of 100MSPS. For quadrature modulation, the I and Q channels require dual channel ADC and DAC. The data conversion boards connect to the FPGA kit through the HSMC interface, which is defined by the *Altera* company.

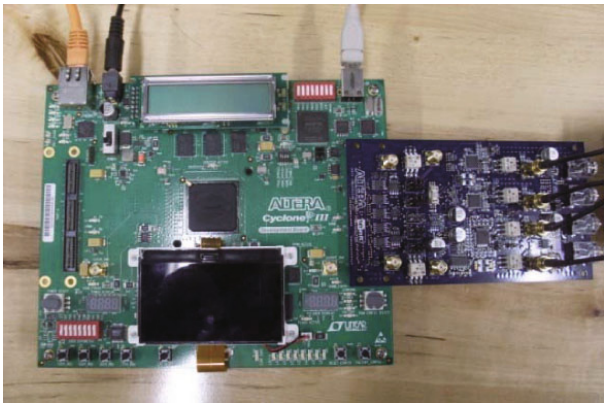

Fig. 4. The snapshot of the prototyping based on the Cyclone III FPGA FPGA development kit

Fig. 3 and Fig. 4 show the diagram and snapshot of the prototype. The programs consist of the C codes running over the embedded Nois II processor within the FPGA chip and VHDL codes on LEs.

1) **C programs upon Nios II processor:** The C codes realize the MAC protocol as well as Ethernet interface. Two tasks named as *Package Receiving*

and *MAC Processing* running on μCOS operating system are established to realize the above two functions. Inter-task communications and synchronization are realized by a message queue. The data flow chart of the program at the transmitter is shown in Fig. 5 and Fig. 6. The codes at the receiver is similar but with reverse processing. At transmitter, the task "Package Receiving" keep polling the Ethernet interface using Socket API. Once a user datagram protocol (UDP) package is received, its address will be copied to the message queue. The task "MAC processing" watches on the message queue in a non-blocking manor. After picking up an arrival packet from the queue, it starts the process of the MAC protocol. The flowchart of the processing at transmitter is demonstrated in Fig. 5. The two tasks are synchronized by the message queue.

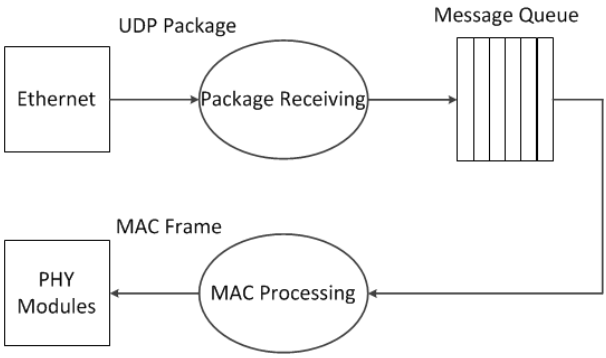

Fig. 5. Data interface and MAC protocol running over the Nios II processor at transmitter

2) **VHDL programs on LEs:** The baseband processing is realized upon LEs using VHDL codes. Fig. 7 and Fig. 8 show the diagrams of the baseband processing at transmitter and receiver, respectively. In addition to the FIFOs and drivers for ADCs and DACs, other function blocks are well defined by the IEEE802.15.4g standard (draf). Nios II processor works in an asynchronous mode while LEs in a synchronous mode. The FIFO needs to be carefully designed to seamlessly pass data between physical layer and MAC layer. Fig 9 shows the interface between them and Fig. 10 the ping-pong mode of the FIFO where a dual buffer is adopted. In the meantime, Quartus provides IP cores for typical function modules including FFT, IFFT and Viterbi decoding. Fig. 11 shows the realization of the time synchronization, consisting of 18 processing in parallel. The flexible trading off between complexity and speed validates our comments on FPGA based solution to wireless system prototyping.

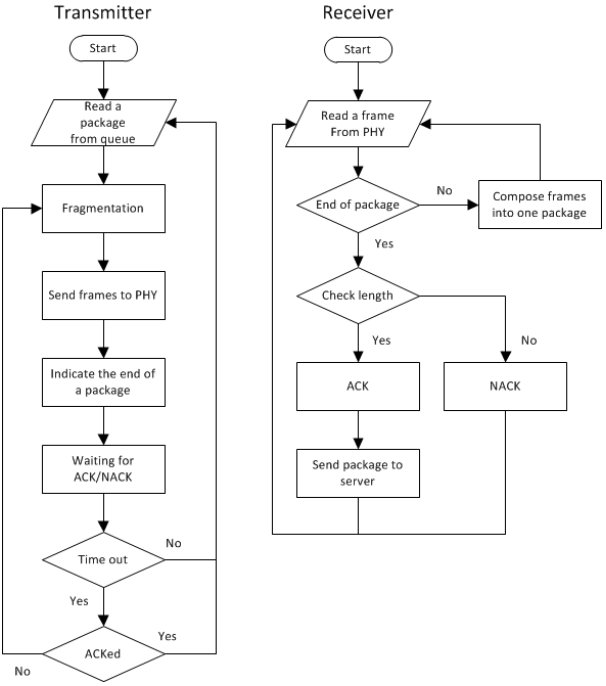

Fig. 6. The flowcharts of the MAC protocol at transmitter

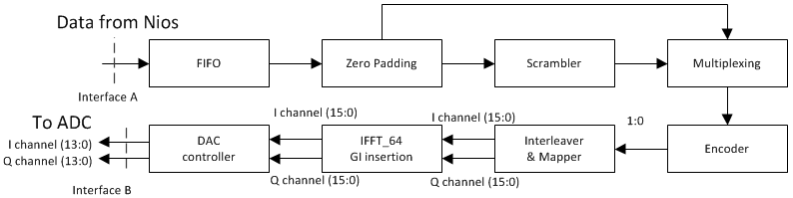

Fig. 7. The baseband processing at transmitter

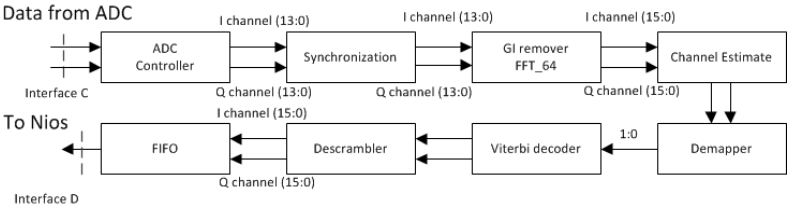

Fig. 8. The baseband processing at receiver

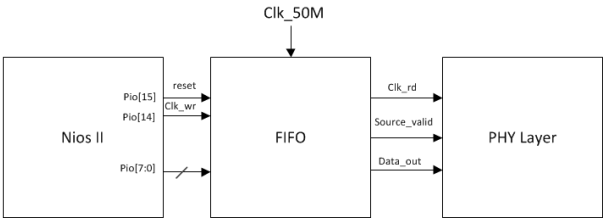

Fig. 9. The interface between PHY and MAC modules

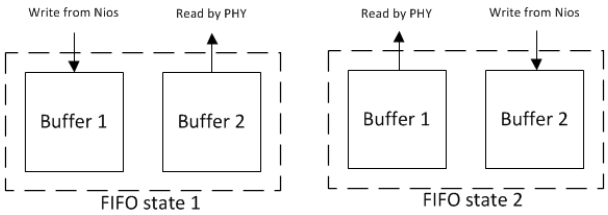

Fig. 10. The ping-pong mode of the FIFO

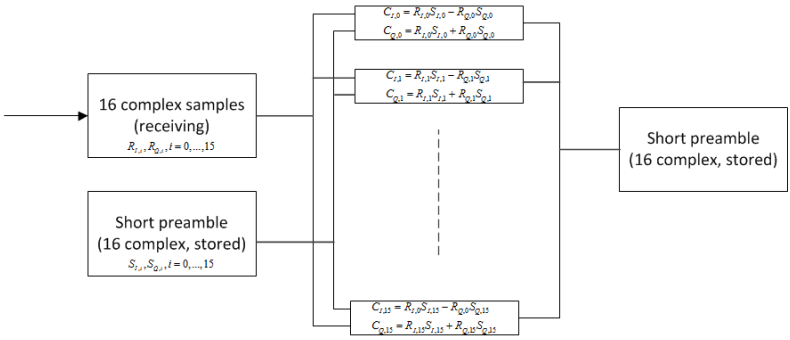

Fig. 11. The realization of time synchronization

4 Results of the SUN Radio Prototype

The performance of the SUN radio is tested in the terms of data rate, processing latency, bit error rate (BER) versus signal-to-noise ratio (SNR). A trial on the application is conducted by transferring a MP3 file from the transmitter to the receiver. Through the time used and the size of the file, the estimate throughout is 110 kbits/s.

The processing latency between the interface A and interface B in Fig. 7, and the latency between the interface C and interface D in Fig. 8, represent the transmission delay and receiving delay. The summation of two delays is

system delay including data buffering for serial-to-parallel conversion as well as the computation delays of the function blocks. System delay is critical for SUN which require real-time message delivery for smart grid applications. The delay generated by MAC functions depends on the factors of SNR, interferences, retransmission time and others, which is counted into the systems delay. Other delays caused by physical circuits including ADC, DAC and amplifiers are much small and thus are ignored.

The measurements of delays and BERs were conducted within a loopback where the interface B and interface C are connected and the delay between the interface A and interface D is measured. Test data from the Nios processor are input to interface A and finally recovered at interface D and thereafter read back to the processor. A C program running on Nios realize data generation, data collection and measurements.

In order to minimize the variations of the timing of operating system, the transmission delay is measured by equipments rather than using Nios itself. A Tektronix digital oscilloscope (DPO7354) is used to monitor the waveforms at interface A and interface D, simultaneously, and a snapshot of captured waveforms is shown in Fig. 12, where data at interface A and interface D are colored in yellow and blue, respectively. The delay is estimated as 1.89ms, meeting the requirement of real-time delivery for smart grid applications.

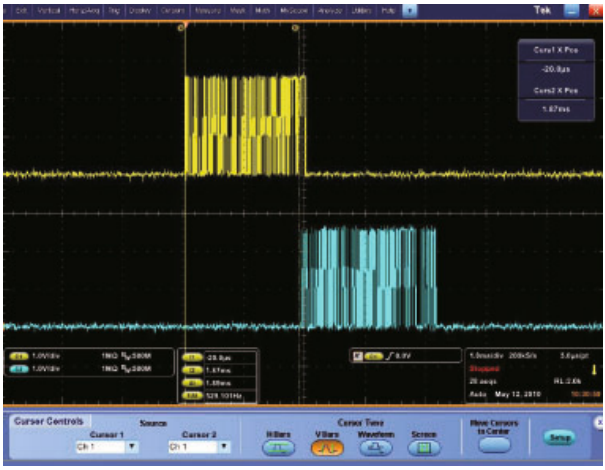

Fig. 12. The captured waveforms at interface A and interface D

For BER testing, a set of digitalized additive white Gaussian noises (AWGNs), of which the powers are adjusted according to the required SNR, are added at the interface C by LEs. BERs under each SNR were calculated by Nios processor through comparing the bit streams at interface A and interface D. Fig. 13 shows the BER-SNR curve. Finally, the delay of the Ethernet interface is tested by

using the *ping* command at the server. Result shows that the round delay of the Ethernet interfaces at the transmitter and receiver including SUN radios is less than 1ms.

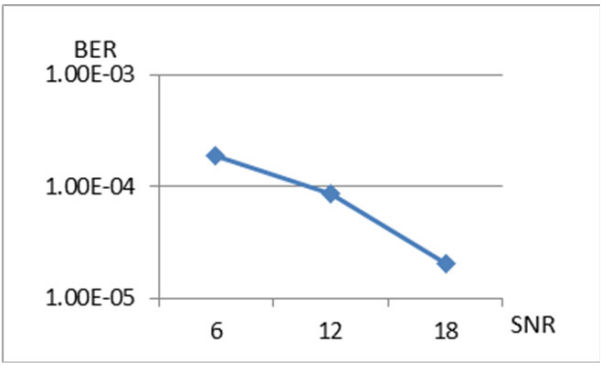

Fig. 13. The measured BER-SNR curve of the SUN radio

5 Conclusions

A SUN radio prototype for smart grid applications upon the new IEEE 802.15.4g standard (draft) is implemented based on the FPGA based solution. Experiments verifies the designed specifications including delay and BER. The prototype features in a single chip solution integrating both baseband modules and MAC protocols that operate upon the embedded Nios processor. RF front ends are adding on and outdoor field testing will be conducted to further evaluate the performance of SUN radios adopting narrow OFDM for large scale low duty WSN applications, such as smart grid, vehicular networks, eHealth and the internet of things.

References

1. Electric Power Research Institute: Report to NIST on the Smart Grid Interoperability Standards Roadmap, http://www.nist.gov/smartgrid/
2. "ZigBee" IEEE Std 802.15.4
3. Preliminary draft P802.15.4g-d0.2
4. Signatec Company, http://www.signatec.com/products/daq/high-speed-fpga-pcix-digitizer-board-pda16.html
5. Arbitrary waveforms generator,http://www.strategic-test.com/eng/Instruments/Arbitrary-Waveform-Generators2
6. GNU Radio, http://gnuradio.org/redmine/wiki/gnuradio
7. Sundance, http://www.sundance.com/
8. Lyrtech, http://www.lyrtech.com/