

# Software-Defined Radio and Broadcasting

Guest Editors: Daniel Iancu, John Glossner, Peter Farkas,  
Mihai Sima, and Michael McGuire





---

# **Software-Defined Radio and Broadcasting**

International Journal of Digital Multimedia Broadcasting

---

## **Software-Defined Radio and Broadcasting**

Guest Editors: Daniel Iancu, John Glossner, Peter Farkas,  
Mihai Sima, and Michael McGuire



---

Copyright © 2009 Hindawi Publishing Corporation. All rights reserved.

This is a special issue published in volume 2009 of “International Journal of Digital Multimedia Broadcasting.” All articles are open access articles distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.



## Editor-in-Chief

Fa-Long Luo, Element CXI, USA

## Associate Editors

Sos S. Agaian, USA

Jörn Altmann, South Korea

Ivan Bajic, Canada

Abdesselam Bouzerdoun, Australia

Hsiao Hwa Chen, Taiwan

Gerard Faria, France

Borko Furht, USA

Rajamani Ganesh, India

Jukka Henriksson, Finland

Shuji Hirakawa, Japan

Y. Hu, USA

Jiwu Huang, China

Jenq-Neng Hwang, USA

Daniel Iancu, USA

Thomas Kaiser, Germany

Dimitra Kaklamani, Greece

Markus Kampmann, Germany

Alexander Korotkov, Russia

Harald Kosch, Germany

Massimiliano Laddomada, USA

Ivan Lee, Canada

Jaime Lloret-Mauri, Spain

Thomas Magedanz, Germany

Guergana S. Mollova, Austria

Alberto Morello, Italy

Algirdas Pakstas, UK

Kiran Ranga Rao, USA

M. Roccetti, Italy

Peijun Shan, USA

Ravi S. Sharma, Singapore

Tomohiko Taniguchi, Japan

Wanggen Wan, China

Fujio Yamada, Brazil

# Contents

**Software-Defined Radio and Broadcasting**, Daniel Iancu, John Glossner, Mihai Sima, Peter Farkas, and Michael McGuire

Volume 2009, Article ID 698402, 2 pages

**3G Long Term Evolution Baseband Processing with Application-Specific Processors**, Perttu Salmela, Juho Antikainen, Teemu Pitkänen, Olli Silvén, and Jarmo Takala

Volume 2009, Article ID 503130, 13 pages

**The Sandblaster Software-Defined Radio Platform for Mobile 4G Wireless Communications**,

V. Surducun, M. Moudgill, G. Nacer, E. Surducun, P. Balzola, J. Glossner, S. Stanley, Meng Yu, and D. Iancu

Volume 2009, Article ID 384507, 9 pages

**Software-Defined Radio Demonstrators: An Example and Future Trends**, Ronan Farrell,

Magdalena Sanchez, and Gerry Corley

Volume 2009, Article ID 547650, 12 pages

**Exploiting Redundancy in an OFDM SDR Receiver**, Tomas Palenik and Peter Farkas

Volume 2009, Article ID 194148, 7 pages

**Implementing a DVB-T/H Receiver on a Software-Defined Radio Platform**, Yong Jiang, Wen Xu, and Cyprian Grassmann

Volume 2009, Article ID 937848, 7 pages

**Galois Field Instructions in the Sandblaster 2.0 Architecture**, Mayan Moudgill, Andrei Iancu, and Daniel Iancu

Volume 2009, Article ID 129698, 5 pages

**Low-Cost Transceiver Architectures for 60 GHz Ultra Wideband WLANs**, S. O. Tatu, E. Moldovan, and S. Affes

Volume 2009, Article ID 382695, 6 pages

**Multiband Antennas for SDR Applications**, E. Surducun, V. Surducun, D. Iancu, and J. Glossner

Volume 2009, Article ID 460143, 9 pages

**A Geometry-Inclusive Analysis for Single-Relay Systems**, Meng Yu, Jing (Tiffany) Li, and Hamid Sadjadpour

Volume 2009, Article ID 146578, 9 pages

## Editorial

# Software-Defined Radio and Broadcasting

**Daniel Iancu,<sup>1,2</sup> John Glossner,<sup>1</sup> Mihai Sima,<sup>3</sup> Peter Farkas,<sup>4</sup> and Michael McGuire<sup>3</sup>**

<sup>1</sup> Sandbridge Technologies, Inc., Tarrytown, NY 10591, USA

<sup>2</sup> Tampere University of Technology, Korkeakoulunkatu 1, FIN-33720 Tampere, Finland

<sup>3</sup> Department of Electrical and Computer Engineering, University of Victoria, Victoria, BC, Canada V8W 3P6

<sup>4</sup> Department of Telecommunication FEI STU, University of Slovakia, 812 19 Bratislava, Slovakia

Correspondence should be addressed to Daniel Iancu, diancu@sandbridgetech.com

Received 12 October 2009; Accepted 12 October 2009

Copyright © 2009 Daniel Iancu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Software-Defined Radio is not a myth anymore. As process technologies transitioned to 65 nm and lower, power consumption, size, and cost are no longer impediments to SDR adaption. Many SDR products have been introduced in the consumer market in the past few years, in both wireless infrastructure and user equipment, ranging from 2G to 4G. As process technology evolves, processors become more computationally capable pushing the borderline between software and hardware closer to the antenna. One of the main advantages of SDR is software reconfigurability that leads to significant design simplification for multimode receivers. Therefore, processor reconfigurability must be accompanied by multimode antennas as well as reconfigurable radios.

This special issue highlights the most recent advances in the topics of Software-Defined Radio from both industry and research. After two-round peer-reviews, nine papers are selected to be included in this special issue.

The first paper entitled “3G Long Term Evolution Baseband Processing with Application-Specific Processors,” by Salmela et al., addresses the challenges encountered by next-generation receivers. The most computationally intensive functions, list sphere decoding, QR decomposition, fast Fourier transform, and turbo decoding are implemented in application-specific processors.

The second paper entitled “The Sandblaster Software Defined Radio Platform for Mobile 4G Wireless Communications,” by Surducan et al., presents an inexpensive PCI-e 4G SDR platform built around Sandblaster technology. With the multimode antenna system and flexible RF receiver, the platform is capable of executing in real-time multiple 3G and 4G communication protocols.

In the third paper entitled “Software Defined Radio Demonstrators: An Example and Future Trends,” by Farrell

et al., flexible Software-Defined Radio platform developed to investigate the use of Software-Defined Radio in the provision of infrastructure elements in telecommunication applications.

The fourth paper entitled “Exploiting redundancy in an OFDM SDR receiver,” by Tomas Palenik and Peter Farkas, proposes an easy-to-implement modification to an existing SDR OFDM receiver resulting in improved error correction capabilities while preserving full compatibility with the existing standards.

The fifth paper entitled “Implementing a DVB-T/H Receiver on A Software Defined Radio Platform,” by Jiang et al., presents the feasibility of a software implementation of Digital Video Broadcasting on the Software-Defined Radio platform, MuSIC from Infineon Technologies

The sixth paper entitled “Galois Field Instructions in the SANDBLASTER 2.0 Architecture,” by Moudgill et al., describes the Galois Field instruction set implemented in the Sandblaster 2.0 Architecture from Sandbridge Technologies. The instruction set takes the advantage of the fact that for some applications, it is not necessary to execute both a polymultiply and polyremainder for each GF multiplication resulting in significant computational expense saving.

The seventh paper entitled “Low-Cost Transceiver Architectures for 60 GHz Ultra Wideband WLANs,” by Tatu et al., proposes a multiport architecture that allows the complete integration of circuits including antennas, in planar technology on the same substrate improving cost and receiver performance.

The eighth paper entitled “Multi-band Antennae for SDR Applications,” by Surducan et al., deals with the challenges and implementation of multiband antennas for multimode communication systems. It is shown that starting from a

composite-folded dipole structure, a more complex antenna configuration capable of supporting multiple communication protocols is derived.

The last paper entitled “A Geometry-Inclusive Analysis for Single-Relay Systems,” by Yu et al., investigates the impact of a relay’s location on the system capacity and outage probability for amplify-forward and decode-forward schemes. It is shown that a candidate pool of 3 to 5 nodes is enough to obtain most of the cooperative gain provided by a selective single-relay system.

## **Acknowledgments**

The authors are grateful to the reviewers for their timely and insightful comments on the submitted manuscripts. Without their invaluable work, this special issue would have not been possible.

*Daniel Iancu  
John Glossner  
Mihai Sima  
Peter Farkas  
Michael McGuire*

## Research Article

# 3G Long Term Evolution Baseband Processing with Application-Specific Processors

**Perttu Salmela,<sup>1</sup> Juho Antikainen,<sup>2</sup> Teemu Pitkänen,<sup>1</sup> Olli Silvén,<sup>3</sup> and Jarmo Takala<sup>1</sup>**

<sup>1</sup> Department of Computer Systems, Tampere University of Technology, P.O. Box 553, 33101 Tampere, Finland

<sup>2</sup> Centre for Wireless Communications, University of Oulu, P.O. Box 4500, 90014 Oulu, Finland

<sup>3</sup> Information Processing Laboratory, Department of Electrical and Information Engineering, University of Oulu, P.O. Box 4500, 90014 Oulu, Finland

Correspondence should be addressed to Perttu Salmela, perttu.salmela@tut.fi

Received 13 November 2008; Accepted 6 January 2009

Recommended by Daniel Iancu

Data rates in the upcoming 3G long term evolution (LTE) standard will be manifold when compared to the current universal mobile telecommunications system. Implementing receivers conforming with the high-capacity transmission techniques is challenging due to the complexity and computational requirements of algorithms. In this study, the software defined radio (SDR) is targeted and the four essential baseband functions of the 3G LTE receiver, namely, list sphere decoding, fast Fourier transform, QR decomposition, and turbo decoding, are addressed and the functions are implemented as application specific processors (ASPs). As a result, the design space that describes the essential computational challenges of 3G LTE receivers is clarified and estimates of area, power, and interprocessor communication buffer requirements are presented.

Copyright © 2009 Perttu Salmela et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

## 1. Introduction

The upcoming 3G long term evolution (LTE) standard will support data rates up to 100 Mbps [1]. Such a high data rate will be achieved in 20 MHz bandwidth by using transmission techniques like orthogonal frequency division multiplexing (OFDM) [2], multiple-input multiple-output (MIMO) [3], that is, the use of multiple antennas, and an efficient forward error correction method, the turbo coding [4]. As these techniques are applied, the receiver needs to realize very sophisticated algorithms. The design complexity or difficulty of designing implementations of such algorithms calls for flexible software-based solutions, that is, software defined radio (SDR). On the other hand, the computational complexity of algorithms advocates dedicated hardware accelerators for maximizing the performance. Thus, the implementation technique of choice should possess the benefits of both approaches.

High throughput and efficiency can be achieved with highly parallel hardware accelerator which is designed for the application in hand. As a drawback, designing is time consuming and any further changes can be difficult with

unprogrammable fixed hardware. Programmable processor-based implementations tend to suffer from a lower throughput, unused resources, and memory throughput bottlenecks but they allow a shorter development time and higher flexibility due to the programmability. A solution, which strives to achieve the benefits of both hardware accelerators and processor-based implementations, is to use application-specific processors (ASPs) with highly parallel computing resources. With proper tools, ASPs can be designed and programmed rapidly, yet high throughput can be obtained with highly parallel computing resources. Flexibility and efficiency are obtained with accurate control at software level.

On the contrary to focusing on the implementation of solely one function, even a couple of interoperating functions complicate the design. For example, the number of clock domains and the most suitable clock frequencies must be determined for all the functions. In addition, there is always a tradeoff between area and throughput. Furthermore, even if the throughput is adequate, the delay can be too long. Thus, the dimensions of the design space include clock frequency, area, power, parallelism,

number of processors, clock domains, and so forth. To find answers to the multivariable and multiobjective design problems, the design space must be explored by focusing on promising candidates, that is, design alternatives, and analyzing them. Naturally, such analysis is far away from evaluation of a fully functional system-on-chip (SoC) but it provides inevitable insight into the design problem in hand.

In this paper, efficient ASPs, whose performance rivals pure hardware implementations, are applied to the 3G LTE baseband processing. The targeted essential and computationally demanding baseband functions are list sphere decoding (LSD), fast Fourier transform (FFT), QR decomposition, and turbo decoding. Baseband functions are separated from system level operations as the area and power analysis focuses on the core computations. The assisting interprocessor communication (IPC) is analyzed in terms of data buffer requirements of ideal IPC links. The presented work forecasts how demanding the implementation of these baseband functions of the 3G LTE receiver would be, and what would be the number of logic gate equivalents (GE), power, number of processors, and IPC requirements with realistic clock frequencies. The results also show how strongly an efficient symbol detection method dominates the total complexity.

The next section introduces some previous implementation techniques and fundamentals of the addressed functions and system. In Section 3, a high-level description of the targeted receiver is presented. The applied ASP implementations are presented in Section 4. Multiprocessing requirements and complexity are analyzed in Section 5 before the conclusions.

## 2. Previous Work

The upcoming 3G LTE, MIMO-OFDM, and the main transmission parameters are discussed in depth in [1]. In [5], the fundamentals of MIMO communications, including the capacity gain, channel model, and receiver algorithms are explained. As an example of the high potential of MIMO-OFDM systems with sophisticated symbol detection, a  $4 \times 4$  MIMO-OFDM system and maximum likelihood (ML) detection achieves over 1 Gbps throughput in [6]. The MIMO-OFDM is applied also in 4G telecommunications systems and WLANs. The entire baseband processing chain of a 4G SDR is addressed in [7]. A hardware implementation of MIMO-OFDM system for WLANs is presented in [8] and implementations of two vital functions, the matrix decomposition and symbol detection with sphere decoder, are considered in [9].

Typical DSPs like TI's C64x [10] are tempting candidates for baseband processing as they have parallel computing resources and special instructions suitable for many of the required tasks. For example, the FFT can be computed with an off-the-shelf library routine [11]. Alternatively, a dedicated FFT processor can be used [12], and with FPGAs, off-the-shelf IP cores can be used for the FFT [13]. In this

paper, we have applied the FFT implementation presented in [14] for complexity and power estimations.

There are many alternative techniques and algorithms for QR decomposition. Since the MIMO receiver requires a relatively small matrix, extensively parallel systolic array processors [15, 16] can be oversized solutions. The QR decomposition requires the computation of a highly nonlinear operation, namely division by a norm or, alternatively, multiplication with an inverse of square root operation. One approach is to carry out the computations in  $\log_2$  domain as in [17]. A Nios processor with CORDIC accelerators on FPGA is used in [18]. In [19], a scalable architecture using squared Givens rotations is presented. In this paper, the QR decomposition implemented in [20] has been applied.

In many practical MIMO systems, the ML symbol detection can be too complex. Alternatively, for example, zero forcing or linear minimum mean square error (LMMSE) principles can be applied [21]. In this paper, LSD is assumed as it approximates the ML detection with reduced computational complexity. There are several LSD variants. A  $K$ -best LSD is assumed in this study and in [22–24] where architectures for the algorithm are presented. The  $K$ -best LSD processor used in this paper is presented in detail in [25].

Turbo decoder can be implemented, for example, as a coprocessor of a DSP as in [26] or a hardware accelerator [27] or an ASP [28]. Naturally, there are variants of the algorithm, and the level of parallelism and clock frequency mainly determine the throughput. In this paper, a programmable turbo decoder presented in [29] is applied.

The ASP template, which is applied in this paper, uses the transport triggered architecture (TTA) [30]. There exists many multiprocessor systems applying TTA processors. In [31], a simple asynchronous communication link between TTA processors is enabled with units containing an FIFO buffer. TTA and LEON3 processors are connected with an AMBA bus in [32]. On the contrary to a shared bus, a network-on-chip approach has been applied in [33] where two Coffee RISC processors, a TTA processor, and a shared memory are connected with a network. A bioinspired multiprocessor system is presented in [34, 35] where TTA processors are abstracted as cells of a biological system. In this paper, the IPC requirements of a multiprocessor system are analyzed, and an abstract multiprocessor system using shared memory banks as communication links is assumed.

Several inevitable building blocks for baseband processing are presented in the aforementioned references. On the contrary to focusing solely on one particular function without practical motivation for the achieved throughput, we focus on a baseband processing chain consisting of FFT, QR decomposition, LSD, and turbo decoder and we derive the processing requirements from the 100 Mbps peak data rate of the upcoming 3G LTE systems. In this paper, we consider especially the ASPs in [14, 20, 25, 29] and their applicability for baseband processing. In order to obtain realistic estimates, the considered ASPs are resynthesized for the prevailing operating conditions, complexity, and power estimates are given for a 3G LTE compliant system configuration.



### 3. System Model

A high-level description of the targeted 2-antenna MIMO-OFDM receiver is presented in Figure 1. The input ports are connected to radio-frequency functions of the receiver. The functional block diagram is only a high-level model as it does not suggest how the functions should be mapped to the processors nor it does not suggest how data is passed between the functions and whether the data vectors have serial or parallel presentations. In the following, the targeted transmission techniques are presented briefly.

**3.1. Orthogonal Frequency Division Multiplexing.** OFDM uses the frequency spectrum efficiently as the used frequency band is divided into several orthogonal subcarriers. OFDM uses the FFT and inverse FFT (IFFT) for efficient conversions between the time and frequency domains. The time domain signal is generated in the transmitter side with inverse transform

$$X_T = \text{IFFT}(X_F), \quad (1)$$

that is, data belonging to several parallel subcarriers is fed to the IFFT. In the receiver side, parallel subcarriers,  $X_F$ , are extracted from the time domain signal  $X_T$  with

$$X_F = \text{FFT}(X_T). \quad (2)$$

To alleviate timing synchronization, additional cyclic prefix is inserted to the signal. The channel estimation can be alleviated with pilot symbols.

In the receiver side, distortion of the channel can be equalized conveniently in frequency domain by multiplying the received symbols with equalizing factors. Before the FFT, the cyclic prefix must be removed from the signal, and timing synchronization is responsible for feeding the time domain signal, whose length equals the FFT length, with correct timing offset to the FFT block.

**3.2. Multiple-Input Multiple-Output.** In a spatial multiplexing MIMO system, multiple antennas are used to transmit independent data streams. Spatial multiplexing gain, that is, increase in capacity, is proportional to the number of antennas and it does not require extra power nor bandwidth. Two transmit and receive antennas are a highly probable configuration for the first 3G LTE systems, since a higher number of antennas increases the computational requirements of symbol detection significantly.

Computational complexity of ML detection of transmitted symbols depends exponentially on the number of spatial channels. Therefore, even with a modest number of antennas, simpler approximated methods must be used. The usage of list sphere decoding algorithms is tempting as they can achieve higher performance than LMMSE [36], even though they are computationally demanding. The sphere detector restricts the search space by evaluating only the symbols inside the sphere centered in the received symbol. In the system model in Figure 1,  $K$ -best LSD is assumed. The  $K$ -best LSD operates by gradually increasing the dimension

of the symbol vector. At each level, a list of the  $K$  best partial solutions is selected for continued processing.

In principle, an MIMO system with a complex-valued channel matrix,  $\mathbf{H}$ , noise vector,  $\mathbf{n}$ , transmitted symbol,  $\mathbf{s}$ , and received symbol,  $\mathbf{y}$ , can be described with

$$\mathbf{y} = \mathbf{H}\mathbf{s} + \mathbf{n}. \quad (3)$$

The number of receive and transmit antennas equals the numbers of rows and columns of  $\mathbf{H}$ , respectively. The transmitted symbol  $\mathbf{s}'$  can be estimated by ML detection by solving

$$\mathbf{s}' = \arg \min_{\mathbf{s}} \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2, \quad (4)$$

which gives the optimal result. However, solving (4) is intractable with multiple antennas and large constellations.

Instead of solving (4), the symbol estimation can be simplified by using QR decomposition of  $\mathbf{H}$ . With this practice, the computational complexity is lowered. Instead of ML detection, a substitute

$$\mathbf{s}' = \arg \min_{\mathbf{s}} \|\mathbf{y}' - \mathbf{R}\mathbf{s}\|^2 \quad \text{where } \mathbf{y}' = \mathbf{Q}^H \mathbf{y} \quad (5)$$

is used. As the  $\mathbf{R}$  is in upper triangular form, approximation of  $\mathbf{s}'$  is computationally simpler with the aid of (5). The simplified approximation is based on computing the Euclidean distance in (5) by gradually increasing the dimensions of the symbol vector. Basically, there will be partial solutions which are too far away from the received symbols and when such partial solutions are discarded, the search space is efficiently limited. The  $K$ -best LSD applies the aforementioned principles by maintaining a  $K$ -length list of the best partial solutions found so far.

**3.3. Forward Error Correction.** The function of the forward error correction is to introduce redundancy in the transmitted signal in order to alleviate error detection and correction. In 3G LTE, a similar turbo coding as in the contemporary 3G systems will be used. The only difference is the definition of the interleaving function [37, 38]. The new interleaving function covers longer code blocks and it is simpler to implement than the contemporary 3G interleaving. Naturally, the longer code block size affects the memory requirements.

Turbo decoding is an iterative process, which runs a soft-in soft-out (SISO) component decoder several times. The arguments of the component decoder are extrinsic information  $\lambda^{\text{in}}$ , systematic bit,  $\mathbf{y}^s$ , and parity bit vector,  $\mathbf{y}^p$ . As a result, it generates new extrinsic information,  $\lambda^{\text{out}}$ , and soft bit estimate vectors,  $\mathbf{L}$ , that is,

$$(\lambda^{\text{out}}, \mathbf{L}) = \mathbf{f}_{\text{SISO}}(\lambda^{\text{in}}, \mathbf{y}^s, \mathbf{y}^p). \quad (6)$$

The *a posteriori* information is generated on the previous half iteration, and used as *a priori* information on the next half iteration. The information exchange takes place by passing the extrinsic information between the component decoder processes. The main difference between the half



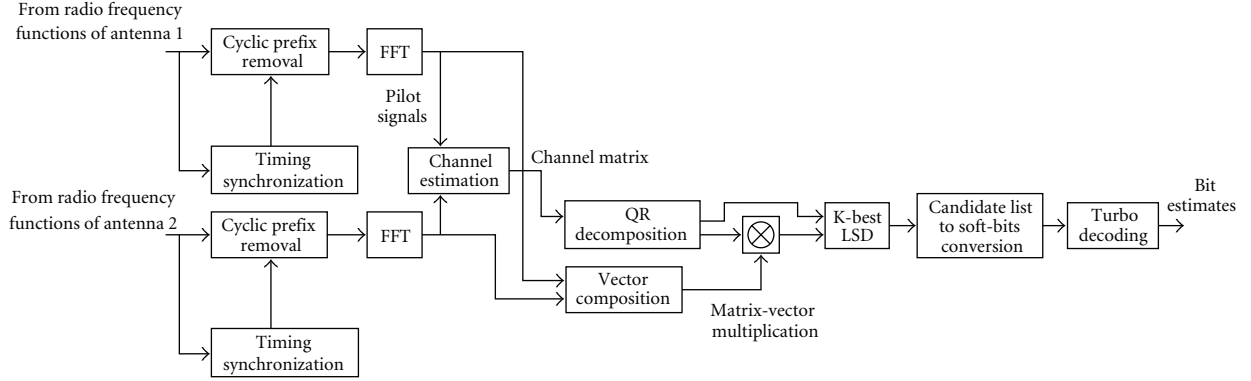


FIGURE 1: A simplified block diagram of baseband processing of a two-antenna MIMO-OFDM receiver using  $K$ -best LSD for symbol detection. ASP implementations for FFT, QR decomposition, list sphere detection, turbo decoding, and multiplications are considered in this paper.

iterations is that every second half iteration processes data related to the interleaved systematic bits.

The applied turbo decoder processor in Section 4.6 uses the max-log-MAP algorithm for SISO decoding. In principle, max-log-MAP algorithm generates the forward path metric at state  $u$  at trellis stage  $k$ ,  $\alpha_k(u)$  recursively as

$$\alpha_k(u) = \max_{u'} (\alpha_{k-1}(u') + d_k(u', u)), \quad (7)$$

where  $d_k(u', u)$  is the branch metrics. The backward path metric is defined in the same way as

$$\beta_{k-1}(u') = \max_u (\beta_k(u) + d_k(u', u)). \quad (8)$$

The soft output,  $L_k$ , is a function of the forward, backward, and branch metrics, that is,

$$L_k = \max_{u', u: x^s=0} (\alpha_{k-1}(u') + \beta_k(u) + d_k(u', u)) - \max_{u', u: x^s=1} (\alpha_{k-1}(u') + \beta_k(u) + d_k(u', u)). \quad (9)$$

In (9), the first maximum corresponds to the state transitions where the transmitted systematic bit  $x^s = 0$ , and the second maximum is computed based on all the state transitions where  $x^s = 1$ . The sigum function is used to calculate the final hard bit estimates based on  $L_k$ . The new extrinsic information  $\lambda_k^{\text{out}}$  is computed with the aid of the received soft systematic bit,  $y_k^s$ , *a priori* information,  $\lambda_k^{\text{in}}$ , and  $L_k$ , that is,

$$\lambda_k^{\text{out}} = \frac{1}{2} L_k - y_k^s - \lambda_k^{\text{in}}. \quad (10)$$

#### 4. Transport Triggered Architecture Processor Implementations

The targeted baseband functions are implemented on a customizable ASP template. The implementations are presented shortly in the following sections.

*4.1. Principles of Transport Triggered Architecture Processors.* In this paper, TTA [30] has been used as the architecture template for ASPs. Processors with similar efficiency and

performance could be implemented also with some other ASP templates supporting sufficient parallelism and customizability. Since there exists up-to-date tool support for TTA processors [39], we have exploited the template and the baseband functions have been implemented with TTA processors.

The main difference when compared to a pure hardware solutions is that the TTA processors are fully programmable. TTA reminds VLIW machine but the interconnection is exposed to the programmer unlike in traditional processors. TTA is one form of application-specific instruction set processor where the instruction set of the processor is tailored for the given application. In this sense, code for customized TTA processor is not compatible with another TTA processor. In TTA, the computations are triggered by data transported to the computing unit, which is contrary behavior to conventional operation-triggered architectures. The processor is programmed with data transports, which reflects the architecture to the programmer. The maximum number of parallel data transports is determined by the number of buses of the interconnection network. As the interconnection network connecting the computing resources is visible to the programmer, there is accurate control of all the operations.

The modularity of TTA processors allows to tailor them by including only the necessary function units (FU). Application-specific functions are implemented as user defined special FUs (SFU) which are utilized in a similar way as conventional FUs, that is, by transporting data on assembly level or by using function-like macros in C language. Due to frequent direct data transports between the FUs or SFUs, the register pressure is very low. However, the modularity of the processor allows a variable number of register files (RF) with variable numbers of input and output ports. In Figure 2, a high-level example of a TTA processor is given. The figure highlights the modular and customizable structure of the processor by denoting the variable numbers of the respective resources. The control unit (CU) in Figure 2 allows data transports to access the program counter and the return address register, which is required for jump or call operations.

The load on the buses of the interconnection network can be lowered by excluding the unnecessary connections if the work load of the processor is known beforehand. In this case, the targeted application program determines which connections are used. Typically, one application requires only a fraction of all the possible connections between the computing resources. If any other application is run on the same processor, it must be able to use the same connections. As a consequence of the limited connectivity and lowered load on the buses, the maximum clock frequency of the interconnection network is raised.

**4.2. Multiprocessor Systems with TTA Processors.** There exists many multiprocessor systems applying TTA processors as listed in Section 2. However, the required number of processors for baseband processing in Section 5 is far higher than the number of processors in [31–33]. In addition to the bioinspired abstraction of multiple TTA processors [34, 35], multiple processors could be also abstracted as a hierarchical structure where the SFUs would be comprised of TTA subprocessors. Another way would be to combine all the TTA processors to a set of loosely connected clusters inside a single TTA processor. However, assembly programming such a processor would be error prone due to the extremely long instruction word and the scheme would limit the control flow of the clusters very strictly to a single combined flow. Regardless of the applied structure of the multiprocessor system, generating and controlling a multiprocessor system consisting of dozens of processors would be a demanding task.

Since it would be uneconomical to produce results of computations faster than they can be transferred to the next stage, shared memory banks or RFs running with the same clock frequency,  $f_i$ , as the processors must be assumed for the IPC at the lowest level. Fortunately, the applied TTA processor template has flexible memory interfaces, which can simplify the IPC. For example, simple point-to-point connections between two processors could be implemented with an SFU interfacing a shared single- or dual-port memory. Furthermore, if complex address generation or bank selection is required, it can be included to the same SFU, which slightly raises the abstraction level of the IPC visible to the programmer. Such an incorporation of all the memory related logic to the same unit could enable a seamless IPC.

**4.3. FFT Processor.** The applied FFT TTA processor is presented in detail in [14]. The processor implements mixed-radix FFT consisting of radix-2 and radix-4 computations and it supports several power-of-two transform sizes. It has 11 RFs containing 25 general-purpose registers and three Boolean registers, 17 buses in the interconnect network, a conventional adder, a comparison unit, and two-load/store units. The main computations are carried out with the following SFUs.

**Complex Adder Unit.** It supports four different summations composed of four alternative operands.

**Complex Multiplier.** It alleviates the butterfly operation with four real multipliers and two real adders.

**Address Generator Unit.** It generates two addresses with bitwise reversal and rotation operations.

**Coefficient Generator.** It generates the twiddle factors instead of loading them from a memory.

The processor applies a complex-valued number presentation where the real and imaginary parts both take 16 bits. Data is stored in single-port memory banks and the kernel loop applies the principles of software pipelining. Code compression is applied to enhance the code density and lower the power consumption.

**4.4. QR Decomposition Processor.** The QR TTA processor presented in [20] is based on the modified Gram-Schmidt algorithm [40]. With complex-valued arithmetic units the processor can compute equally well both the complex- and real-valued decompositions. The only conventional units of the processor are the two-load/store units and an RF consisting of five general purpose registers. The interconnection network contains seven buses. The applied SFUs are as follows.

**Complex Adder/Subtractor Unit.** It is for native complex-valued computations.

**Complex Multiplier Unit.** It can optionally conjugate the other input. The conjugation is required for the computation of the real-valued norm.

$1/\sqrt{x}$  unit is for a fast estimation of the highly nonlinear function. The function is used in the QR decomposition to avoid division operations.

As the processor has a bit accurate complex multiplier, it can be used also for other tasks where the accuracy of 16-bit fixed-point number system is sufficient. The  $1/\sqrt{x}$  unit and the multiplier can be used also for computation of square root as  $x(1/\sqrt{x}) = \sqrt{x}$ .

**4.5. K-Best LSD Processor.** The LSD TTA processor in [25] generates a 16-element list of candidate solutions to approximate the transmitted symbol  $\mathbf{s}'$  in (5). The processor uses 16-bit arithmetic and it is targeted for  $2 \times 2$  antennas and 64-quadrature amplitude modulation (QAM). Instead of  $2 \times 2$  complex-valued matrix, a real-valued matrix with doubled dimensions is processed. Therefore, a real-valued  $4 \times 4$  QR decomposition is required for the LSD. The interconnection network is very sparse and contains 16 buses. The arithmetic operations are computed with two addition units, a subtraction unit, a multiplier, and a squaring unit. The following SFUs are targeted for the applied K-best algorithm.

**Insertion Sorter Unit.** It sorts a list of 16 samples according to the partial Euclidean distances (PED). Internally, the list

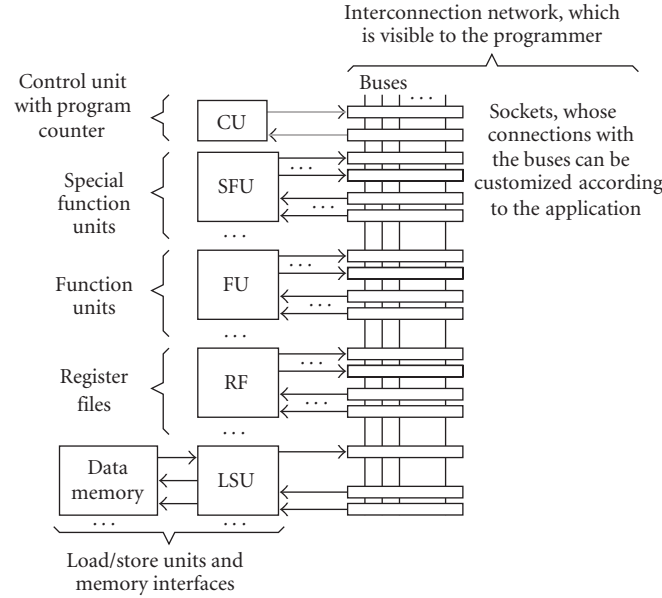


FIGURE 2: TTA processors consist of a CU and variable number of FUs, SFUs, RFs, and LSUs. Unused connections between the resources can be excluded from the interconnection network.

is kept in a shift register and the new value is inserted to the register pointed by comparison logic.

*PED Extractor Unit.* It extracts the PED from the internal storage format, that is, the unit accesses bits by hardwiring.

*Multiplexer and Look-Up-Table Unit.* It consists of a multiplexer selecting the bits, which index the look-up-table. In principle, the unit converts a bit pattern to fixed-point format.

*Storage Format Composer Unit.* It composes a 28-bit word consisting of symbol information and the corresponding PED.

There are three RFs of sizes 16, 10, and 4 registers. On the contrary to conventional processors, the LSD TTA processor does not have load/store units nor data memory, since there is no need for accessing large arrays. The input data is passed via two RFs and the results of the computations are available in the registers of the insertion sorter SFU.

**4.6. Turbo Decoder Processor.** The turbo decoder TTA processor is presented in [29]. It has a sparsely connected interconnect network of 30 buses and the high number of buses is a consequence of high parallelism. The only conventional FUs are the addition and comparison units. There are only two RFs, both of them containing one general purpose register. As there are not many conventional FUs, the applied max-log-MAP algorithm is computed solely with the following SFUs.

*Control Unit.* It generates a control word which is used as an argument to all the other SFUs.

*Address Generator.* It generates addresses for accessing the branch metric buffer.

*Forward Process Unit.* It computes forward path metrics according to (7).

*Backward Process Unit.* It computes backward path metrics as defined in (8) and extrinsic information and soft output bit estimates according to (10) and (9), respectively.

*Branch Metric Generator.* It generates and buffers the branch metrics for the forward and backward processes.

The turbo decoder TTA processor applies high parallelism as it processes one trellis stage in 1.016 clock cycles on average, that is, both forward and backward path metrics are computed in one clock cycle. Such a high parallelism requires also a high memory throughput. Therefore, the processor does not have conventional load/store units. Instead, the SFUs access memory interfaces of the processor directly. As (7)–(10) indicate, the main computations in the SFUs are carried out with basic arithmetic, add-compare-select, and maximum operations. The processor includes memory bank selection, address generation, and access buffer logic to allow parallel interleaved accesses of the extrinsic information with four-single-port memory banks. The interleaving function is excluded from the processor and it is accessed via external interface of the processor.

## 5. Processing Requirements and Complexity

The number of processors, their total area and memory requirements, and interprocessor communication requirements are derived from the targeted 100 Mbps throughput.

**5.1. Time and Throughput Requirements.** There are seven OFDM symbols per transmit antenna in 0.5 millisecond time frame in 3G LTE downlink. Thus, the processing time requirement  $T_{\text{FFT}} = 0.5 \text{ millisecond}/7 = 71 \text{ microseconds}$  includes also the additional time contributed by the cyclic prefix of the OFDM symbol. The FFT must be computed for both antennas.

The QR decomposition must be processed in the coherence time,  $T_{\text{coh}}$ , of the channel. If bullet train speed  $v_r = 500 \text{ km/h}$  is assumed for the receiver, the coherence time is  $T_{\text{coh}} = c/(Fv_r) = 0.9 \text{ millisecond}$  where  $c$  is the speed of light and  $F = 2.4 \text{ GHz}$  is the carrier frequency. However, with a more rapidly varying channel, the QR decomposition must be computed more frequently, that is, shorter  $T_{\text{coh}}$  must be used in (12). A single QR decomposition combines information from all the antennas. In other words, the matrix and vector sizes of the QR decomposition depend on the number of antennas.

The LSD must be computed for each subcarrier. So, the time requirement equals to the time requirement of the FFT. However, even if the maximum length of the FFT is 2048, only 1201 subcarriers are in use. A single LSD processes the signals of both antennas, that is, it outputs estimates of symbols transmitted from both antennas.

Since the turbo decoder processes soft bits instead of QAM symbols, it is meaningful to express throughput as data rate. The throughput requirement of turbo decoding equals the maximum data rate of 100 Mbps. Naturally, with code rate  $R = 1/2$  and 64-QAM symbols, the data rate on the LSD side is 200 Mbps and symbol rate 33.3 Msps.

**5.2. Required Number of Clock Cycles.** The FFT TTA processor in [14] takes 12332 clock cycles for the 2048-point transform and the transform must be computed for both antennas. So, the required clock cycles of the FFT task are

$$C_{\text{FFT}} = 2 \times 12332 = 24664. \quad (11)$$

The QR decomposition algorithm is of order  $O(n^3)$  and the QR decomposition TTA processor in [20] takes 139 clock cycles for a  $4 \times 4$  matrix. The dimensions of the decomposed matrix are doubled, since the LSD TTA processor applies real-valued computation. Since the  $\mathbf{Q}$  matrix is the argument of matrix-vector product in (5), the products are mapped to the same processor. The products must be computed continuously for each received symbol vector, but the QR decomposition only once in the coherence time. So, the average number of clock cycles in  $T_{\text{FFT}}$  time period, for both computations is approximately

$$C_{\text{QR,avg}} = 1201 \times \left( 139 \times \left( \frac{T_{\text{FFT}}}{T_{\text{coh}}} \right) + 16 \right) = 32386, \quad (12)$$

where  $4 \times 4$  matrix multiplication takes 16 clock cycles. Naturally, with more rapidly varying channel, the  $C_{\text{QR,avg}}$  increases as the  $T_{\text{coh}}$  must be decreased. The products take approximately 59% of the  $C_{\text{QR,avg}}$ . The maximum number of clock cycles is spent when the decomposition of a new channel matrix is computed for each subcarrier, that is,

$$C_{\text{QR}} = 1201 \times (139 + 16) = 186155. \quad (13)$$

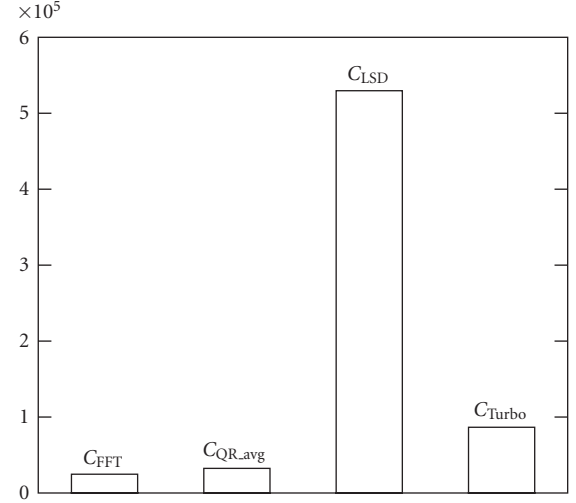


FIGURE 3: Required number of clock cycles of the processing tasks in  $T_{\text{FFT}} = 71 \text{ microseconds}$  time frame.

The average number of clock cycles,  $C_{\text{QR,avg}}$ , is only 17% of the maximum,  $C_{\text{QR}}$ .

The LSD TTA processor in [25] takes 441 clock cycles for processing one symbol vector. Thus, in  $T_{\text{FFT}}$  time period the number of required clock cycles for the LSD,  $C_{\text{LSD}}$ , is approximately

$$C_{\text{LSD}} = 1201 \times 441 = 529641. \quad (14)$$

Fortunately, the LSD can be parallelized among the subcarriers.

In order to compare turbo decoding with the other baseband functions, the clock cycles of turbo decoding must be normalized to clock cycles,  $C_{\text{Turbo}}$ , taken in  $T_{\text{FFT}}$  time frame. The turbo decoder TTA processor in [29] takes 1.016 clock cycles per trellis stage processed in half iteration. With six iterations, each trellis stage is processed 12 times. Therefore,

$$C_{\text{Turbo}} = T_{\text{FFT}} \times 100 \times 10^6 \times 12 \times 1.016 = 86563, \quad (15)$$

where the first multiplications  $T_{\text{FFT}} \times 100 \times 10^6$  express how many bits are processed in  $T_{\text{FFT}}$ . Turbo decoding can be parallelized to several processors with block-by-block pipelining where each processor decodes a code block of its own independently.

The required number of clock cycles of all the four functions are illustrated in Figure 3. The figure shows clearly how the LSD dominates the computation load. Obviously, the requirements cannot be met with single-processor systems with currently achievable clock frequencies.

**5.3. Number of Processors.** The required minimum number of processor is determined by the throughput per processor, clock frequency,  $f_i$ , and parallelization scheme of the targeted functions. If a task  $i$  can be parallelized to several processors and the throughput is directly proportional to the number of processors, then the minimum required number of



processors,  $P_i$ , of the task  $i$  taking  $C_i$  clock cycles in time frame  $T_{\text{FFT}}$  is

$$P_i = \left\lceil \frac{(C_i/T_{\text{FFT}})}{f_i} \right\rceil. \quad (16)$$

The utilization,  $U_i$ , of the processor,  $P_i$ , dedicated to task  $i$  tells how efficiently the computing resources are used. It can be defined in a similar way as

$$U_i = \frac{C_i}{(P_i T_{\text{FFT}} f_i)}. \quad (17)$$

Naturally,  $100 \times (1 - U_i)$  tells how many percent of the time the processor  $P_i$  idles. For the QR decomposition and matrix-vector product task, the average number of clock cycles,  $C_{\text{QR\_avg}}$ , is used to calculate the minimum number of processors and utilization. The total utilization of the whole processing chain can be computed as

$$U = \sum_{i \in S_{\text{tasks}}} \frac{C_i}{(T_{\text{FFT}} \sum_{i \in S_{\text{tasks}}} P_i f_i)}, \quad (18)$$

where the sums are computed for all the elements of the task set  $S_{\text{tasks}} = \{\text{FFT}, \text{QR\_avg}, \text{LSD}, \text{Turbo}\}$ . The total utilization in (18) expresses the ratio between the required execution cycles of all the tasks and the available execution cycles of all the processors.

**5.4. Delay.** The delay of a task depends on the maximum size of the processed data vector and the scheduling. Except for the first half iteration, the turbo decoder requires that the whole code block is received before decoding. The maximum code block length is 6144 [37], which is about 20% longer than in the current 3G systems. With code rate  $R = 1/2$ , the required number of soft bits is naturally  $2 \times 6144 = 12288$ . For two OFDM symbols, the LSD generates symbol candidate lists, which can be converted to  $2 \times 1201 \times 6 = 14412$  soft bit estimates with 64-QAM (6 bits per symbol). Since the number of soft bits exceeds the required number for the maximum code block length, the analysis of the delay of FFT and LSD can be limited to the processing of two OFDM symbols.

With at maximum two processors, the delay of the FFT is simply

$$D_{\text{FFT}} = \frac{C_{\text{FFT}}}{(P_{\text{FFT}} f_{\text{FFT}})}, \quad P_{\text{FFT}} \in \{1, 2\}, \quad (19)$$

and in a similar way the delay of the LSD is

$$D_{\text{LSD}} = \frac{C_{\text{LSD}}}{(P_{\text{LSD}} f_{\text{LSD}})}, \quad (20)$$

where  $P_{\text{LSD}} \in \{1, 2, \dots, 1201\}$  as the LSD can be parallelized among the subcarriers. The QR decomposition processor has two tasks, the QR decomposition and the matrix-vector products, of which the QR decomposition is computed only once in the coherence time,  $t_{\text{coh}} = 0.9$  millisecond. Thus, the worst-case delay when both tasks are computed is

$$D_{\text{QR}} = \frac{C_{\text{QR}}}{(P_{\text{QR}} f_{\text{QR}})}, \quad (21)$$

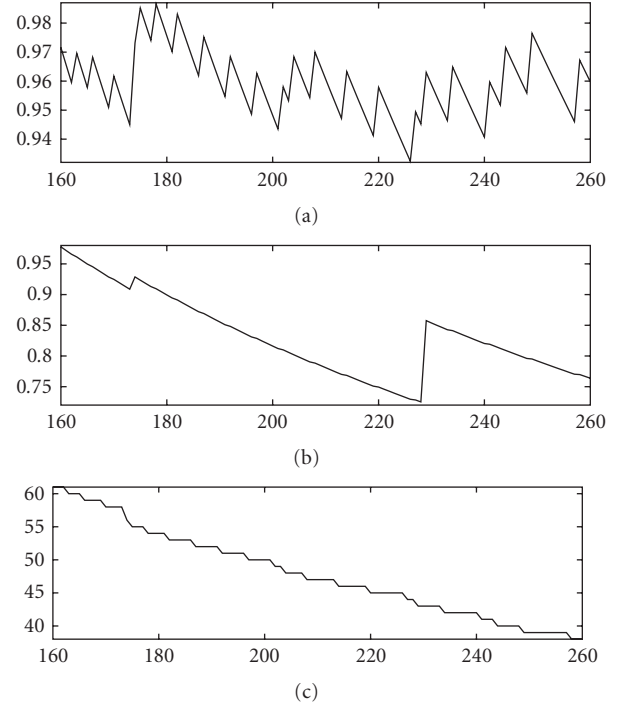


FIGURE 4: Configurations as a function of  $f_i$  with single clock domain: (a) total utilization, (b) total delay in millisecond, (c) the number of processors. The x-axis denotes  $f_i$  in MHz.

where  $P_{\text{QR}} \in \{1, 2, \dots, 1201\}$  as the decompositions and multiplications can be parallelized among the subcarriers. For an average delay,  $C_{\text{QR\_avg}}$  can be used in a similar way. The delay of turbo decoding is determined by the maximum code block size, 6144. Thus, the delay with six turbo iterations is

$$D_{\text{Turbo}} = \frac{6144 \times 6 \times 2 \times 1.016}{f_{\text{Turbo}}}, \quad (22)$$

where processing one trellis stage with the turbo decoder TTA processor takes on average 1.016 clock cycles. Distributing the turbo decoding to several processors with block-by-block pipelining would affect only the throughput but not the delay and, therefore, the number of processors is omitted from (22).

**5.5. TTA Processor Configurations as Function of Clock Frequency.** Utilization, delay, and number of processors are analyzed in Figure 4 as functions of clock frequency. The total utilization in Figure 4(a) shows that the utilization is always greater than 0.93 in the explored clock frequency range. High utilization can be obtained easily, since the LSD dominates the computational load and it can be parallelized with very fine granularity. In other words, since the utilization of the LSD task is always high, also the utilization of the whole processing chain is relatively high. The peaks in the utilization occur, when the number of processors of some task can be decremented. In that case, the utilization grows. On the contrary, if the number of processors remains untouched and the clock frequency is

TABLE 1: The baseband processing chain with TTA processors,  $2 \times 2$  antennas, 1201 subcarriers, 64-QAM, 6144-length turbo code block, list length  $K = 16$ , data rate 100 Mbps.

Clk. freq. $f_i$ (MHz)	Task $i$	No. of procs. $P_i$	Util. $U_i$	Delay $D_i$ (ms)	Area (kGE)	Area $\times P_i$	Power est. (mW)	Power est. $\times P_i \times U_i$	Tech. ( $\mu\text{m}$ )	Ref.
250	FFT	2	0.69	0.049	30.5	61.0	36.3	50.1	0.13	[14]
250	Turbo dec.	5	0.98	0.300	35.1	175.5	50.8	248.9	0.13	[29]
250	QR & prod.	2	0.91	0.372	17.7	35.4	13.1	23.8	0.13	[20]
250	LSD	30	0.99	0.071	23.6	708.0	20.8	617.8	0.13	[25]
Total		39	0.97	0.792		979.9		940.6		

TABLE 2: An example baseband processing chain with  $2 \times 2$  antennas, 1201 subcarriers, 16-QAM, 4804-length turbo code block, data rate 68 Mbps.

Clk. freq. $f_i$ (MHz)	Task $i$	No. of units $P_i$	Util. $U_i$	Delay $D_i$ (ms)	Area (kGE)	Area $\times P_i$	Power est. (mW)	Power est. $\times P_i \times U_i$	Tech. ( $\mu\text{m}$ )	Ref.
600 & 300	FFT & Turbo	5	0.36	0.396	—	—	718	1303	0.13	[26]
223	QR	1	0.29	0.259	198	198	—	—	0.13	[9]
213	Sphere Decoder	1	0.92	0.065	61	61	—	—	0.13	[9]
Total		7	0.38	0.720		—		—		

increased the utilization decreases. The discontinuations of delay in Figure 4(b) originate from the same phenomenon. The greatest discontinuation at 229 MHz takes place as the QR decomposition is mapped from three to two processors. The number of processors in Figure 4(c) decreases quite steadily, since it is dominated by the LSD task, which requires the largest number of processors.

**5.6. Analysis.** An example configuration of TTA processor-based baseband processing chain is presented in Table 1. A single clock domain with  $f_i = 250$  MHz is applied and the processors have been synthesized with  $0.13 \mu\text{m}$  technology for obtaining complexity and power estimates. The area and power estimates exclude the memories. The power estimates are scaled with the number of respective processors and their utilization in the ninth column of Table 1. The results in Table 1 show that since the LSD task takes only 441 clock cycles per subcarrier and it can be computed for each subcarrier independently, the task can be easily divided among several processors to achieve a high utilization. On the contrary, it is more difficult to obtain very high utilization for both the FFT and the QR processors with the same clock frequency, as the granularity of the tasks is more coarse. As a second remark, the delay of the QR decomposition is long when compared to other functions, even though the other functions are more complex. However, the QR decomposition must be computed only once in the coherence time  $t_{\text{coh}} = 0.9$  millisecond, that is, the delay in Table 1 is the worst case delay. On average, the delay of the QR decomposition and the matrix-vector products is only 17% of the delay in Table 1.

In principle, the FFT and QR tasks could be mapped to the same processor. The processor should be formed as a

hybrid of both processors in this case. Since both functions require complex arithmetic, the same resources could be shared efficiently. With  $f_i = 402$  MHz, both tasks could be mapped to two hybrid FFT/QR TTA processors and a utilization,  $U_{\text{FFT/QR}} = 1.00$ , would be obtained.

Mapping the turbo decoding and some other function to the same processor could not benefit as much from sharing the resources, since the turbo decoding requires mostly real-valued add-compare-select operations. Shortening the delay of the turbo decoding is difficult for two reasons. Firstly, turbo decoding is an iterative process where the previous iteration must be finished before the next one can begin. Secondly, the component decoder applying the radix-2 algorithm processes at maximum one trellis stage in one clock cycle. The next path metrics cannot be computed according to (7) and (8) before the previous ones are computed. For these reasons, increasing the clock frequency or applying the radix-4 algorithm are the only ways to shorten the delay of the turbo decoding task in Table 1.

To illustrate more deeply the computational requirements of the baseband processing, example configurations consisting of other implementations are shown in Tables 2–4. As the respective implementations in Tables 2–4 are not necessarily targeted to the 3G LTE system or they are not targeted to operate among each other, the Tables 1–4 should be not considered as comparisons of TTA processors and other implementations. Instead, the tables show indicative example configurations of baseband processing chains.

For some implementations, all the required information is not available or it is given with different units. The area is reported if it has been given as the GEs. For some implementations, the performance data is not available for the targeted configuration of 2048-length FFT,  $2 \times 2$

TABLE 3: An example baseband processing chain with  $4 \times 4$  antennas, 601 subcarriers, 16-QAM, 4808-length turbo code block, list length  $K = 10$ , data rate 68 Mbps.

Clk. freq. $f_i$ (MHz)	Task $i$	No. of units $P_i$	Util. $U_i$	Delay $D_i$ (ms)	Area (kGE)	Area $\times P_i$	Power est. (mW)	Power est. $\times P_i \times U_i$	Tech. ( $\mu\text{m}$ )	Ref.
45	FFT	2	0.63	0.045	—	—	480	608.45	0.35	[12]
400	Turbo dec.	5	0.82	0.288	64.1	320.5	—	—	0.065	[28]
80	QR	1	0.54	0.489	—	—	—	—	0.25	[8]
50	LSD	2	0.85	0.060	132	264	—	—	0.13	[23]
Total		10	0.80	0.882	—	—	—	—	—	—

TABLE 4: Requirements of 4G baseband processing chain for 100 Mbps data rate [7].

Assumed clk. freq (MHz)	Task $i$ [7]	MCycles/s [7]	Assumed no. of Procs. $P_i$	Util. $U_i$
360	FFT	360	1	1.00
240	STBC	240	1	1.00
385	LDPC	7700	20	1.00
Total		8300	22	1.00

antennas, 64-QAM, and list length 16. For this reason, alternative MIMO-OFDM configurations with lower data rate, 68 Mbps, have been used. Shorter code blocks are assumed for turbo coding in Tables 2 and 3. With shorter code blocks, the delay of the FFT can be limited to processing one OFDM symbol per each antenna.

In the configuration in Table 2, hardware implementations presented in [9] are used for the matrix decomposition and symbol detection. For the FFT and turbo decoding the TI's C6416 DSP has been applied as it can compute the FFT with an efficient software library routine and it includes a turbo coprocessor which runs with halved clock frequency. Since the core DSP and turbo coprocessor are mapped to the same device, the number of required processors is determined by the more dominating task, that is, turbo decoding. The idling of the DSP core while turbo decoding is taken into account when the utilization in Table 2 is calculated, and therefore, the utilization is low in Table 2 but still several processors are required. The hardware implementations for QR and symbol detection in Table 2 are targeted for MIMO-OFDM systems [9]. However, the sphere detector applies a different algorithm than the  $K$ -best LSD which is used in TTA processor implementations.

In Table 3 a 1024-point FFT is applied. The applied turbo decoder processor supports also Viterbi decoding. The list length of the  $K$ -best LSD is 10 symbols. In principle, a complex-valued  $K$ -best LSD with 64-QAM, 2 antennas, and  $K = 16$  must process  $64 + 16 \times 64 = 1088$  nodes and with 16-QAM, 4 antennas, and  $K = 10$  it must process  $16 + 10 \times 16 + 10 \times 16 + 10 \times 16 = 496$  nodes during the symbol detection. Thus, the processing requirements of different symbol detectors can be characterized by the number of visited nodes during a tree traversal of the algorithm. The applied QR decomposition hardware accelerator is presented

TABLE 5: Area of the core processor without memories and data memory requirements of the processors.

TTA processor	Clk. freq. $f_i$ (MHz)	Area (kGE)	Data memory requirements (kbits)
FFT	250	30.5	65.5 divided into 2 single-port memory banks
QR	250	17.7	1.5 dual-port memory
LSD	250	23.6	0.0 (uses only registers)
Turbo decoder	250	35.1	281.7 divided into 16 single-port memory banks

TABLE 6: Additional buffer memory requirements for seamless IPC.

IPC buffer	Memory (words)	Memory (kbits)
FFT: next input	$2 \times 2048$	131.1
FFT: prev. result	$2 \times 2048$	131.1
QR: $\mathbf{R}, \mathbf{Q}^H \mathbf{y}$	$1201 \times (10 + 4)$	538.0
QR: prev. results	$1201 \times (10 + 4)$	538.0
Turbo: next input	$3 \times 6114 + 12$	128.5

in [8] as a part of MIMO-OFDM transceiver for WLANs. The decomposition takes 65 clock cycles for  $4 \times 4$  matrix.

In Table 3, the workload of 4G baseband processing with 100 Mbps is presented in terms of required execution cycles on an SODA architecture [7]. For each task a realistic clock frequency is assumed and the tasks are divided to separate processors. Furthermore, it is assumed that the LDPC error correction decoding task can be parallelized to several processors. The Table 3 shows that the LDPC task dominates clearly the workload.

In conclusion, the results in Tables 2–4 show that in addition to the data rate, the computational requirements depend heavily on the applied algorithms and on the parameters of the algorithms. Furthermore, efficiency in terms of high utilization requires that the tasks can be mapped among the processors or hardware units in a flexible way.

**5.7. Memory Requirements.** The area estimates in Table 1 exclude the memories and memory requirements are reported separately in the Table 5. In other words, the area in terms of logic GE's expresses the complexity of the



actual computations of baseband processing. The separation eases future comparisons, since the memory requirements depend heavily on the targeted data vector lengths and technology. For example, long code blocks are preferred in turbo decoding, as they enhance the error correction performance. A second reason for separating the memories is that the IPC requires also memories, and therefore, the total area with all the memories of the whole baseband processing chain would depend on the implementation method of the IPC.

The data memory requirements in Table 5 show that due to the small matrix size, the QR decomposition requires a very small memory. The LSD processor has no memory requirements at all, as the data is stored in registers. On the other hand, the turbo decoder and the FFT processors require large memories as they have to process long data vectors. The memory of the FFT is divided into two banks and a memory interface hides the banking structure from the programmer, that is, the memory system imitates dual-port memory.

**5.8. Interprocessor Communication Requirements.** As the analyzed processors lack extra facilities for IPC, only requirements but not costs can be stated. There exists many methods for SoCs but they are beyond the scope of this paper, the complexity of computing the main baseband functions. Therefore, the effects of using some particular method or SoC platform are not considered. In Table 6, the IPC requirements are tabulated for an assumed system using shared memory banks between the processors.

The FFT processor uses an in-place algorithm, that is, the result overwrites the input vector and processing does not require additional memory. However, passing the data to and from the FFT processors requires buffer memories. In practice, there must be an extra input buffer which is written while the data in the main memory is processed in-place. In a similar way, there must be an extra output buffer, from which the previous result can be read at the same time. The first two buffers in Table 6 are dedicated for such an IPC. The roles of the three memory banks, that is, the input buffer, the output buffer, and the processing memory, can be interchanged on every two completed OFDM symbols.

The QR processor generates the triangular  $4 \times 4$  matrix,  $\mathbf{R}$ , with 10 nonzero elements and 4-element vector for each subcarrier. The results are written to one buffer. The other identical buffer holds the previous results which are passed to the LSD processors at the same time. Since there are several QR and LSD processors, the buffer must be divided into several parallel accessible banks. Again, the roles of the buffers can be interchanged on OFDM symbol boundaries.

The turbo decoder processors require an additional input buffer which is filled with the soft bits while the decoders are processing. There is no need for and additional output buffer, since the decoder overwrites the previous output only on the last half iteration. The buffer size of the turbo decoder input in Table 6 allows code rate  $R = 1/3$  with the maximum block size. The input word length of the applied turbo decoder TTA processor is 7 bits [29], but all the other applied TTA processors use 16 bits for the real or imaginary parts.

In general, the complexity of IPC buffers depend on the sizes of memory banks, their throughput or clock frequency, and the number of memory banks as each bank requires interfacing logic. In addition, the IPC increases also the computational load which is not included in Tables 1–3. Therefore, if a fully functional SoCs were designed, full utilization should not be targeted when solely the core computations are analyzed. Instead, with lower utilization, computing capacity would be reserved also for the IPC. Also, the total delay in Tables 1–3 exclude the effect of IPC. As it is assumed that one buffer is written while the other is read in a pipelined fashion, it can be assumed that the IPC has a constant delay.

Since the workloads of the processors depend only on the applied block lengths, static scheduling could be applied, which would ease synchronization of the tasks. Even if the number of processors is very high, in principle, similar IPC requirements would be met also with smaller number of processors if they applied higher parallelism internally or if they applied higher clock frequency. The first option would require parallel IPC links and the second option would require smaller number of IPC links but higher throughput for each link.

## 6. Conclusions

The main baseband functions of a 3G LTE conforming MIMO-OFDM receiver were considered in this paper, and ASP implementations were assumed for each function. The main emphasis was on the complexity of the actual computations, that is, the data path, of the functions implemented with the ASPs. The complexity was derived by estimating the required number of respective processors and the clock frequency to meet real-time requirements. The area and power estimates of the functions processed with the ASPs showed the demands of the baseband processing with the current technology. It was shown that especially the LSD dominates the computational load. However, due to the fine granularity and convenient parallelization of the LSD, it can be distributed among several processors and high utilization can be achieved. Also other processors or hardware accelerators of the addressed functions were analyzed to further illustrate the computational demands and costs. The IPC requirements were estimated by a block by block processing model with processors connected via shared memory banks.

## Acknowledgment

This work has been supported by the Finnish Funding Agency for Technology and Innovation under research funding decision 40163/07.

## References

- [1] R. Bachl, P. Gunreben, S. Das, and S. Tatesh, "The long term evolution towards a new 3GPP\* air interface standard," *Bell Labs Technical Journal*, vol. 11, no. 4, pp. 25–51, 2007.

- [2] R. W. Chang and R. A. Gibby, "A theoretical study of performance of an orthogonal multiplexing data transmission scheme," *IEEE Transactions on Communication Technology*, vol. 6, no. 4, pp. 529–540, 1968.
- [3] G. J. Foschini and M. J. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless Personal Communications*, vol. 6, no. 3, pp. 311–335, 1998.
- [4] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and encoding: turbo-codes. 1," in *Proceedings of IEEE International Conference on Communications (ICC '93)*, vol. 2, pp. 1064–1070, Geneva, Switzerland, May 1993.
- [5] A. J. Paulraj, D. A. Gore, R. U. Nabar, and H. Bölcskei, "An overview of MIMO communications—a key to gigabit wireless," *Proceedings of the IEEE*, vol. 92, no. 2, pp. 198–218, 2004.
- [6] K. Higuchi, H. Kawai, N. Maeda, H. Taoka, and M. Sawahashi, "Experiments on real-time 1-Gb/s packet transmission using MLD-based signal detection in MIMO-OFDM broadband radio access," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 6, pp. 1141–1153, 2006.
- [7] M. Woh, S. Seo, H. Lee, et al., "The next generation challenge for software defined radio," in *Proceedings of the 7th International Workshop on Embedded Computer Systems: Architectures, Modeling, and Simulation (SAMOS '07)*, vol. 4599 of *Lecture Notes in Computer Science*, pp. 343–354, Springer, Samos, Greece, July 2007.
- [8] D. Perels, S. Haene, P. Luethi, et al., "ASIC implementation of a MIMO-OFDM transceiver for 192 Mbps WLANs," in *Proceedings of the 31st European Solid-State Circuits Conference (ESSCIRC '05)*, pp. 215–218, Grenoble, France, September 2005.
- [9] B. Cerato, G. Masera, and E. Viterbo, "Enabling VLSI processing blocks for MIMO-OFDM communications," *VLSI Design*, vol. 2008, Article ID 351962, 10 pages, 2008.
- [10] "TMS320C64x Technical Overview," Texas Instruments, SPRU395B, January 2001.
- [11] "TMS320C64x DSP Library Programmer's reference," Texas Instruments, SPRU565B, October 2003.
- [12] Y.-T. Lin, P.-Y. Tsai, and T.-D. Chiueh, "Low-power variable-length fast Fourier transform processor," *IEEE Proceedings: Computers and Digital Techniques*, vol. 152, no. 4, pp. 499–506, 2005.
- [13] S. Y. Lim and A. Crosland, "Implementing FFT in a FPGA coprocessor," in *Proceedings of the International Embedded Solution Event*, pp. 230–233, Santa Clara, Calif, USA, September 2004.
- [14] T. Pitkänen, R. Mäkinen, J. Heikkinen, T. Partanen, and J. Takala, "Low-power, high-performance TTA processor for 1024-point fast fourier transform," in *Proceedings of the 6th International Workshop on Embedded Computer Systems: Architectures, Modeling, and Simulation (SAMOS '06)*, vol. 4017 of *Lecture Notes in Computer Science*, pp. 227–236, Springer, Samos, Greece, July 2006.
- [15] S. Y. Kung, *VLSI Array Processors*, Prentice-Hall, Upper Saddle River, NJ, USA, 1987.
- [16] A. Maltsev, V. Pestretsov, R. Maslennikov, and A. Khoryaev, "Triangular systolic array with reduced latency for QR-decomposition of complex matrices," in *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS '06)*, pp. 385–388, Kos, Greece, May 2006.
- [17] C. K. Singh, S. H. Prasad, and P. T. Balsara, "VLSI architecture for matrix inversion using modified gram-schmidt based QR decomposition," in *Proceedings of the 20th International Conference on VLSI Design jointly with the 6th International Conference on Embedded Systems (VLSID '07)*, pp. 836–841, Bangalore, India, January 2007.
- [18] Altera Corporation, "Implementation of CORDIC-based QRD-RLS algorithm on Altera Stratix FPGA with embedded Nios soft processor technology," White Paper WP-STXQRD-01, Altera Corporation, San Jose, Calif, USA, March 2004.
- [19] F. Edman and V. Öwall, "A scalable pipelined complex valued matrix inversion architecture," in *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS '05)*, vol. 5, pp. 4489–4492, Kobe, Japan, May 2005.
- [20] P. Salmela, A. Burian, H. Sorokin, and J. Takala, "Complex-valued QR decomposition implementation for MIMO receivers," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '08)*, pp. 1433–1436, Las Vegas, Nev, USA, March–April 2008.
- [21] M. Myllylä, J.-M. Hintikka, J. R. Cavallaro, M. Juntti, M. Limingjoja, and A. Byman, "Complexity analysis of MMSE detector architectures for MIMO OFDM systems," in *Proceedings of the 39th Asilomar Conference on Signals, Systems and Computers*, pp. 75–81, Pacific Grove, Calif, USA, October–November 2005.
- [22] Z. Guo and P. Nilsson, "Algorithm and implementation of the K-best sphere decoding for MIMO detection," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 3, pp. 491–503, 2006.
- [23] M. Wenk, M. Zellweger, A. Burg, N. Felber, and W. Fichtner, "K-best MIMO detection VLSI architectures achieving up to 424 Mbps," in *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS '06)*, pp. 1151–1154, Kos, Greece, May 2006.
- [24] K.-W. Wong, C.-Y. Tsui, R. S.-K. Cheng, and W.-H. Mow, "A VLSI architecture of a K-best lattice decoding algorithm for MIMO channels," in *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS '02)*, vol. 3, pp. 273–276, Phoenix, Ariz, USA, May 2002.
- [25] J. Antikainen, P. Salmela, O. Silvén, M. Juntti, J. Takala, and M. Myllylä, "Fine-grained application-specific instruction set processor design for the K-best list sphere detector algorithm," in *Proceedings of the International Conference on Embedded Computer Systems: Architectures, Modeling, and Simulation (SAMOS '08)*, pp. 108–115, Samos, Greece, July 2008.
- [26] S. Agarwala, T. Anderson, A. Hill, et al., "A 600-MHz VLIW DSP," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 11, pp. 1532–1544, 2002.
- [27] Xilinx, "3GPP Turbo Decoder v3.1," DS318, May 2007.
- [28] T. Vogt and N. Wehn, "A reconfigurable application specific instruction set processor for viterbi and log-MAP decoding," in *Proceedings of IEEE Workshop on Signal Processing Systems Design and Implementation (SIPS '06)*, pp. 142–147, Banff, Canada, October 2006.
- [29] P. Salmela, H. Sorokin, and J. Takala, "A programmable max-log-MAP turbo decoder implementation," *VLSI Design*, vol. 2008, Article ID 319095, 17 pages, 2008.
- [30] H. Corporaal, "Design of transport triggered architectures," in *Proceedings of the 4th IEEE Great Lakes Symposium on VLSI (GLSV '94)*, pp. 130–135, Notre Dame, Ind, USA, March 1994.
- [31] I. Karkowski and H. Corporaal, "A framework for design of heterogeneous multi-processor embedded systems," Tech. Rep. 1-68340-44/1997/12, Delft University of Technology, Delft, The Netherlands, 1997.
- [32] J. Guo, K. Dai, and Z. Wang, "A heterogeneous multi-core processor architecture for high performance computing," in

- Proceedings of the 11th Asia-Pacific Conference on Advances in Computer Systems Architecture (ACSAC '06)*, vol. 4186 of *Lecture Notes in Computer Science*, pp. 359–365, Springer, Shanghai, China, September 2006.
- [33] T. Ahonen and J. Nurmi, “Integration of a NOC-based multimedia processing platform,” in *Proceedings of the International Conference on Field Programmable Logic and Applications (FPL '05)*, pp. 606–611, Tampere, Finland, August 2005.
- [34] G. Tempesti, P.-A. Mudry, and R. Hoffmann, “A move processor for bio-inspired systems,” in *Proceedings of NASA/DoD Conference on Evolvable Hardware (EH '05)*, pp. 262–271, Washington, DC, USA, June–July 2005.
- [35] J. Rossier, Y. Thoma, P.-A. Mudry, and G. Tempesti, “MOVE processors that self-replicate and differentiate,” in *Proceedings of the 2nd International Workshop on Biologically Inspired Approaches to Advanced Information Technology (BioADIT '06)*, vol. 3853 of *Lecture Notes in Computer Science*, pp. 160–175, Springer, Osaka, Japan, January 2006.
- [36] M. Myllylä, P. Silvola, M. Juntti, and J. R. Cavallaro, “Comparison of two novel list sphere detector algorithms for MIMO-OFDM systems,” in *Proceedings of the 17th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '06)*, pp. 1–5, Helsinki, Finland, September 2006.
- [37] 3GPP, “Multiplexing and channel coding (release 8),” Technical Specification TS.36.212 v1.0.0, Group Radio Access Network, 3rd Generation Partnership Project, Cedex, France, 2007.
- [38] 3GPP, “Multiplexing and channel coding (FDD) (release 5),” Technical Specification TS 25.212 v5.3.0, Group Radio Access Network, 3rd Generation Partnership Project, Cedex, France, 2002.
- [39] P. Jääskeläinen, V. Guzman, A. Cilio, T. Pitkänen, and J. Takala, “Codesign toolset for application-specific instruction-set processors,” in *Multimedia on Mobile Devices*, vol. 6507 of *Proceedings of SPIE*, pp. 1–11, San Jose, Calif, USA, January 2007.
- [40] G. H. Golub, *Matrix Computations*, John Hopkins University Press, Baltimore, Md, USA, 1989.

## Research Article

# The Sandblaster Software-Defined Radio Platform for Mobile 4G Wireless Communications

**V. Surducu,<sup>1,2</sup> M. Moudgill,<sup>1</sup> G. Nacer,<sup>1</sup> E. Surducu,<sup>1,2</sup> P. Balzola,<sup>1</sup> J. Glossner,<sup>1</sup> S. Stanley,<sup>1</sup> Meng Yu,<sup>1</sup> and D. Iancu<sup>1,3</sup>**

<sup>1</sup> Sandbridge Technologies, 120 White Plains Road, 4th Floor, Tarrytown, NY 10591, USA

<sup>2</sup> Department of Molecular and Biomolecular Physics, National Institute for Research and Development of Isotopic and Molecular Technologies, 65-103 Donath Street, 400293 Cluj Napoca, Romania

<sup>3</sup> Department of Computer Systems, Tampere University of Technology, Korkeakoulunkatu 3, 33720 Tampere, Finland

Correspondence should be addressed to V. Surducu, vsurducu@gmail.com

Received 9 December 2008; Revised 18 May 2009; Accepted 9 September 2009

Recommended by Mihai Sima

We present a tier 2 Software Defined-Radio platform (SDR), built around the latest Sandbridge Technologies' multithreaded Digital Signal Processor (DSP) SB3500, along with the description of major design steps taken to ensure the best radio link and computational performance. This SDR platform is capable of executing 4G wireless communication standards such as WiMAX Wave 2, WLAN 802.11 g, and LTE. Performance results for WiMAX are presented in the conclusion section.

Copyright © 2009 V. Surducu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

## 1. Introduction

SDR is a collection of hardware and software technologies that enable reconfigurable system architectures for wireless networks and user terminals. The SDR should provide an efficient and comparatively inexpensive solution to the problem of building multimode, multiband, multifunctional wireless devices that can be enhanced using software upgrades [1].

*Tier 2 Software-Defined Radios* provide software control of a variety of modulation techniques, wide-band or narrow-band operation, communication security functions, and waveform requirements of current and evolving standards over a broad frequency range. The frequency bands covered may still be constrained at the front-end requiring a switch in the antenna system.

The platform we present in this paper is a tier 2 SDR MIMO capable expresscard, based on the latest DSP from Sandbridge Technologies, the SB3500. It supports both PCI Express and USB 2.0 connectivity and it can also be used standalone, powered from a wall wart 3.3 V/1.5 A supply. In the stand alone mode, data connectivity is provided through a standard USB 2.0 interface and a small adaptor board.

Due to small form factor (54×110 mm in frame) and low power consumption, the platform can easily be transitioned to mobile applications such as smart phones and PDAs.

There are numerous frameworks for SDR platforms [2–4] targeting both research and development for 4G wireless systems, with cost ranging from a few hundred to tens of thousands of dollars. A large variety of SDR platforms bearing multiple processors and/or expensive FPGAs are currently available. To our knowledge the platform we present in this paper is one of the most cost/performance effective form factor designs currently available. Small form factor designs are quite challenging tasks. They need to meet several contradicting criteria such as low power and very high processing speed, multiple frequency bands spread over a large spectrum with reasonable antenna gain in each band, good signals separation and integrity with densely packed components, and low-cost manufacturing. One of the most challenging limitations in the design process is the thickness of the circuit board. The coupling between signals grows with the circuit board thickness resulting in higher noise and signal interference. At the same time, the cost of the circuit board increases inversely with the thickness. A good compromise between circuit board cost and signal integrity

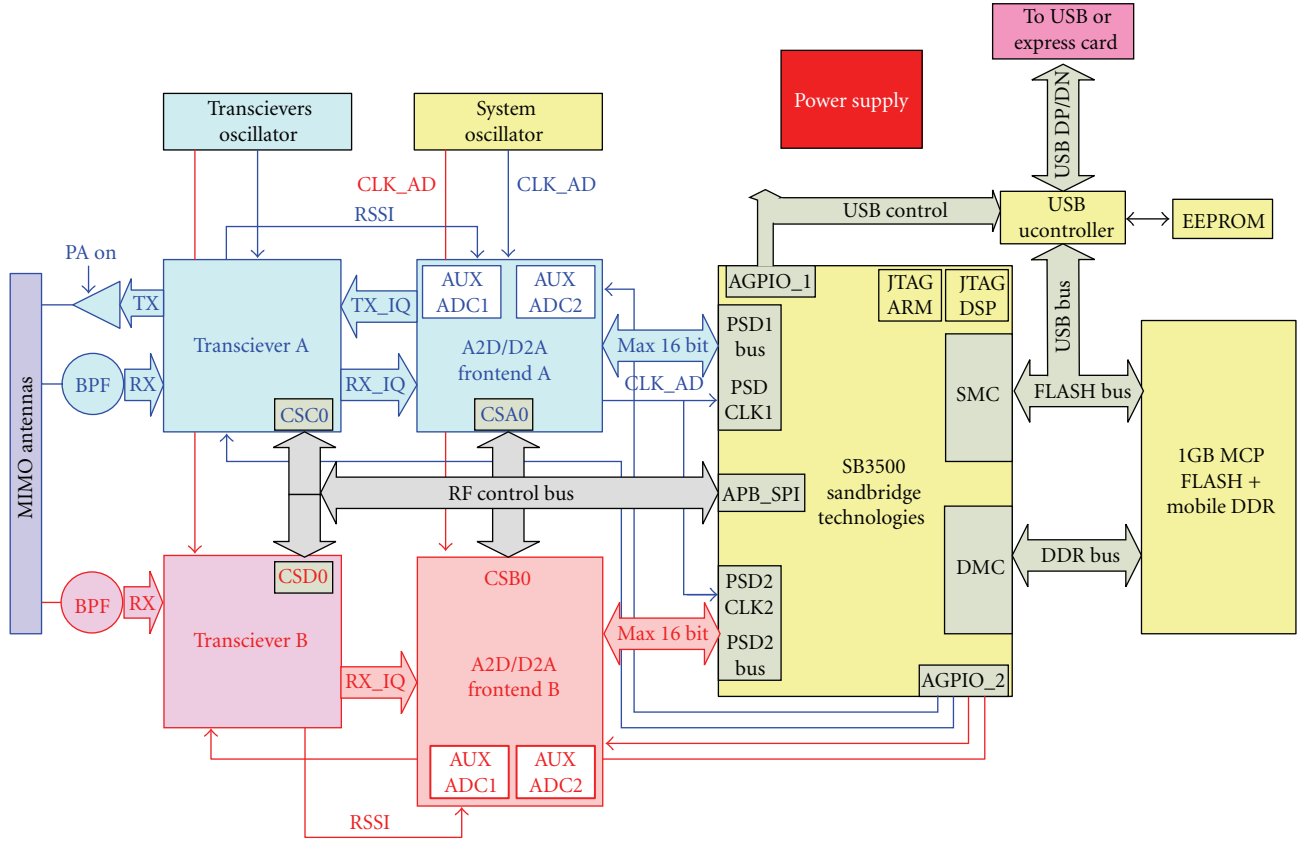


FIGURE 1: SDR platform block diagram.

makes a material difference in the overall performance of the device. High-density Ball Grid Array (BGA) devices will also drive the cost of the circuit board. Finally, at GHz range frequencies, the consistency of the circuit board physical properties need to be tightly controlled, since the RF design often requires better than 10% accuracy for trace impedances.

This paper is structured as follows Section 2 is dedicated to the hardware platform high-level description. Section 3 is dedicated to the SB3500 processor with a brief description of the Instruction Set Architecture (ISA). Section 4 describes the power supply, the Analog Front End (AFE), and the Radio Frequency (RF) front-end. Printed board circuit design issues and noise minimization methods applied are in Section 5. Measurements on the SDR board are presented in Section 6. The conclusions are provided in Section 7.

## 2. The Hardware Platform

The hardware block diagram is illustrated in Figure 1. As shown, the SDR platform includes two symmetrical Zero Intermediate Frequency (ZIF) RF transceivers, capable of Full Frequency Duplexing (FFD) in one receiver and one transmitter configuration or Time Division Duplexing (TDD) mode in two receivers and one transmitter mode ( $2 \times 1$  Multiple Input Multiple Output (MIMO)). Both ZIF channels (A and B in the figure) are digitized by two

high-speed 10 bit data converters connected to the System on Chip (SoC) through parallel busses. The I input to the A2D is sampled every rising edge of the sampling clock CLK\_AD while the Q input is sampled on the falling edge. Both RF transceivers and data converters are controlled through the SPI interface with multiple chip selects (CSA0, CSB0, CSC0, and CSD0) from the SoC. Also, if latency is an issue, the transceivers' gain control can be done through two separated parallel busses (not shown in the figure) connected to one of the General Purpose Input Output (GPIO) ports of the SoC. While transceiver A can be configured as both receiver and transmitter, the transceiver B is fixed to only receive mode. There are three antennas in the system, one transmit antenna fed through a Power Amplifier (PA) and two receive antennas connected to the RF transceivers through Band Pass Filters (BPF). Because of low phase noise requirements (more details about phase noise will come in the following sections) the RF reference oscillator is separated from the system clock.

On the digital side, the SoC is interfaced to the Multiple Chip Package (MCP) through FLASH and DDR memory busses. A separate USB controller, memory mapped in the flash memory area, is used to communicate with the host. Since the FLASH memory is used mostly for the booting sequence, the FLASH bus is less busy.

The whole system is powered using a management scheme compliant with the Point Of Load (POL) architecture, adapted for the size and supply constraints defined



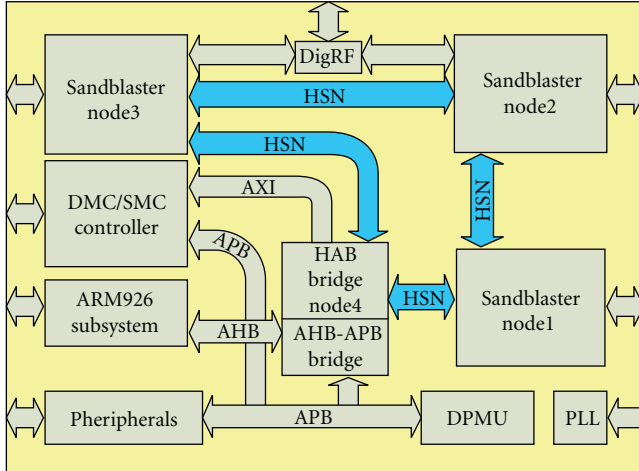


FIGURE 2: The simplified structure of SB3500.

by the SoC and the applications. The custom Power Management-Integrated Circuit (PMIC) used for this design can be programmed by the SoC using a power management (PM) I2C bus, driven directly by the on chip Device Power Management Unit (DPMU) as shown on Figure 4.

### 3. The SB3500 Processor

At the center of the design is the Sandbridge system on chip, SB3500 [5]. The simplified block diagram of SB3500 is illustrated in Figure 2. It consists of three nodes containing the SBX cores with one DigRF interface and an ARM926 subsystem with the facility of Direct Memory Access (DMA), Serial Data Input Output interface (SDIO), Universal Serial Bus interface (USB) and an LCD and camera interfaces, a Dynamic Memory Controller (DMC), a Static Memory Controller (SMC), a DPMU and various peripherals such as timers, General Purpose Input-Output GPIO, audio codec, PS2 interface, SPI interface, smart card interface, I2C interface, and UART/IRDA interface. The SBX sandblaster nodes are connected together in a ring topology on a High Speed Synchronous Network (HSN) 64 bits wide at maximum 300 MHz with programmable bus frequency. The peak bus rates are up to 19.2 Gbps.

The lower-speed busses are connected to the HSN using an Advanced Microcontroller Bus Architecture (AMBA) bridge (HAB bridge) forming the fourth node of the HSN. The HAB bridge drives the memory controller on an Advance eXtensible Interface (AXI). AXI supports separate address/control from data busses, unaligned data transfers, and burst-based transactions. The ARM core communicates with HAB either as a master or as a slave, on the Advanced High performance Bus (AHB), using a bus protocol with a fixed pipeline between address/control and data phases. The ARM can also access the peripherals or program the DPMU using an Advanced Peripheral Bus (APB) with the AMBA-AHB to the AMBA-APB bridge. The APB bus uses a simple protocol for general purpose peripherals. All peripherals are

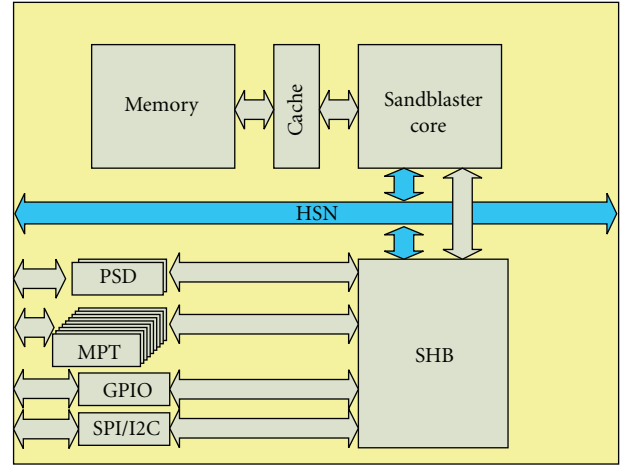


FIGURE 3: The SBX node block diagram.

available for either the SandBlaster Cores or the ARM 926EJ-S, via their base addresses.

A programmable PLL generates the clock, referenced from an external Temperature Compensated eXternal Oscillator (TCXO) source (10 MHz to 50 MHz). The clock generation block from the DPMU distributes several programmable internal clocks to various subsystems. The DPMU directly controls the power domains for the three Sandblaster cores and for the ARM926EJ-S processor. It is also capable of controlling via an external power management IC (using an I2C, SPI, or GPIO) all other power domains for DigRF, DMC, SoC, and IO interfaces separately. Debugging and programming the SoC is possible using separate JTAG interfaces for both the SBX and ARM subsystems.

A detailed structure of the SBX node is present in Figure 3. The Sandblaster core is a multithreaded processor with four independent threads. It contains a 32 KB instruction cache and a 256 KB internal data memory accessible to all hardware threads as well as from external sources via the HSN. Every core has two Parallel Streaming Data (PSD) interfaces for baseband I and Q,  $9 \times 24$  bit wide Multipurpose Timers (MPTs), an SPI interface which can address up to four devices, and an I2C interface. On some nodes, SPI input-outputs are shared with PSD or I2C. The PSDs are used to control input/output data flow between the SB3500 device and a fast external device (e.g., an A2D/D2A front end converter); the data direction is set either externally via PSD\_DIR pins or programmed internally in software.

The Instruction Set Architecture for this processor is simple and orthogonal, with Single Instruction Multiple Data (SIMD) for the processing unit. Each cycle, a thread can execute three instructions. The SB3500 Sandblaster core architecture 2.0 was developed to allow the software implementation of the physical layer of the 4G standard. The major change into the 2.0 architecture is the introduction of 16-wide vector operations and instructions specialized for efficient execution of 4G kernel. Those operations are implementing FFTs with 4 complex multiplies per cycle, polynomial multiply, multiply-reduce, and multiply and

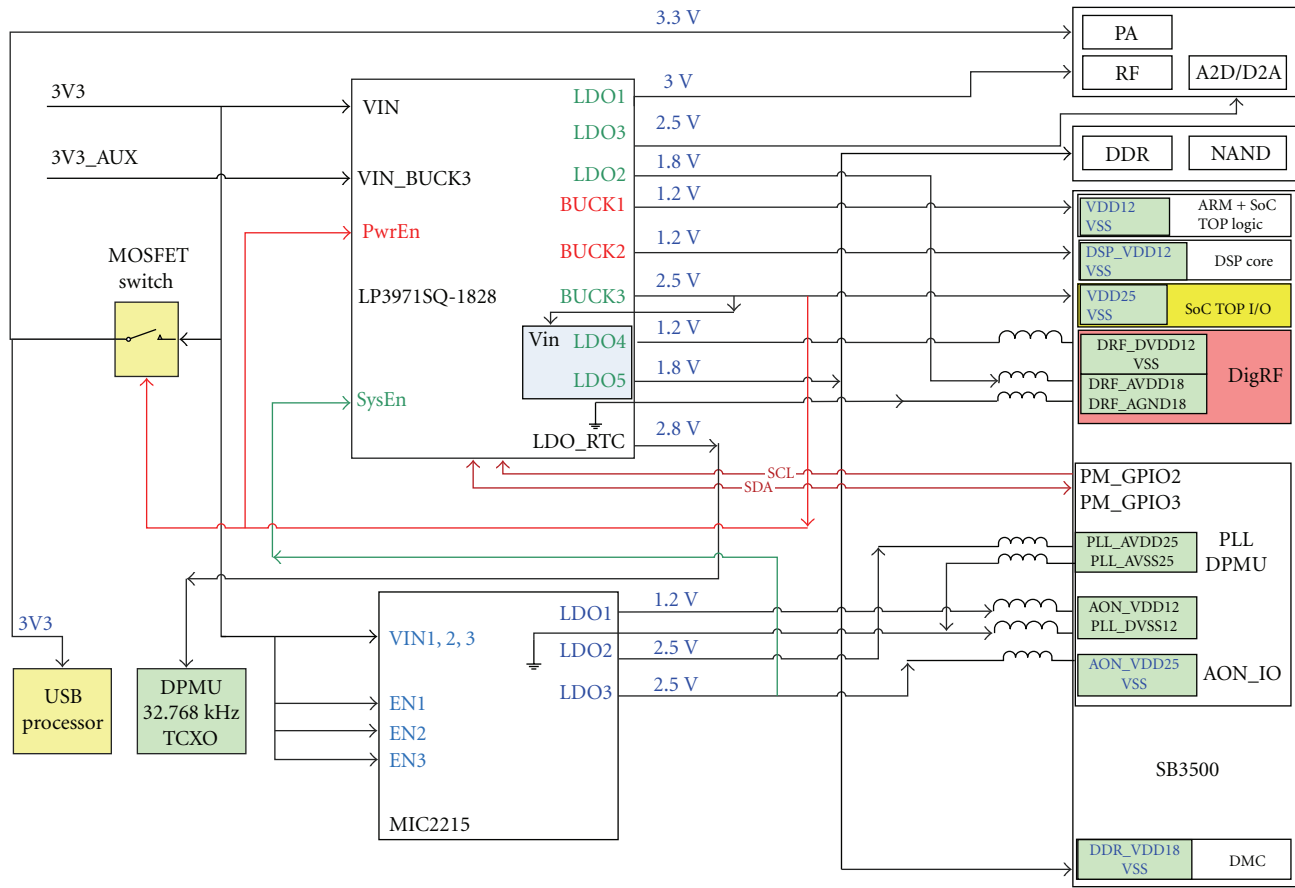


FIGURE 4: Power supply block diagram.

add, computing the polynomial modulus (galois field arithmetic support). Viterbi decoding is possible with 16 viterbi butterflies in parallel. Turbo decoding is supported for the constraint length of the convolutional codes of 3. There are also available vector operations which rearrange data into registers (packing/unpacking 8-bit to 16-bit data and 16-bit to 32-bit data, shuffling the elements of a pair of register, rotating register pairs, copying or shifting the accumulator into register). Digital signal processing typically uses fixed-point arithmetic. All the vector operations that do addition, subtraction, multiplies, and left-shift have a fixed-point version.

#### 4. Hardware Design Considerations

The maximum power available from the expresscard slot is around 2.2 W on 3.3 V, 1.2 W on 3.3 VAUX, and 0.6 W on 1.5 V. Thus the maximum useable power for the SDR board from the expresscard interface is 3.4 W. The power supply, illustrated in Figure 4, uses a PMIC and a triple Low Drop Out (LDO) for power domains lower than 3.3 V and a MOSFET switch for the 3.3 V power domains. PMIC power enable and system enable inputs are used to implement the sequence required by the SB3500 as shown in Figure 5. Once the SB3500 is running, DPMU may be programmed via the

ARM to turn off the unused power domains. The PMIC may also be programmed through the PM I2C interface (PM\_GPIO2 and PM\_GPIO3) to modify all the output voltages or to shut off the unused domains. For handheld applications this versatile scheme allows conserving battery power when the firmware is partially using the SoC hardware resources. The DPMU gets its clock from a low-frequency TCXO which keeps running as long as the card is powered. This way, the DPMU can run in sleep mode with the ARM powered down. After a complete power supply sequence, two resets are generated from the supervisor ICs: an asynchronous power on reset (nPOR) which is initializing the power management, PLL control, and debug interface blocks, and an asynchronous master reset (NRST) for the rest of the subsystems.

Some analog design criteria will be described in the following. A careful analog front-end design will result in less noise and signal interference, and as a consequence, less processing power will be needed in order to meet the minimum performance requirements.

The analog section uses an ultra-low-power mixed-signal Analog Front End (AFE) which integrates a dual 10-bit, 45 Msps receive Analog-to-Digital Converter (A2D) (RX), and a dual 10-bit 45 Msps Digital to Analog Converter D2A (TX) with a theoretical signal-to-noise ratio (SNR) of about



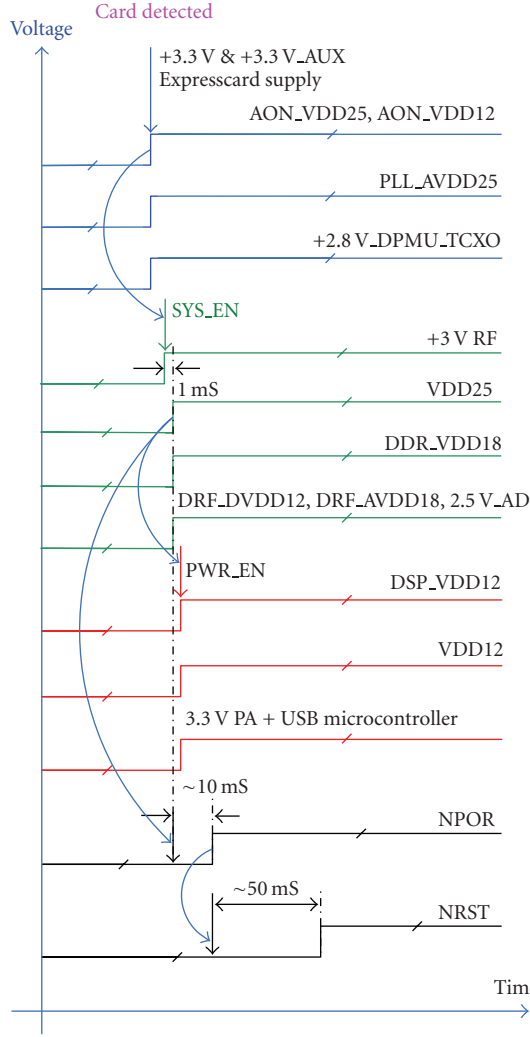


FIGURE 5: Power up sequence diagram.

52 dB for the RX and 57 dB for the TX. The maximum theoretical signal-to-noise ratio (SNR) for  $N = 10$  bits A2D, measured in dB, is given by a well-known equation:

$$\text{SNR(dB)} = 6.02N + 1.76. \quad (1)$$

In practice, the quantization noise is added to the A2D internal noise and harmonic distortion, resulting in a smaller numbers of usable bits as follows:

$$\text{SINAD(dB)} = \text{ENOB} * 6.02 + 1.76, \quad (2)$$

where ENOB is the effective numbers of usable bits and SINAD is the signal to noise and distortion ratio. The accuracy of the A2D highly depends on the quality of the clock being used. A clean and low jitter clock translates to an ENOB value closer to the theoretically computed value. The SNR as a function of clock jitter is equated by the following equation:

$$\text{SNR(dB)} = -20 \log(2\pi f_{\text{analog}} \cdot t_{\text{jitter}}), \quad (3)$$

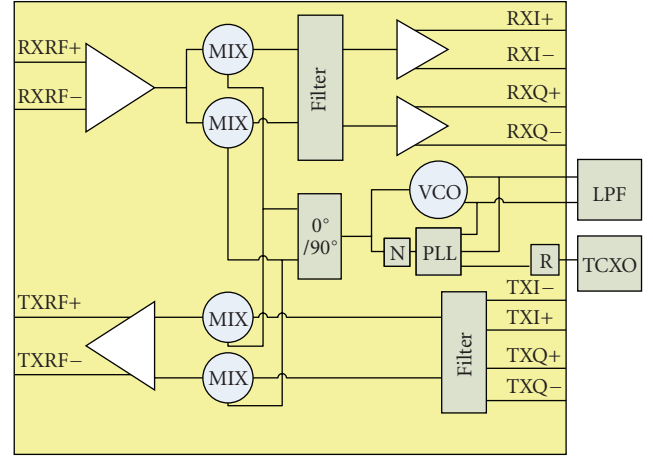


FIGURE 6: Transceivers simplified block diagram.

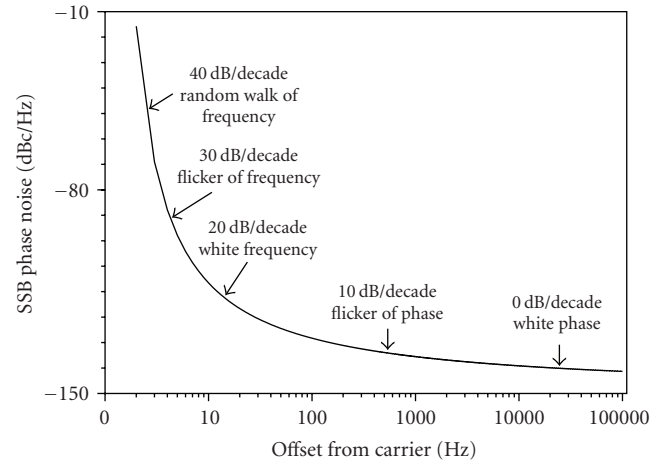


FIGURE 7: Typical phase noise curve for the low phase noise TCXO.

where  $f_{\text{analog}}$ , expressed in Hz, is the sampling rate and  $t_{\text{jitter}}$ , in seconds, is the RMS value of the jitter. From the previous equation it follows

$$t_{\text{jitter}} = \frac{10^{-\text{SNR}/20}}{2\pi \cdot f_{\text{analog}}}. \quad (4)$$

For a 10 bit analog-to-digital converter [6], at 22 MHz sampling rate, the above equations (2) and (4) leads to the following.

- (i) Maximum theoretical SNR is 61.96 dB.
- (ii) Theoretical SINAD with 8.77 ENOB is 54.6 dB.

To achieve the theoretical SNR, the jitter of the sampling clock must be less than 0.5 nanoseconds. Also, the external noise to the A2D plays a crucial role. Minimizing the switching noise from the power supply such that the A2D Power Supply Rejection Ratio (PSRR) stays in the  $\pm 0.4$  LSB, as specified by the A2D specifications, requires special design rules to be taken in consideration. First, improving the AFE total SNR was possible by employing independent power

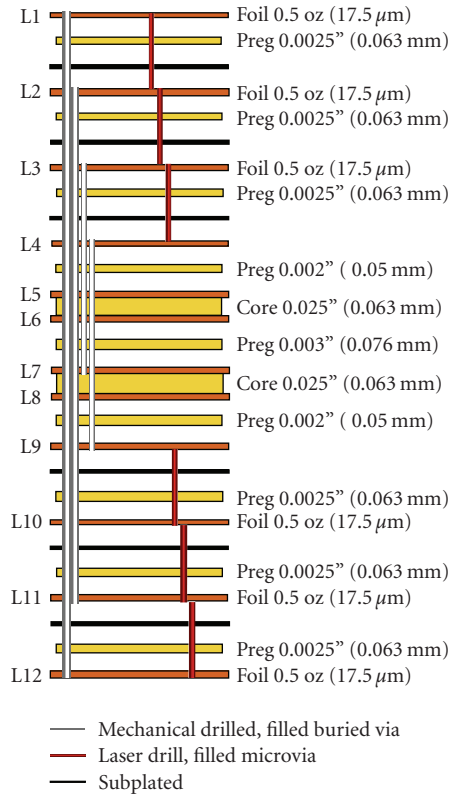


FIGURE 8: Layer stackup.

supplies for the analog side (+3 V) and for the digital side (+2.5 V) of the AFE. Second, we created an analog path for the sampling clock. Separating the analog from digital ground on the PCB in the AFE region usually does not bring the expected results because it creates large ground loops between the analog and digital sections. We choose a common ground plane split into analog and digital regions instead. However, keeping the ground plane noise free, in a small PCB size mixed signal design, is a major challenge.

Each ZIF transceiver integrates the Low-Noise Amplifier (LNA), the digital gain control, Voltage Controlled Oscillator (VCO), fast settling Sigma Delta fractional  $N$  synthesizer, and the programmable baseband filters. A simplified functional block diagram is illustrated in Figure 6.

The frequency reference for both transceivers is provided by a single clipped sinusoidal frequency, derived from very low phase noise oscillator.

Jitter and phase noise are different ways of quantifying the same phenomenon: the measure of the uncertainty at the output of an oscillator. Jitter is the time domain measure of the timing accuracy of the oscillator period and phase noise is a frequency-domain view of the noise spectrum around the oscillation frequency. There is not any known correlation between all sources of jitter; thus jitter cannot be predicted in practice [7]. In communication systems, the reference frequency phase noise will directly impair the overall performance by increasing the error vector magnitude EVM of the demodulator [8].

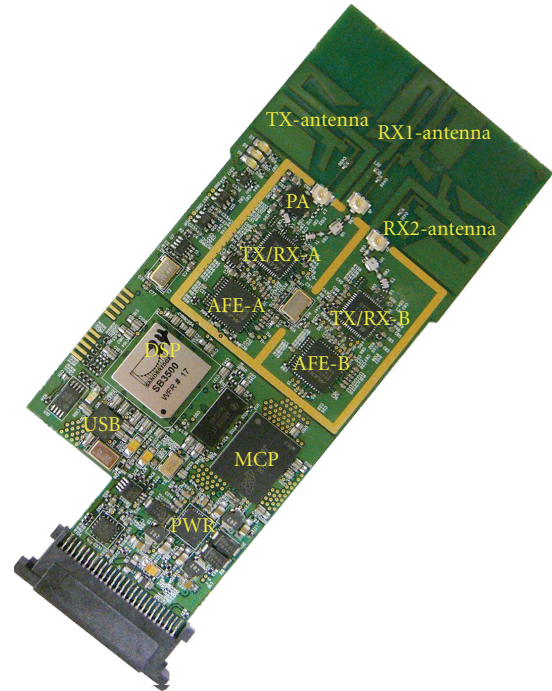


FIGURE 9: Expresscard top view, overall dimensions 48 × 108 mm.

Clipped sine wave exhibits less harmonic content; thus the induced noise in the analog section is less. To keep the noise as low as possible, the PLL circuits are supplied separately from an ultra-low-noise LDO with a PSRR of about 54 dB at 10 KHz. For the analog clock distribution we choose 50 Ohm impedance traces with simple DC blocking capacitors between the TCXO output and transceivers' PLL clock inputs. Running a strip line for the clock trace between two adjacent ground planes will significantly lower the clock noise, interference, and reflection (the load impedance is matched to the trace). Careful design of the clock distribution network is required to minimize the phase noise. Any clock distribution IC, based on our experience, will add extra phase noise. The phase noise degradation can be dramatic. For instance, a TCXO, with specified phase noise better than  $-145$  dBc/Hz at 10 KHz, may loose more than 20 dBc if the signal is buffered even with very low jitter and skew buffers. The reason is because the dominant noise type is flicker of phase, with a slope of 10 dB/decade, at around 10 KHz offset from carrier (see Figure 7) which will be amplified by the buffer's jitter component. To avoid the extra noise added by the clock distribution network we choose to use two separate oscillators one for the RF front end frequency reference and the other for the sampling clock.

## 5. Circuit Board Design and Noise Minimisation

Circuit board design complexity increases with increased component density and smaller board size. The lowest pitch component, in other words the largest ball density, determines the layer stackup configuration for the best optimized trace escape solution under the BGA packages.

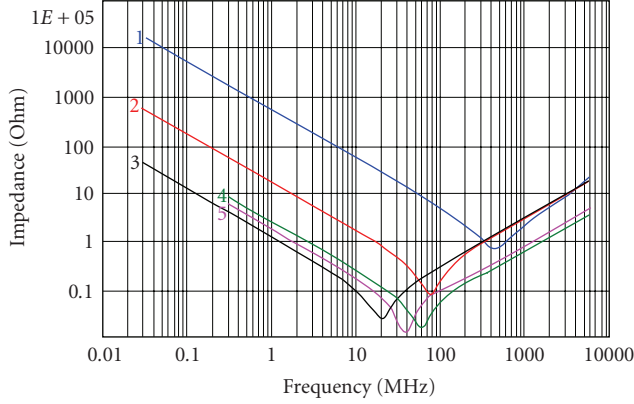


FIGURE 10: Simulation of impedance versus frequency for different used types of Murata capacitors [9]: (1) 330 pF 0402 size, X7R, GRM series, (2) 10 nF 0402 size, X7R, GRM series, (3) 0.15 uF, 0402 size, Y5R, GRM series, (4) 10 nF, 0306 size, X6S, LLL series, (5) 0.1 uF 0204 size, X6S, LLL series.

Even though there are available standard layer stackup configurations, trace density on small size designs often requires nonstandard solutions for the printed circuit board design. Our board is designed on a 12-layer stack as presented in Figure 8, using 0.008" (0.20 mm) mechanical buried vias, 0.010" (0.25 mm) through hole vias, and 0.004" (0.10 mm) laser drilled and filled microvias. The overall board thickness is about 0.040" (1 mm). The critical routing component is the SB3500 SoC with 529 balls distributed on 11 × 11 mm array with a 0.5 mm pitch. The primary escape layers for the SB3500 signals are L1, L2, L3, and L4. Supply layers are L4-L5 and L8-L9 which are using a buried capacitance (ZBC2000) prepreg between them. Layers L7 and L8 are used for clock and differential routes. Layers L10 and L11 are carrying signals while layers L1 and L12 are used for component placement and ground plane.

All signal layers also carry ground planes to increase the noise immunity between signal routes [5]. For the SB3500 escaping signals, triple stacked microvias are necessary on layers L1-L2, L2-L3, and L3-L4. The board must be built symmetrically to equalize interlayer stresses as manufacturing process requirement (to prevent warpage) hence the existence of stacked microvias on L9-L10, L10-L11, and L11-L12. Buried vias are used for transferring signals from L2, L3, and L4 on lower signal layers (L10, L11) but also to create shorter paths for the filtering capacitors placed on the bottom layer L12, below the BGA packages placed on L1. All power supplies are using planes for routing, distributed on the supplying layers. The small power supplies are using signal layers L3 and L10 for routing. The PCB component placement for this SDR design is shown in Figure 9.

In digital systems, filtering capacitors are used to suppress the noise generated by the switching clock at least up to the third or fifth harmonic. The high-frequency noise component must be suppressed near or as close as possible to the source. Sometimes this is impossible, as the noise source, the supply ball of a PLL from a BGA, for instance, can be reached only with a trace which becomes an RF emitter.

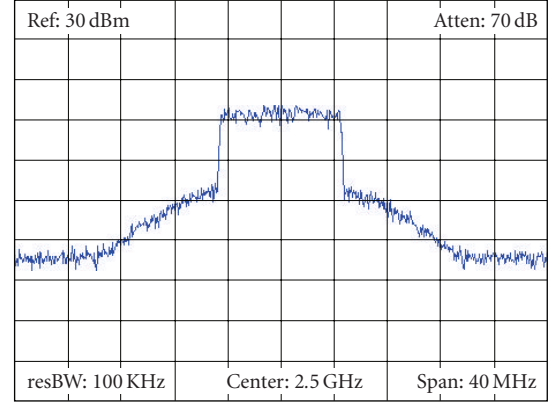


FIGURE 11: QPSK WiMax TX spectrum at +21 dBm output power.

TABLE 1: Measurement results for WiMAX Wave 2 Category 1 mobile station.

Measurement		Value
Max power	Dual RX mode	1.3 W
	TX-RX mode	3.3 W
EVM at 21 dBm TX power		-27.5 dB
Noise level on digital PWR		max 50 mVpp
Noise level on AFE and RF PWR		max.15 mVpp
Symbol clock error		<0.9 ppm
I/Q skew		<300 pS

Each capacitor on the circuit board can be represented as an equivalent series circuit consisting of an ideal capacitor  $C$ , an Equivalent Series Inductance (ESL), and an Equivalent Series Resistance (ESR).

For such a capacitor, the equivalent impedance  $Z_c$  is

$$Z_c[\text{ohm}] = j\omega \cdot \text{ESL} + \frac{1}{j\omega C} + \text{ESR} \quad (5)$$

with minimum value at the resonant frequency:

$$f_{\text{res}} = \frac{1}{2\pi\sqrt{LC}}, \quad (6)$$

where  $C$  is the capacitor value and  $L$  is the parasitic inductance of the capacitor as given in the specifications. From (5), the best filtering capacitor will have  $\text{ESR} = 0$  and the lowest possible ESL. The filtering capacitor will have the best noise suppressing at the frequency which will minimize the equivalent impedance. Lower equivalent cap inductance and resistance is achieved through short connection traces between the cap terminals and BGA balls.

Unfortunately, routing the BGA power balls to the ground and power planes is done using traces and vias which are both inductive and resistive. Physically, a capacitor can be installed on the same layer with the BGA balls, near the BGA package or on the opposite side, below package. In both situations the capacitor is requiring at least two vias to connect the capacitor terminals to the power planes.

Equation (6) and Figure 10 tell us that in order to suppress a large noise frequency range requires two or three

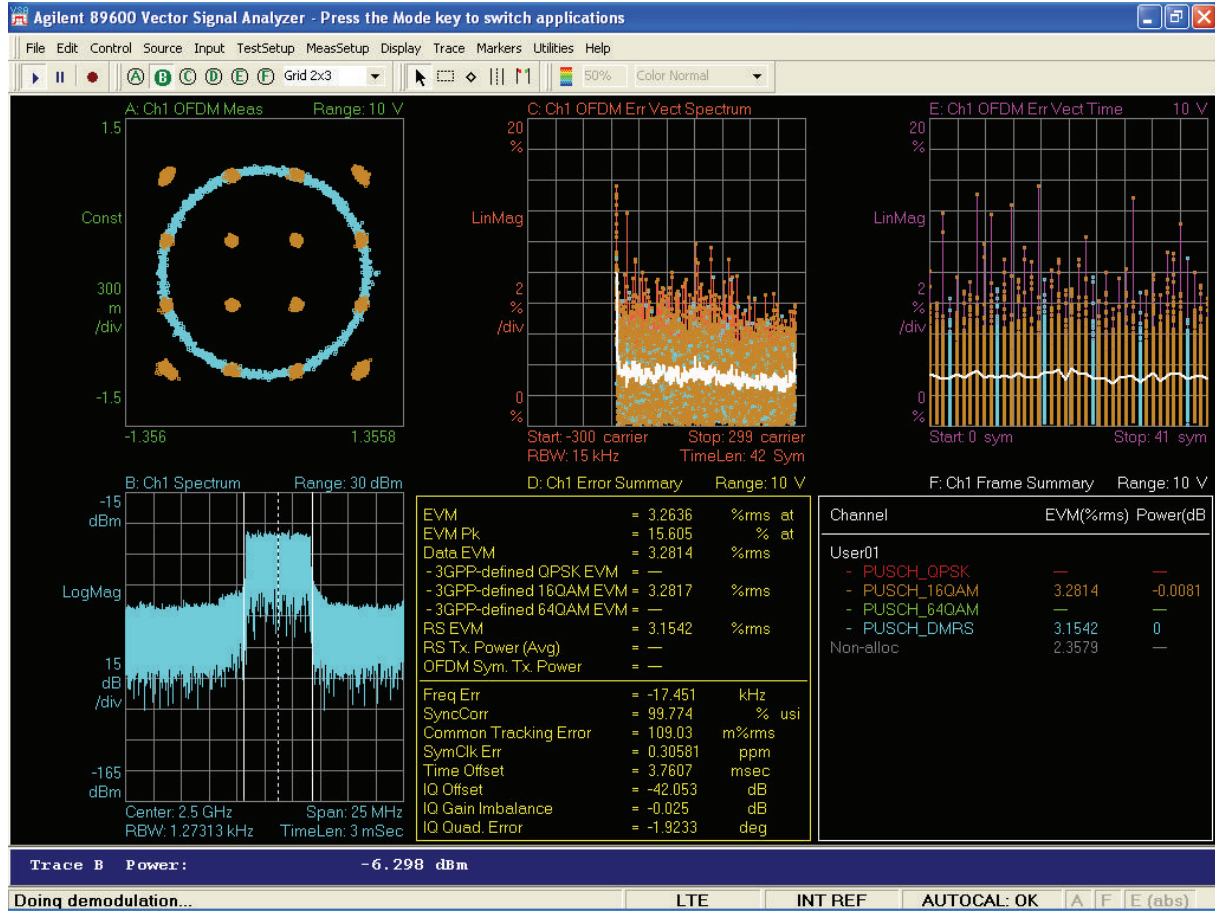


FIGURE 12: Vector signal analyzer LTE uplink screen snapshot: 16QAM PUSCH data (yellow) and PUSCH reference signals (light blue) constellation for one user at +21 dBm output power with 3.1% rms EVM.

standard capacitors with different parameters connected in parallel thus, increasing the number of capacitors to about 500 for a quite small board and, making almost impossible the low-noise routing without even taking the added cost in consideration.

One solution to this problem is the reversed geometry low ESL filtering capacitors. Figure 10 shows the difference between using three standard filtering capacitors mounted in parallel (330 pF, 10 nF, and 0.15 uF), versus two parallel 10 nF and 0.1 uF reversed geometry low ESL capacitors. As seen, the smaller impedance, in the frequency range 25 MHz–1 GHz, is achieved by capacitors 4 and 5 connected in parallel. This way, the total number of filtering capacitors was reduced to about 2/3 compared with standard capacitors. Using high-value capacitors (1 uF–2.2 uF) for the lower-frequency range is still necessary, but the number of capacitors is small and equally distributed on the printed circuit board.

## 6. Platform Validation

Design validation was performed against the WiMAX Forum Mobile Radio Conformance Tests (MRCT) [10] for Category 1 mobile station. Next, we reproduce the most critical

measurements as Error Vector Magnitude (EVM) at the maximum transmit power, spectral mask emission, and maximum receive sensitivity, defined by the standard. The measurement results are illustrated in Table 1. The maximum EVM required by the MRCT specification is  $-24$  dB (6% RMS) while we measured  $-27.95$  dB at 21 dBm transmit power. Figure 12 illustrates the Vector Signal Analyzer (VSA) screen capture for a QPSK waveform at 21dBm transmit power. As shown in Figure 11, at 6 MHz from the central carrier we measured an attenuation of 19 dB, compared to 13 dB required by the WiMAX standard.

For WiMAX Wave 2, the total processor utilization is around 75% with all cores running at 600 MHz.

## 7. Conclusions

We presented a 4G low-cost SDR platform based on the SB3500 DSP from Sandbridge Technologies. Practical design considerations as well as physical measurements and performance data were described throughout the paper. As far as we are aware of, this is the only existing low-power, low-cost, positive gain-MIMO antenna based- [11, 12] SDR platform capable of executing wireless communication protocols such

as WiMAX, WLAN 802.11 g, and LTE. For instance, WiMAX Wave II, TX and RX combined, executes in 1.2 GHz which represents two SB3500 cores.

## References

- [1] <http://www.sdrforum.org/>.
- [2] J. Declerck, E. Umans, A. Dejonghe, M. Trautmann, M. Glassee, and L. Van der Perre, "A software development and validation framework for SDR platforms," in *Proceedings of the Technical Conference and Product Exhibition (SDR '08)*, Washington, DC, USA, October 2008.
- [3] M. S. Mora, G. Corley, J. Lotze, and R. Farrell, "Experiences in the co-design of software and hardware elements in a SDR platform," in *Technical Conference and Product Exhibition (SDR '08)*, Washington, DC, USA, October 2008.
- [4] W. Xiang, T. Pratt, and X. Wang, "A software radio testbed for two-transmitter two-receiver space-time coding OFDM wireless LAN," *IEEE Communications Magazine*, vol. 42, no. 6, pp. S20–S28, 2004.
- [5] R. Hartley, "Controlling radiated EMI through PCB stackup," L3-communications, Avionic Systems, August 2000.
- [6] MAX19706 datasheet, <http://datasheets.maxim-ic.com/en/ds/MAX19706.pdf>.
- [7] R. Poore, "Overview on Phase Noise and Jitter," Agilent EESoft EDA <http://cp.literature.agilent.com/litweb/pdf/5990-3108EN.pdf>.
- [8] R. D. Gitlin and E. Y. Ho, "The performance of staggered quadrature amplitude modulation in the presence of phase jitter," *IEEE Transactions on Communications*, vol. 23, no. 3, pp. 348–352, 1975.
- [9] Murata Chip S-parameter and Impedance library, <http://www.murata.com/designlib/mcsil/index.html>.
- [10] <http://www.wimaxforum.org>.
- [11] E. Surducan, D. Iancu, and J. Glossner, "Modified printed dipole antennas for wireless multi-band communication, part I," US patent no. 7,034,769 B2, 2006.
- [12] E. Surducan, D. Iancu, and J. Glossner, "Modified printed dipole antennas for wireless multi-band communication, part II," US patent no. 7,095,382 B2, 2006.



## Research Article

# Software-Defined Radio Demonstrators: An Example and Future Trends

**Ronan Farrell, Magdalena Sanchez, and Gerry Corley**

*Centre for Telecommunications Value Chain Research, Institute of Microelectronics and Wireless Systems,  
National University of Ireland Maynooth, Maynooth, Co. Kildare, Ireland*

Correspondence should be addressed to Ronan Farrell, [ronan.farrell@nuim.ie](mailto:ronan.farrell@nuim.ie)

Received 30 September 2008; Accepted 14 January 2009

Recommended by Daniel Iancu

Software-defined radio requires the combination of software-based signal processing and the enabling hardware components. In this paper, we present an overview of the criteria for such platforms and the current state of development and future trends in this area. This paper will also provide details of a high-performance flexible radio platform called the maynooth adaptable radio system (MARS) that was developed to explore the use of software-defined radio concepts in the provision of infrastructure elements in a telecommunications application, such as mobile phone basestations or multimedia broadcasters.

Copyright © 2009 Ronan Farrell et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

## 1. Introduction

In recent years the technologies required to implement the concept of software-defined radio (SDR) have matured, and the SDR Forum presents a tier-based taxonomy for the capabilities of various SDR systems [1]. Systems are now appearing that offer flexibility and adaptability to system developers—providing advantages when addressing the issues of constrained spectrum resources, increasingly rapid changes in wireless standards, and cost-effectively developing products for niche markets [2, 3]. As the required technologies have matured, we are now seeing SDR implementations delivering wide bandwidth applications with a high quality of service, for example, in mobile data communications such as WiMAX-e. In the future it can be imagined that SDR architectures will be increasingly used to deliver telecommunication services such as mobile telephony, digital TV and radio broadcasts and heterogeneous combinations such as streaming video in the mobile environment.

As spectrum is a finite-shared resource that is increasingly congested with existing users, obtaining access to spectrum for the delivery of new services is increasingly difficult. Frequency agile SDR systems offer a solution where the flexible SDR radio can avail of an unused slice of spectrum, temporarily, to deliver the service. Originally this concept

met strong resistance from existing spectrum holders and the regulators, however, recently there has been increasing interest from the regulators (who can allow greater diversity of services) and from spectrum holders (who can utilize their spectrum more profitably). One initiative that supports this trend is the developing discussions in Europe on “Wireless Access Platforms for Electronic Communications Services (WAPECS)” where it is proposed that some services may opportunistically use spectrum, if available, in regional and temporal bases [4]. Though at an early stage, these initiatives suggest new opportunities for telecommunication services.

In this paper we will present an overview of the challenges in designing an SDR platform that can be used for research or deployment. We will discuss the issues that need to be addressed and the current state-of-the art in software-defined radio demonstrators. This will then be followed by a detailed description of the maynooth adaptable radio system (MARS), its design criteria, architecture, and some use cases. Finally the paper will be concluded with some comments on the future direction of experimental SDR platforms.

## 2. Design Criteria for SDR Platforms

Software-defined radio platforms are integrated systems of software and hardware that enable SDR applications to be

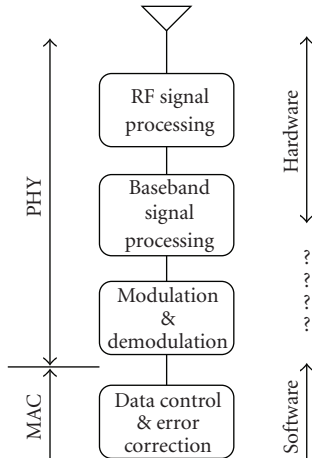


FIGURE 1: Partitioning between software and hardware in an SDR system.

developed and evaluated. Of the two, the software aspects are relatively more mature, and current work in this area focuses on performance enhancement and cognitive radio techniques. The hardware aspects of a platform consist of the radio-frequency (RF) elements, some baseband signal processing and communications link to the software-based signal processing element—perhaps a DSP, FPGA, or a general purpose processor (GPP). One aspect of the software-defined radio concept is that flexibility can be delivered through software. An often overlooked corollary is that the hardware performance to support that flexibility is more challenging than for a single-mode implementation, and optimal solutions remain elusive [5]. This section will comment on some of these issues and how they impact on the hardware architecture of software radio platform.

**2.1. Partitioning of Resources.** The software-defined radio philosophy represents a trend in electronic devices from transistors to software. This has been facilitated by the rapid increase in software capabilities and processing power. In software-defined radio the argument is to implement as much of the radio as possible in software and to control the remaining hardware features. However, the choice of where the partition between hardware and software has a fundamental impact on the design of any SDR platform [6, 7].

One desirable partitioning of functionality is to take all signal processing into the software domain and that only I- and Q-sampled data is passed into the hardware domain. In this scenario the hardware element of the system need undertake no signal processing. This places a severe performance requirement upon the software processing element, particularly where bandwidths in excess of 1 MHz need to be supported. Alternatively some of the software processing load may be allocated to customized hardware (often in the form of an embedded FPGA or a specialist DSP device). In this scenario the load is shared but FPGAs are expensive and arguably offer less flexibility. One of the

important issues to consider when choosing the partitioning is the data communications protocol between the different elements. For unprocessed IQ signals, for every 1 MHz of spectrum, that is, being supported a data link capacity of 40 Mbps is required, assuming 16 bit samples and 8 b/10 b encoding. This is doubled for duplex transceivers. This severely limits the bandwidth capabilities of platforms that are required to connect to standard interfaces on general purpose computers. More complex, higher performance links are possible that will allow greater bandwidths to be supported, for example, Gigabit Ethernet or PCIexpress. Alternatively if on-board processors are included, some local processing could greatly minimize the data throughput requirements.

**2.2. Frequency Flexibility.** Software-defined radios come in two varieties—those that are modulation scheme (or waveform) flexible within a specific frequency range, or those that are waveform and frequency agile. Implementations of the former are more common as it does not require any significant modification to traditional hardware. Modern mobile wireless systems (UMTS, IEEE 802.16) are often implemented in this manner. Frequency agility offers many more benefits such as flexible use of spectrum or dynamic adaptation to different wireless networks. Frequency flexibility places several severe constraints on the design of the supporting radio frontend (RFE).

- (i) Programmable carrier frequency generation.
- (ii) Antenna, filter, and passive network designs.

Frequency selection requires the ability to generate a carrier frequency within the required range. This is normally achieved through the use of a local oscillator. The local oscillator can be generated in many different ways depending on the degree of flexibility and phase noise performance required [8]. These two criteria tend to be inversely related, however there is continual improvement in this area and with careful design; performance and flexibility can be achieved.

Frequency agility places more severe constraints on the design of the passive elements within a radio: the antenna; filters; matching networks. Normally a radio is designed with a narrowband or multiband perspective—multiband is where a finite set of narrowband signals are used. In this scenario filters can be designed to select the band of interest and minimize the effect of other potentially interfering signals or noise. Similarly antennas and matching networks for the low-noise amplifiers and power amplifiers are optimized for maximum gain in the band of frequencies of interest. Where multiband systems are required, the common approach is to switch between the appropriate narrowband solution. Providing flexibility over a wider band means that traditional filter solutions cannot be used and to date useful programmable flexible filters do not exist. Wideband antennas and matching networks can be designed but they are suboptimal. This implies a reduction in efficiency depending on the degree of flexibility required. The lack of frequency selectivity has a significant impact on



the issue of interference, energy efficiency, and sensitivity of the final design.

**2.3. Interference Management.** Frequency flexible radio receivers cannot have the same band select filtering as traditional radios and are vulnerable to interference, both from external sources and self-generated phase-noise from a local transmitter. Considering the external sources first, a wideband radio receiver covering any of the communications ranges (e.g., 700–950 MHz or 1800–2500 MHz) will be exposed to legitimate transmissions from a variety of sources—mobile phone transmissions, WiFi, television. To implement a standards compliant radio receiver, it is necessary to be able to function in the presence of other transmissions to the required level of sensitivity. For example, in GSM, you must be able to receive a  $-98$  dBm signal in the presence of a  $0$  dBm blocker. Requirements such as these have significant impacts on the design of your RF receivers. In a radio receiver it can be shown that reduced filter performance can be achieved at the expense of increased analog-to-digital conversion sensitivity. In the absence of filtering, it can be shown that at least 14 bits of dynamic range are required for an acceptable bit-error rate, and 16 bits would be desirable. Achieving 16 bits analog-to-digital conversion for bandwidths greater than 10 MHz is difficult and expensive in terms of power and cost. SDR platforms must decide whether they attempt to be standard compliant or best effort. For ease of implementation, most platforms ignore the interference issue, and the user selects a frequency range with minimal interference.

The second issue of self-generated interface is more challenging. Modern transmitters are good at controlling phase noise and spurious out-of-band components and, in many scenarios, the receive and transmit bands are sufficiently distant to enable robust filtering. This is important as there can be over 120 dB difference in power levels in a mobile phone handset or 150 dB for a GSM basestation. In the absence of such filtering, transmitter phase noise can leak into the receive path and swamp any received signal. This is problematic as the transmitter and receiver are coincident and thus unlike external transmissions will not be attenuated by distance. This issue is currently without a good solution. The issue can be minimized if a TDD-communication scheme is selected.

**2.4. Transmitter, Receiver or Transceivers.** There are many applications where it is not necessary to implement a transceiver system. If true, then many issues are greatly simplified: improved data throughput; no concerns on self-generated noise; lower cost. Receiver-only applications are popular in the cognitive radio space and in multimedia receivers. In cognitive radio one of the main challenges is in spectrum sensing and identification of existing communication schemes. This is a receiver-only application and benefits from any reduction in self-generated noise. For broadcast applications, such as television, the operators require only transmitters and receivers for the clients. However, for most

wireless communications, bidirectionality is required and a full-transceiver system will be needed.

### 3. Review of Existing SDR Platforms

There are a large number of experimental SDR platforms that have been developed to support individual research projects. A selection of these platforms is included in [9–21]. The various experimental SDR platforms have made different choices in how they have addressed the issues of flexibility, partitioning, and application. To highlight the variety of architectures, four popular platforms will be discussed briefly prior to introducing the maynooth adaptable radio system.

**3.1. Universal Software Radio Peripheral (USRP).** The USRP is one of the most popular SDR platforms currently available and it provides the hardware platform for the GNU Radio project [8, 9]. The first USRP system, released in 2004, was a USB connected to a computer with a small FPGA. The FPGA was not only used primarily for routing information but also allowed some limited signal processing. The USRP could realistically support about 3 MHz of bandwidth due primarily to the performance restrictions of the USB interface. The second generation platform was released in September 2008 and utilizes gigabit Ethernet to allow support for 25 MHz of bandwidth. The system includes a capable Xilinx Spartan3 device which allows for local processing. The radio-frequency performance of the USRP is limited and is more directed toward experimentation rather than matching any communications standard.

**3.2. Kansas University Agile Radio (KUAR).** The KUAR platform was designed to be a low-cost experimental platform targeted at the frequency range 5.25 to 5.85 GHz and a tunable bandwidth of 30 MHz [11]. The platform includes an embedded 1.4 GHz general purpose processor, Xilinx Virtex2 FPGA and supports gigabit Ethernet and PCI-express connections back to a host computer. This allows for all, or almost all processing, to be implemented on the platform, minimizing the host-interface communications requirements. The platform was designed to be battery powered thus allowing for untethered operation. The KUAR utilizes a modified form of the GNU Radio software framework to complete the hardware platform.

**3.3. NICT SDR Platform.** The Japanese National Institute of Information and Communications Technology (NICT) constructed a software-defined radio platform to trial next generation mobile networks [12]. The platform had two embedded processors, four Xilinx Virtex2 FPGA, and RF modules that could support 1.9 to 2.4 and 5.0 to 5.3 GHz. The signal processing was partitioned between the CPU and the FPGA, with the CPU taking responsibility for the higher layers. An objective of this platform was to explore selection algorithms to manage handover between existing standards. To this end, a number of commercial standards were implemented, for example, 802.11a/b/g, digital terrestrial

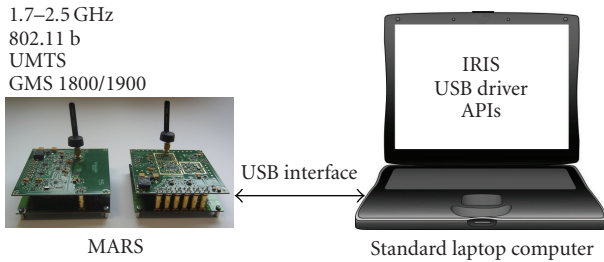


FIGURE 2: MARS architecture.

broadcasting (Japanese format), wCDMA, and a general OFDM communication scheme.

**3.4. Berkeley Cognitive Radio Platform.** This platform is based around the Berkeley emulation engine (BEE2) which is a platform that contains five high-powered Virtex2 FPGAs and can connect up to eighteen daughterboards [13]. In the Cognitive Radio Platform, radio daughterboards have been designed to support up to 25 MHz of bandwidth in an 85 MHz range in the 2.4 GHz ISM Band. The RF modules have highly sensitive receivers and to avoid self-generated noise operate either concurrently at different frequencies (FDD) or at the same frequency in a time-division manner (TDD). This cognitive radio platform requires only a low-bandwidth connection to a supporting PC as all signal processing is performed on the platform.

#### 4. Maynooth Adaptable Radio System

The maynooth adaptable radio system (MARS) has been in development since 2004 and had the original objectives of a programmable radio front-end that was to be connected to a personal computer (PC) where all the signal processing is implemented on the computers general purpose processor (Figure 2) [14]. The platform was to endeavor to deliver a performance equivalent to that of a future mobile telephony base station and the wireless communication standards in the frequency 1700 to 2450 MHz. The software framework selected for initial development was the IRiS framework (Implementing Radio in Software) from our collaborators in Trinity College Dublin. This section will present some of the design criteria, issues encountered, our solutions, and then some results from the final implemented system.

The platform high-level objectives drive a range of technical design choices.

**Future Base stations.** Most 2 G base stations supported a frequency band no greater than 5 MHz, adjustable within the full GSM band. However, there is strong interest in a base station that could simultaneously support distinct and separated bands of frequencies—enabling base station sharing between operators or where operators may own different bands of frequency. This drove a specification that full-band support should be explored, 70 MHz, over an approximately 700 MHz range. Since the start of the project, wideband schemes such as wCDMA, WiMAX have become

increasingly popular, and bandwidths of at least 25 MHz need to be supported.

**General Purpose Computer Connected.** Much of the work on software-defined and cognitive radios has been undertaken by researchers who are more familiar with general-purpose processors than with FPGA or DSP devices. All available software frameworks are PC-based and for our project we utilized the IRiS SDR framework developed by our collaborators, Trinity College Dublin [15]. Similar to the USRP, it was necessary to provide an interface with a general purpose computer in which modulated baseband data is passed between the computer and the radio platform. This can be easily identified as a performance bottleneck as one must choose a standardized interface. At the start of this project, widely used high-performance interfaces were limited. The USB 2.0 standard was selected as most suitable despite its obvious performance limitations. The platform design was designed to be modular so that this performance bottleneck could be removed when higher performance interfaces became available.

**Communication Modes between 1700 and 2450 MHz.** This range of frequencies is comparatively narrow but is the most congested frequency range for personal communications. As a project specification we identified the following communication modes that were to be supported:

$$\begin{array}{ll} \text{GSM1800} & \text{PCS1900} \\ 802.11 \text{ b} & \text{UMTS(wCDMA)}. \end{array} \quad (1)$$

In addition, the Irish communications and spectrum regulator (ComREG) licensed to our university two 25 MHz bands of spectrum at 2.1 and 2.35 GHz.

**4.1. Design Issues.** To determine the RF system specifications it was necessary to analyze the individual parameters and spectral masks for each standard and integrate them to produce a single worst-case specification. The primary parameters of interest for the design of the platform are receiver sensitivity, receiver third-order intermodulation product (IP3), receiver noise figure (NF), transmitter power levels, and transmitter phase noise. These parameters determine the blocking performance of the receiver, the spectral and spurious masks of the transmitter, and the expected receiver bit error rate.

One of the most challenging requirements is that of capturing the minimum allowable signal in the presence of blockers. Under the assumption that strong filtering does not exist (as the system is frequency flexible), the radio system must have sufficient dynamic range for digital signal processing to extract the desired signal in the presence of blockers and interferers. Figures 3 and 4 show typical interference profiles for the GSM and wCDMA standards. The GSM standard (at all frequencies) presents the most challenging requirement as it requires successful reception of a  $-104$  dBm signal (in the base station,  $-102$  dBm for a GSM/GPRS handset) in the presence of a 0 dBm blocker. As

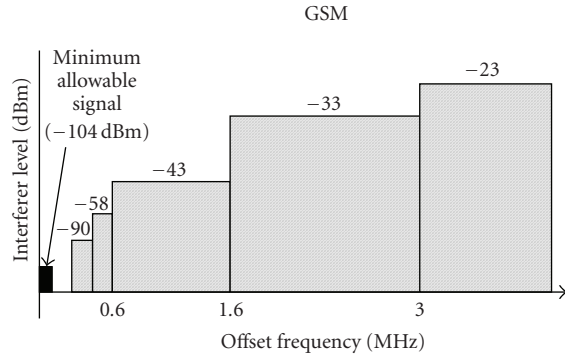


FIGURE 3: GSM receiver interference profile.

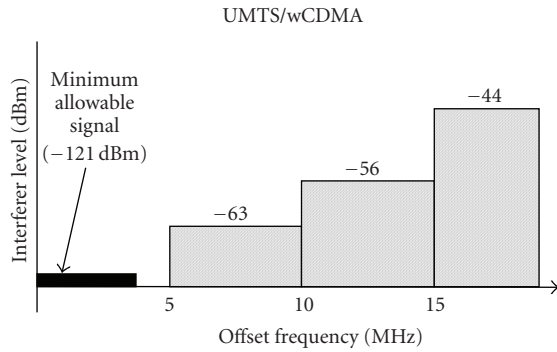


FIGURE 4: wCDMA receiver interference profile.

TABLE 1: RF specifications for various standards.

	GSM	UMTS	802.11 b
Noise figure (dB)	9	9.6 <sup>(2)</sup>	9 <sup>(3)</sup>
IIP2 (dBm) <sup>(1)</sup>	43	8.0	10
IIP3 (dBm)	-18	-21.0	-18

<sup>(1)</sup> IIP2 is required for zero-IF or low-IF architectures

<sup>(2)</sup> Assuming a processing gain of 25 dB

<sup>(3)</sup> Assuming a processing gain of 10.4 dB

our signal capture band is targeted at a complete communication band (e.g., the complete GSM band), blockers of that magnitude can be expected in the receive bandwidth of our platform. To capture the smallest required signal in the presence of such a blocker would suggest an analog-to-digital converter with dynamic range of over 106 dB, assuming an ideal receive signal chain. As a first-order approximation this is acceptable though a more detailed analysis shows that for a given bit-error rate, a lower dynamic range can be used [22].

The following table displays the receiver requirements for each of the communication standards. A composite specification for the receiver can be calculated by taking the most stringent requirement.

For the MARS platform, it was decided to go with a direct conversion architecture for both receiver and transmitter (Figure 5). Selecting an appropriate intermediate frequency in a frequency flexible system is difficult and thus a direct-conversion architecture allows one to avoid this issue. On the other hand, this approach places additional constraints

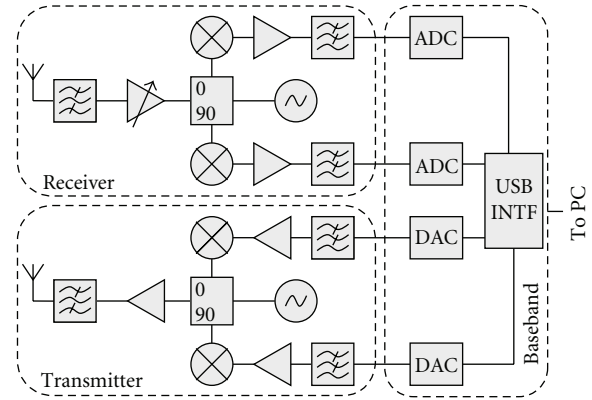


FIGURE 5: MARS platform architecture.

on the receiver, with signal chain performance dependent on linearity and to a large degree on IIP2 performance. In addition there have historically been issues with local-oscillator leakage resulting in dc distortion in the receiver. As most communication schemes have content at and near dc, this has been a reason to avoid direct conversion architectures in favor of low-IF or heterodyne solutions. Recently-released products have shown significant improvements and direct-conversion solutions are increasingly viable.

Given our direct-conversion architecture, the performance of the data-converters is important. We used 16 bits data converters in each direction so as to provide the necessary receive sensitivity and to minimize out-of-band transmit noise. The performance bottleneck of the overall platform is that of the USB connection which is limited to a sustained throughput of 256 Mbps. Our ideal target bandwidth of 70 MHz would require a data rate of approximately 10 Gbps—beyond the scope of any standard PC interface. In 2006, the best choice we had available was USB 2.0 which had a maximum sustained throughput of about 380 Mbps, allowing us a bandwidth of about 3 MHz (simplex) or 1.5 MHz (duplex). Modifying the sample resolutions allows us to double our throughput. It is acceptable to reduce the transmitter resolution as typically 60 dB of SNR will suffice, yielding a 25% increase in throughput. This was the fundamental performance bottleneck for our platform. There are only two solutions: place a processor or FPGA on the board or use a higher performance link. For the initial development, these options were not followed and the RF performance was throttled to match the USB interface. A modular design for the RF and baseband units was followed so that the overall platform could benefit from improvements in the data link throughput.

The following sections detail some of the components selected. In many cases it is easier to select wideband components rather than frequency agile components. With wideband components the complexity then resolves to the quality of the local oscillator, the data converters, and the passive structures (filters and matching networks). Local oscillators are a mature technology and phase-lock-loops (PLLs) are excellent at delivery agility and low noise. The

passive structures remain the most difficult and this issue is addressed by keeping any filters as relaxed as possible.

**4.2. Receiver.** In a direct-conversion receiver architecture, there is a direct tradeoff between RF band select filtering and the performance requirements of the analog-to-digital converter (ADC). In the absence of strong filters, the ADC must have sufficient resolution to support the dynamic range required to separate interferers from weak signals. An ADC with a signal bandwidth of 70 MHz and 106 dB (in excess of 17 bits) resolution is highly challenging but devices available at the time of development were capable of delivering 16 bit performance at high speeds though with high power consumption. We selected a family of pin-compatible ADCs from Linear Technologies, Calif, USA that can deliver up to 105 MSps (LTC220\* family). This will enable lower performance ADCs to be used seamlessly where the baseband signal processing cannot support higher speeds.

The RF low-noise amplifier selected was the Freescale MBC13720. This part is a low-noise amplifier with bypass switch. It generates a gain of 12 dB and noise figure of 1.55 dB at a frequency of 2.4 GHz. The LNA is able to operate in a frequency range from 400 MHz to 2.4 GHz. It features two enable pins to control the amplification stage which are software-controlled. The gain at this stage had limited programmability. For noise-mitigation maximizing early-stage gain is the preferred option, with greater gain control available at the baseband stage.

The performance of the demodulator is important in a direct-conversion architecture. The AD8347 device, from Analog Devices was chosen. It is a direct quadrature demodulator with RF and baseband automatic gain control (AGC) amplifiers. Its noise figure (NF) is 11 dB at maximum gain and it provides excellent quadrature phase accuracy of 1° and I/Q amplitude balance of 0.3 dB. This high accuracy is achieved by the polyphase filters employed by the local oscillator quadrature phase splitter. The dc offset problem is minimized by an internal feedback loop. Any remaining dc-offset effects could be corrected by digital correction but this was not implemented in the current prototypes.

In a frequency flexible system an agile local oscillator is required. Often a clock-data recovery circuit would be used to lock onto the transmission frequency, however in an SDR architecture a band of frequencies are captured and clock-recovery is undertaken digitally. The primary criteria for the local oscillator, in an SDR RF front-end, are agility and low-phase noise. We selected a low-power delta-sigma Fractional-N PLL from national semiconductor (LMX2470) with the MiniCircuit VCO ROS-2500. The sigma-delta modulated fractional-N divider has been designed to drive close-in spur and phase noise energy to higher frequencies. The modulator order is programmable up to fourth order, permitting us to alter the phase noise characteristics at different frequency offsets. The device can operate in the range 500–2600 MHz with a phase noise of –200 dBc/Hz. It is optimally operated in a smaller range but this can be adjusted by changing the local oscillator frequency.

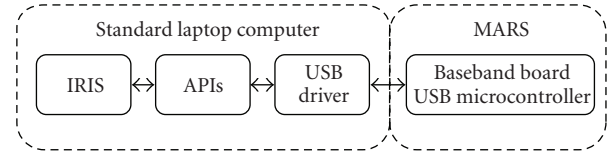


FIGURE 6: Computer to MARS transceiver interface.

**4.3. Transmitter.** The three main components in a direct conversion transmitter are the power amplifier, modulator, and the digital-to-analog converter (DAC).

The modulator chosen is the analog devices AD8349. It is a quadrature modulator that is able to operate with an output frequency range from 700 MHz to 2700 MHz. It features a modulation bandwidth from dc to 160 MHz and a noise floor of –156 dBm/Hz. Dual different IQ inputs are provided from the DAC and to improve the noise performance the local oscillator (LO) drive. The output power generated by the modulator is within the range of –2 to +5.1 dBm.

The power amplifier is constructed as a two-stage element: a fixed gain power amplifier and a digitally controlled variable gain amplifier. The power amplifier used is the MGA-83563 from (Avago, Calif, USA) which is a broadband high linearity amplifier. It works in the frequency range of 40 to 3600 MHz and achieves a small signal gain of 20 dB with a noise figure of 4.1 dB. This variable gain amplifier is the Analog Devices ADL5330 which operates from 10 MHz to 3 GHz frequencies, with a gain control range of 60 dB. The combined system can deliver 22 dBm of power in 256 programmable steps.

Digital-to-analog converters are more capable than ADCs for any given technology. For this application it was possible to get a dual-path 16-bit DAC from Maxim (MAX5875) that can support output rates of up to 200 MSps. It features an integrated +1.2 V bandgap reference and control amplifier to ensure high accuracy and low-noise performance. The output rate is adjustable based on the provided clock frequency.

**4.4. USB Communications.** As this is a nonstandard USB application, a customized USB driver and firmware were developed to maximize throughput and deliver sustained performance. As stated, the maximum throughput of the USB link was performance limiting factor in the platform. Even though USB offers 480 Mbps, in practice sustained performance is substantially less. Sustained performance is necessary as gaps in the data flow are unacceptable and excessive buffering will introduce latency effects. A specialist Linux driver was written to ensure suitable performance, and an efficient API library was implemented to provide a robust interface with third party software engines. Figure 4 shows a high-level vision of the interconnection between the elements of the integrated radio platform.

The USB connect was provided through a USB 2.0 Cypress EZ-USB device with an on-board 8051-compatible microcontroller (Figure 6). The function of the microcontroller was to route the data between the general purpose



interface (GPIF) of the USB device and the data converters. Through the use of USB endpoints, it was also possible to implement a control channel for reconfiguring the system. This control plane can be accessed during operation; but to maximize data throughput, it is recommended that it be only used between communication sessions.

The main software elements of our platform were some embedded code running on the USB microcontroller, an optimized Linux USB driver and an API library providing an interface with IRIS. Linux was selected due to its superior real-time performance and access to low-level device drivers. The principal challenges were first to provide high-speed and continuous data transfer without data loss and second to enable the reconfigurability of the hardware devices. High-speed data transfer without data loss was achieved by using optimized techniques in both USB driver and embedded code. Due to their ability to be queued, the USB driver utilizes USB request blocks (URBs) as the data structure for transmitting or receiving information [23, 24]. This queue of URBs guarantees that there will be always information waiting to be processed in the communication channel, which causes maximum usage of bandwidth and a continuous stream of information. Using the bulk transfer communication mode guaranteed delivery of data, solving the data loss problem. With this optimized driver, it was possible to achieve a maximum sustained throughput of 256 Mbps, an improvement over other driver implementations but substantially less than the 480 Mbps peak transport. Finally hardware reconfiguration is obtained through API functions, for example, for configuring the sampling rate, the local oscillator frequency, and the receive chain gain control.

**4.5. Software Radio Framework.** The software radio framework utilized in our system is the IRIS software radio framework. IRIS has been under development at Trinity College Dublin since 1999. It is a highly flexible and highly reconfigurable software radio platform for a general purpose processor running either Windows or Linux.

The IRIS architecture is illustrated in Figure 7 and comprises of DSP components which are configurable through an XML file. Examples for such components are modulators, framers, or filters. Each of the components has a set of parameters and an interface to the control logic, which allow for reuse in different radio configurations. The control logic is a software component designed for a specific radio configuration, that is, it is aware of the full radio chain while the processing components are not. This control logic can subscribe to events triggered by radio components and change radio parameters or reconfigure the radio structure. This enables the IRIS framework to support cognition through this control mechanism.

To design a radio with IRIS, an Extensible Markup Language (XML) configuration file is written that specifies the radio components, their parameters, and connections. Optionally the radio designer can implement a control logic manager for dynamic radio reconfiguration. On start up the XML file is parsed and the run-time engine creates the radio by instantiating and connecting the specified components.

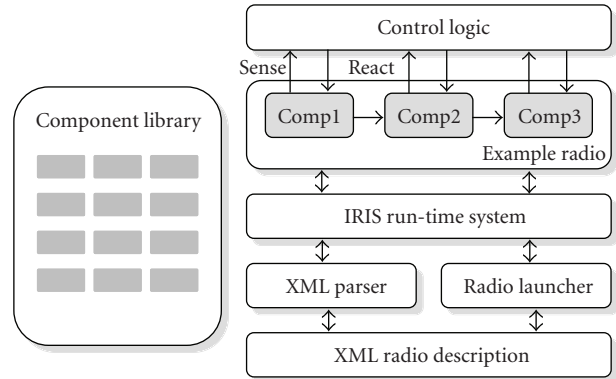


FIGURE 7: IRIS architecture.



FIGURE 8: MARS receive and transmit boards.

The run-time engine then loads the control logic and attaches it to the components. Finally the radio is started, and blocks of data generated by the source component will be processed by each of the components in the radio chain. The control logic can react to events triggered by components, with anything from diagnostic output to a full reconfiguration of the radio.

**4.6. Final Design.** The implementation of the MARS platform was as two separate simplex elements: a receive-only and a transmit-only boards (shown in Figure 8). Duplex operation was avoided due to the limitations of the USB throughput. A version of the baseband board exists that allows for duplex operation but at half the bandwidth. As the MARS platform is part of an ongoing research project into software radio platforms, there have been subsequent improvements on the design which will be detailed later.

## 5. Performance and Use Cases

The MARS platform has been tested under a number of use cases—for example,

- (i) Spectrum sensing.
- (ii) Still image and video transmission.
- (iii) Novel communication schemes.
- (iv) Interoperability testing with the USRP.



To test the proposed SDR platform together with IRIS we successfully transmitted an image [25]. To isolate platform artifacts, a USRP and an MARS platform were used interchangeably as transmitter and receiver. The IRIS software engine has appropriate software interfaces for the two platforms. The IRIS software engine read a bit-map image, framed the data using a simple structure, with appropriate data whitening and error correction encoding. Differential quadrature phase shift keying (DQPSK) was used to modulate the data into four symbols. To limit the spectral footprint of the signal, it is upsampled and filtered with a root raised cosine pulse shaper. The resulting IQ samples were delivered over USB to the radio front-end. At the receiver, the MARS platform demodulates the data and delivers unprocessed IQ samples over USB to the software engine. IRIS then undertakes filtering, clock data recovery, and demodulation. The data is then deframed and reconstructed into the image. In this experiment we used a 1 MSps transfer rate. In this mode of operation we could operate over six times faster, but are limited primarily by the processing performance of the PC or laptop used. The results of this experiment are shown in Figure 9 where the resulting image and constellation diagram are presented. The constellation diagram provides an indication that the error vector magnitude is acceptably small and good communication is possible.

In another example, a video sample was transmitted and received using MARS platforms (Figure 10). A DBPSK modulation scheme was used. The transmitted signal bandwidth was approximately 300 kHz with an IQ sample rate of 2 MSps. This proved acceptable for video transmission but higher throughput could be obtained with higher order modulation schemes. The error vector magnitude suggests that a more dense constellation diagram could be implemented without significant impairment of performance. The limitation on using a higher modulation scheme lies in the software engine and this is likely to improve with time and processing power.

The strength of the MARS platform is in the quality of the RF elements of the circuit. Deliberate effort went into designing a high-quality receive chain in accordance with the requirements of the various standards. Table 2 presents the characteristics of the MARS platform in context to the other SDR testbed platforms. Though more powerful and capable systems exist, the MARS platform should be compared with the USRP for complexity and performance. In that context, it offers a similar level of baseband capacity with superior radio frontend performance. The two are interchangeable and offer users the ability to assess the performance of their software radio schemes independently of a specific hardware implementation and associated artifacts.

## 6. Future Trends

The first generation of available SDR platforms occurred around 2004–2006. Technology has progressed since then and there have been significant improvements in signal processing performance, connectivity, and in the quality of RF components such as mixers and data converters. With

current capabilities it has become possible to implement most narrowband communication schemes (e.g., GSM) though not without significant effort and expertise. However, in recent years there has been a movement toward wider band solutions such as wCDMA and OFDM technologies. The effect is that SDR platforms are challenged by increasing bandwidths, reducing minimum signal strengths, and reducing maximum allowable error vector magnitudes. Application specific SDR platforms can be constructed with a combination of available technologies. General purpose experimental SDR platforms still face challenges and will be driven by three trends:

- (i) Increased capacity platform interfaces.
- (ii) An increasingly diverse range of processors.
- (iii) Increased on-board processing capability.

The USRP2 from Mark Ettus is the first of the next generation of SDR platforms, and these trends are visible in the new design: significant on-board FPGA and a gigabit Ethernet connection.

**6.1. Increased Platform Interfaces.** The first generation of SDR platforms either used Ethernet or USB to provide connectivity to computers and other users. Ethernet can now commonly offer 1 Gbps, but existing SDR platforms used only 10/100 Mbps links which in practice delivered less than half that when routing overheads are considered. USB 2.0 offered a superior performance with 480 Mbps and a maximum sustained rate of 256 Mbps. In practice, to deliver 25 MHz of bandwidth to a duplex transceiver, a minimum of 2.4 Gbps would be required and a more conservative estimate would suggest 4.8 Gbps [26]. This problem is exacerbated when considering multiple element systems such as in the MIMO variant of 802.11(n). This problem can be partially solved by improving the interface communication speeds to the platform. Preserving compatibility with generic computers, multigigabit Ethernet and PCIexpress are likely to be seen in future platforms. PCIexpress was used in the KUAR platform as an internal protocol; but with PCIexpress in most computers, it has become feasible to use it as a communication interface. PCIexpress is used in graphics cards and is optimized for streaming data. In version 2.0, it offers 4000 Mbps in each direction per lane. This is sufficient for encoding 25 MHz of bandwidth, however it is possible to combine multiple PCIexpress lanes and increase performance. Most computers have at least two lanes as an expansion port, and this is likely to increase in the future. If you access the graphics bus, up to 32 lanes are available, providing 128 Gbps of bidirectional data. Alternatively, newer forms of gigabit Ethernet offer up to 100 Gbps which is also sufficient for most applications. One common trend in all these schemes is the move to optical connections. This is a trend being encouraged by developments in the mobile telephony base station industry, where fibre-optic links deliver electrical isolation, ability to place RF elements away from the processing unit, and provide an upgradeable communications infrastructure. One

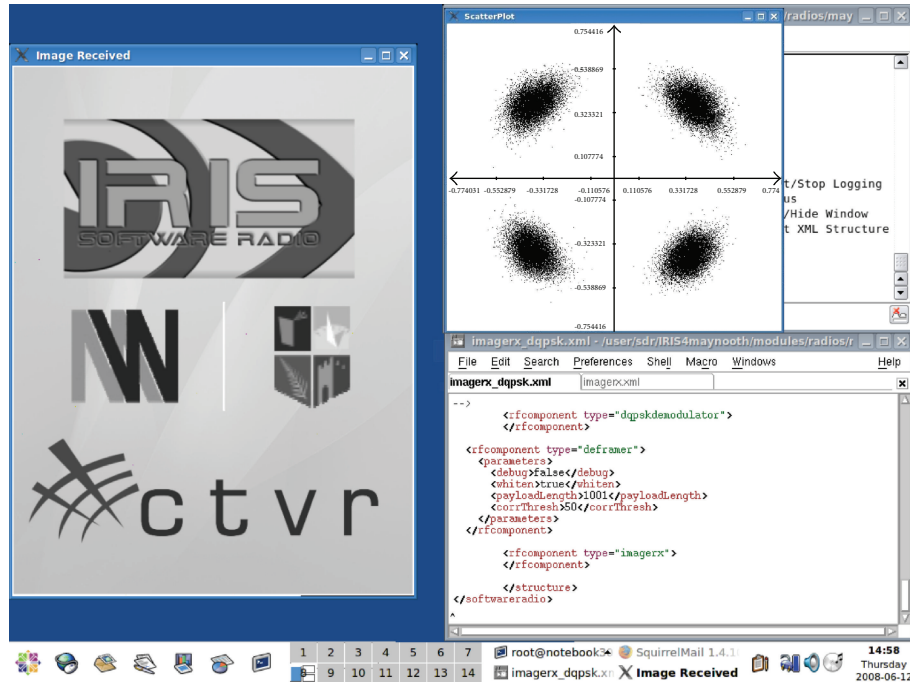


FIGURE 9: Transmitted image with constellation diagram.

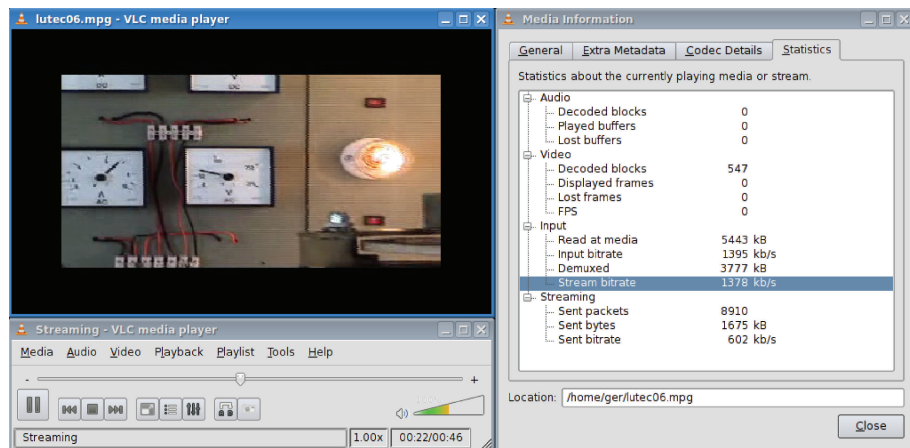


FIGURE 10: Example of video being received using the SDR platform with performance statistics.

implication of the increasing communications capacity is the requirement for increased on-board processing capacity.

**6.2. Increased on-Board Processing Capability.** The concept of placing the majority of the signal processing off-board on a computer was a valid concept that derived from the software engineering/computer science researchers who were active from the earliest days. This concept is exemplified by the commercial products developed by (Vanu, Inc. Mass, USA) and by the GNU radio architecture. This approach faces two challenges: modern communication schemes expect a very low-latency response, particularly in the initial handshaking events—which is very difficult to do when passing the data over a link to a separate processor; secondly general purpose

processors are not optimally suited for many of the computational intensive aspects of a communications scheme, for example, inverse-FFTs. Specialists DSPs and FPGAs offer superior performance for many of these functions and these can be used to reduce latency by processing signals closer to the antenna prior to transport to a processing unit. This approach partitions the signal processing optimally to the available processor architectures and has the benefit of reducing the quantity of data needed to be transported and maximizing the system capacity. One extreme is to place one or more processors on board and allow the board to be functionally independent of any external source (e.g., the KUAR and BEE2 platforms). However, general purpose processors still offer superior flexibility and ease of use when developing new systems, but with higher speed connections

TABLE 2: Comparison of available SDR platforms.

	MARS	MARS3	KUAR	USRP	USRP2	BEE2	NICT
Year of release	2007	2009	2005	2005	2008	2007	2005
RF bandwidth (MHz) <sup>(1)</sup>	70	25	30	5	25	25	25
Frequency range (GHz) <sup>(2)</sup>	1.7–2.5	1.7–2.5	5.25–5.85	2.3–2.9 <sup>(4)</sup>	2.3–2.9 <sup>(4)</sup>	Fixed (2.45)	1.9–2.4 5.0–5.3
Processing partition	Off-board	Mixed	On-board	Off-board	Mixed	On-board	On-board
Processor architecture	GPP	FPGA	GPP FPGA	GPP	GPP FPGA	FPGA	GPP FPGA
Connectivity	USB	PCIexpress GigEthernet	USB Ethernet	USB	GigEthernet	USB Ethernet	USB Ethernet
No. of antennas or RF paths	2	16	2	4	2 <sup>(3)</sup>	16	2
Standards aware (RF)	yes	yes	no	no	no	no	yes
Standards aware (baseband)	yes	yes	no	yes	yes	yes	yes
Strengths	Low cost	Large bandwidth		GNU radio integration	Large bandwidth	Processing power	Standard compliance
Weaknesses	Limited bandwidth		Frequency range	Limited bandwidth	Complexity		Limited availability

<sup>(1)</sup> Assuming no baseband or connectivity restrictions.

<sup>(2)</sup> Within a single RF board.

<sup>(3)</sup> Extendable through linking multiple platforms.

<sup>(4)</sup> Wide selection of frequency ranges available.

and wider bandwidths, partitioning of processing functions between the computer, and the SDR hardware platform appears unavoidable.

**6.3. Diverse Range of Processors.** Typical implementations of software-defined radio (SDR) systems include a general-purpose processor (GPP), a digital signal processor (DSP), or an FPGA, though dedicated DSP chips are being challenged by FPGAs with embedded DSP cores [27]. FPGA-based systems can deliver the performance but at the cost of increased design complexity. General purpose processors are less effective at physical layer processing but excel at the higher layers and are more accessible to the general software designer. Neither is optimal and this has resulted in a broader range of processor types being developed and used for software-radio applications. These can be broadly, and not exclusively, categorized as complex multicore systems; specialist floating-point calculators such as found in graphics chips.

Multicore systems are common with many computer processors containing multiple cores. These are, however, multiple versions of the same core. A heterogeneous multicore system could contain a mixture of embedded FPGAs, DSPs or general purpose processors, with functions being allocated to match the strengths of a specific core. Examples of these devices include recent generations of FPGAs are now including dedicated DSP slices and complete processor cores, but are programmed using traditional FPGA design tools. Another example is the Sandbridge Sandblaster processor

which contains multiple DSP cores and an ARM9 processor, and is treated as a DSP device [28]. Future SDR platforms will likely take advantage of this trend to deliver increased capacity, with the likelihood of the increasingly complex FPGAs being utilized first due to their relative maturity.

The other interesting development is the use of graphics chips to deliver the floating point processing power needed for wideband physical layer processing. Graphics chips are dedicated floating point processors which are optimized to deliver sustained performance. As part of a commodity market, it is difficult to match their processing power per cost ratio and they come with a well-developed software development environment. One of the most powerful devices is IBM CELL processor as used in the Sony Playstation3. The CELL processor is another multicore device and is designed to excel at parallel processing. It has a theoretical maximum performance of 204.8 GFLOPS (single precision)—sufficient for any software-defined radio [29]. There is already an initiative to port GNU radio to the cell processor to avail of this capability [30].

## 7. Future Development

The development of the MARS platform was an exploration of the challenges in implementing a base station-orientated reconfigurable platform. As such it has provided us with many insights in how the technical issues are subtly different than those experienced in handheld designs. As part of an ongoing research project, we are currently working on the

next generation of the MARS platform. The MARS platform did not include any on-board processing power, the next platform, MARS2 is in testing and will include a Xilinx Spartan3 device to enable local processing. Though this will still use a USB connection, it will allow us to avail of the greater capabilities of the RF boards. Most of our current activity is focused on the third generation of the MARS platform. This platform is focused on supporting a wider bandwidth and to have substantial localized processing. The key characteristics of this design are as follows

- (i) A PCIexpress connection to a computer, providing up to 4 Gbps connectivity.
- (ii) A baseband processor board with one or more Virtex4 processors, capable of supporting 8 transmit and receive paths (16 in total).
- (iii) Fibre-optic CPRI/OBSAI [31, 32] links for distribution of data to remote RF boards
- (iv) Remote RF boards that are enhancements of existing MARS boards with fibre-optic links, gigabit Ethernet and USB as back.
- (v) Flexible RF performance supporting 25 MHz of bandwidth.

This platform is significantly more complex than before but it is designed to be modular so that the superior RF frontends can be used in isolation or as part of a network of links boards. Though the bandwidths are substantially higher, the platform will remain compatible with the IRiS and GNUradio software frameworks. First prototypes of the new platform are expected in the summer of 2009.

## 8. Summary

Though software-defined radio offers many compelling benefits to radio system designers, there remains many open questions on how to effectively implement and manage flexibility in a wireless system. Software radio platforms and testbeds offer researchers and developers the ability to develop their applications in advance of designing customized hardware. In recent years there have been substantial improvements in technology, and low-cost platforms are now possible though few are generally available.

In this paper, we presented a brief overview of the state-of-the art of SDR platforms and the future technology trends in this area. We also presented an experimental platform developed at the National University of Ireland, Maynooth. This platform is currently being used by our collaborators and we wish to share this platform with new collaborators to develop a broader community of users and diverse applications.

## Acknowledgment

This material is based upon work supported by Science Foundation Ireland under Grant no. 03/CE3/I405 as part of the Centre for Telecommunications Value-Chain Research (CTVR) at the National University of Ireland, Maynooth.

## References

- [1] "Software Defined Radio (SDR) Forum, Technical Definitions," <http://www.sdrforum.org>.
- [2] J. Mitola, "Software radios—survey, critical evaluation and future directions," in *Proceedings of IEEE National Telesystems Conference (NTC '92)*, vol. 13, pp. 15–23, Washington, DC, USA, May 1992.
- [3] W. H. W. Tuttlebee, "Software radio technology: a European perspective," *IEEE Communications Magazine*, vol. 37, no. 2, pp. 118–123, 1999.
- [4] M. J. Marcus, "WAPECS—Europe moves toward technical flexibility for wireless systems," *IEEE Wireless Communications*, vol. 15, no. 1, pp. 4–5, 2008.
- [5] U. Ramacher, "Software-defined radio prospects for multi-standard mobile phones," *Computer*, vol. 40, no. 10, pp. 62–69, 2007.
- [6] G. Hueber, L. Maurer, G. Strasser, K. Chabrak, R. Stuhlberger, and R. Hagelauer, "SDR compliant multi-mode digital-front-end design concepts for cellular terminals," in *Proceedings of the 4th WSEAS International Conference on Electronics, Hardware, Wireless and Optical Communications (EHAC '05)*, pp. 1–5, Salzburg, Austria, February 2005.
- [7] D. R. Oldham and M. C. Scardelletti, "JTRS/SCA and custom/SDR waveform comparison," in *Proceedings of IEEE Military Communications Conference (MILCOM '07)*, pp. 1–5, Orlando, Fla, USA, October 2007.
- [8] B. Razavi, *RF Microelectronics*, Prentice Hall, Upper Saddle River, NJ, USA, 1998.
- [9] E. Blossom, "GNU radio: tools for exploring the radio frequency spectrum," *Linux Journal*, no. 122, June 2004.
- [10] "Universal Software Radio Platform, Ettus Research," <http://www.ettus.com>.
- [11] G. J. Minden, J. B. Evans, L. Searl, et al., "KUAR: a flexible software-defined radio development platform," in *Proceedings of the 2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN '07)*, pp. 428–439, Dublin, Ireland, April 2007.
- [12] H. Harada, "Software defined radio prototype toward cognitive radio communication systems," in *Proceedings of the 1st IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN '05)*, pp. 539–547, Baltimore, Md, USA, November 2005.
- [13] S. M. Mishra, D. Cabric, C. Chang, et al., "A real time cognitive radio testbed for physical and link layer experiments," in *Proceedings of the 1st IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN '05)*, vol. 1, pp. 562–567, Baltimore, Md, USA, November 2005.
- [14] L. Ruiz, G. Baldwin, and R. Farrell, "A platform for the development of software defined radio," in *Proceedings of the 18th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '07)*, pp. 1–5, Athens, Greece, September 2007.
- [15] P. Mackenzie, K. E. Nolan, L. Doyle, and D. O'Mahony, "An architecture for the development of software radios on general purpose processors," in *Proceedings of the Irish Signals and Systems Conference (ISSC '02)*, pp. 275–280, Cork, Ireland, June 2002.
- [16] F. Adachi, H. Wakana, H. Morikawa, et al., "Network and access technologies for new generation mobile communications—overview of National R&D Project in NICT," *Wireless Communications and Mobile Computing*, vol. 7, no. 8, pp. 937–950, 2007.



- [17] A. Pouttu, H. Romppainen, V. Tapio, T. Bräysy, P. Leppänen, and T. Tuukkanen, "Finnish software radio programme and demonstrator," in *Proceedings of IEEE Military Communications Conference (MILCOM '04)*, vol. 3, pp. 1371–1376, Monterey, Calif, USA, October 2004.
- [18] W. Schacherbauer, A. Springer, T. Ostertag, C. C. W. Ruppel, and R. Weigel, "A flexible multiband frontend for software radios using high if and active interference cancellation," in *Proceedings of the International Microwave Symposium Digest (MWSYM '01)*, vol. 2, pp. 1085–1088, Phoenix, Ariz, USA, May 2001.
- [19] R. J. DeGroot, D. P. Gurney, K. Hutchinson, et al., "A cognitive-enabled experimental system," in *Proceedings of the 1st IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN '05)*, pp. 556–561, Baltimore, Md, USA, November 2005.
- [20] A. Ibing, D. Kühling, M. Kuszak, C. V. Helmolt, and V. Jungnickel, "Flexible demonstrator platform for cooperative joint transmission and detection in next generation wireless MIMO-OFDM networks," in *Proceedings of the 4th International Conference on Testbeds and Research Infrastructures for the Development of Networks & Communities*, pp. 1–6, Innsbruck, Austria, March 2008.
- [21] A. Polydoros, J. Rautio, G. Razzano, et al., "WIND-FLEX: developing a novel testbed for exploring flexible radio concepts in an indoor environment," *IEEE Communications Magazine*, vol. 41, no. 7, pp. 116–122, 2003.
- [22] D. Naughton, G. Baldwin, and R. Farrell, "Performance requirements for analog-to-digital converters in wideband reconfigurable radios," in *VLSI Circuits and Systems II*, vol. 5837 of *Proceedings of SPIE*, pp. 582–589, Seville, Spain, May 2005.
- [23] J. Corbet, A. Rubini, and G. Kroah-Hartman, *Linux Device Drivers*, O'Reilly, Cambridge, Mass, USA, 3rd edition, 2005.
- [24] G. Kroah-Hartman, *Linux Kernel in a Nutshell*, O'Reilly, Sebastopol, Calif, USA, 2006.
- [25] M. Sánchez, J. Lotze, G. Corley, and R. Farrell, "Experiences in the co-design of software and hardware elements in an SDR platform," in *Technical Conference and Product Exhibition (SDR '08)*, Washington, DC, USA, October 2008.
- [26] A. Wyglinski, M. Nekovee, and T. Hou, *Cognitive Radio Communications and Networks: Principles and Practice*, Elsevier Press, New York, NY, USA, 2009.
- [27] R. Baines and D. Pulley, "A total cost approach to evaluating different reconfigurable architectures for baseband processing in wireless receivers," *IEEE Communications Magazine*, vol. 41, no. 1, pp. 105–113, 2003.
- [28] M. Schulte, J. Glossner, S. Mamidi, M. Moudgill, and S. Vassiliadis, "A low-power multithreaded processor for baseband communication systems," in *Proceedings of the 3rd and 4th International Workshops on Computer Systems: Architectures, Modeling, and Simulation (SAMOS '04)*, vol. 3133 of *Lecture Notes in Computer Science*, pp. 393–402, Samos, Greece, July 2004.
- [29] T. Chen, R. Raghavan, J. Dale, and E. Iwata, "Cell broadband engine architecture and its first implementation: a performance view," *IBM Journal of Research and Development*, vol. 51, no. 2, pp. 559–572, 2007.
- [30] F. Ge, Q. Chen, Y. Wang, C. W. Bostian, T. W. Rondeau, and B. Le, "Cognitive radio: from spectrum sharing to adaptive learning and reconfiguration," in *Proceedings of IEEE Aerospace Conference*, pp. 1–10, Big Sky, Mont, USA, March 2008.
- [31] "Common Public Radio Interface (CPRI) Standard," <http://www.cpri.info/>.
- [32] "Open Base Station Architecture Initiative," <http://www.obsai.org/>.



## Research Article

# Exploiting Redundancy in an OFDM SDR Receiver

**Tomas Palenik and Peter Farkas**

*Department of Telecommunications, Slovak University of Technology, Ilkovicova 3, 812 19 Bratislava, Slovakia*

Correspondence should be addressed to Tomas Palenik, palenikt@ktl.elf.stuba.sk

Received 1 September 2008; Accepted 10 February 2009

Recommended by Daniel Iancu

Common OFDM system contains redundancy necessary to mitigate interblock interference and allows computationally effective single-tap frequency domain equalization in receiver. Assuming the system implements an outer error correcting code and channel state information is available in the receiver, we show that it is possible to understand the cyclic prefix insertion as a weak inner ECC encoding and exploit the introduced redundancy to slightly improve error performance of such a system. In this paper, an easy way to implement modification to an existing SDR OFDM receiver is presented. This modification enables the utilization of prefix redundancy, while preserving full compatibility with existing OFDM-based communication standards.

Copyright © 2009 T. Palenik and P. Farkas. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

## 1. Introduction

Thanks to their flexibility, SDR platforms enable relatively easy modifications to the communication system subblocks, which could bring advantages or gains to existing system. It is even possible sometimes to exploit already standardized protocols in an unexpected new way. In this paper, this future will be illustrated on an example of the standard OFDM transmission technique.

In current standards, for example, IEEE 802.16e, the OFDM system uses a cyclic prefix (CP) to mitigate the effects of channel impairment. For each transmitted OFDM symbol, the length of the prefix is a fraction of useful symbol length. The main purpose of using cyclic prefix is protection against intersymbol interference (ISI), or more precisely interblock interference (IBI), in connection with simple equalization in frequency domain. Most of current receivers throw the cyclic prefix away or use it only for channel estimation. In this paper, we present a method for exploiting the redundancy introduced by the cyclic prefix, by means of decoding two serially concatenated codes. To be more precise, we understand the prefix insertion as an inner repetition coding, where only a part of the samples is repeated. With code rate  $R = 8/9$  (assuming the repetition of one eighth of the samples as in [1]), this code is very weak, but in cooperation with a powerful outer code (which is always present in a practical system), it could bring an error

performance gain. Of course the received copy of the prefix data is corrupted by IBI and has to be processed first.

In the next section, we give a short description of current OFDM systems along with a brief overview of recent methods for exploiting of cyclic prefix. The third section refreshes the principle of concatenated coding and depicts the possible OFDM error performance improvement using a simulation with real-life settings. The fourth section is devoted to description of the received prefix processing, which is necessary for successful extraction of repetition data. The last section describes the whole modified OFDM receiver and then presents simulation results showing the actual error performance improvement in a multipath environment.

## 2. Current OFDM Systems and Modifications

**2.1. OFDM System Overview.** Figure 1 shows a simplified standard OFDM system model (defined in [2]). Each of the depicted functional blocks is implemented in software, using a high-level programming language such as ANSI C [3], as a separate function or module, possibly taking advantage of underlying platform specific hardware acceleration. Since this paper primarily focuses on software processing of floating or fixed point vectors, the analog and hybrid blocks, such as amplifiers and A/D converters, are intentionally omitted in Figure 1.

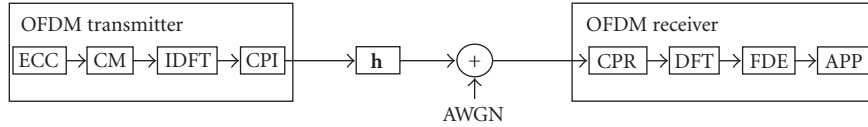


FIGURE 1: Standard OFDM system model: ECC, outer error correcting code encoder; CM, constellation mappings; (I) DFT, (inverse) discrete Fourier transform;  $h$ , channel impulse response; CPR, cyclic prefix removal; FDE, frequency domain equalizer; APP, outer code posterior probability decoder.

OFDM processing begins with blocks of ECC encoded binary data that are first digitally modulated in frequency domain (CM Block), then transformed to time domain by IDFT. A cyclic prefix is attached in the CPI block. The OFDM symbol (or time domain block) consisting of many (1024 and more) samples travels through multipath environment, which can be modeled by a convolution with channel impulse response  $h$ . The prolonging of the block caused by the channel convolution is source of ISI and IBI. Furthermore, AWGN noise is superimposed onto the signal. The receiver first selects a major subset of samples of the received OFDM symbol in the CPR block; the rest of the block samples is discarded because it is corrupted by IBI. The selected part of the samples is then transformed to frequency domain, where it can be easily equalized by an efficient single-tap FDE equalizer. Easy equalization is a consequence of the cyclic prefix insertion and removal. Equalization in time domain would require a much more complicated computation of channel convolution inversion. The quality of the equalization process, however, depends on the precision of the CSI estimate. The output of equalization is a noisy estimate of original signal space mapped data block. After symbol detection, this block is finally transformed to a block of log-likelihood ratio (LLR) values for further soft-input error correcting decoding.

**2.2. Advanced Cyclic Prefix Usage.** The channel estimation and symbol detection blocks are omitted from Figure 1 as this paper focuses on the decoding of the code cascade. However, several conceptually different approaches for increasing throughput and/or exploiting the cyclic prefix of a coded OFDM system already exist.

The first one is reducing of the CP size to less than the channel delay spread and overcoming the resulting IBI by modifying the iterative ECC decoder to work over two consecutive OFDM symbols so that it is able to fix errors resulting from IBI that occurs in case of insufficient CP size [4].

A different approach is exploiting of the CP (of size greater than the channel delay spread) to improve the channel estimation [5–8] or symbol detection [9]. However, the methods described in [6, 8] practically apply only to xDSL environment and are not suitable for variable channel conditions in a wireless transmission, while the redundant CP samples in [9] are used only for reducing the noise variance for the replicated samples using a time domain max ratio combining (MRC) algorithm in single carrier system—not OFDM.



FIGURE 2: Serial concatenated encoding.

The residual intersymbol interference cancellation (RISIC) is presented in [5]. The principle is to cancel the residual IBI resulting from an insufficient CP by iterating channel estimation, cyclicity restoration, and soft output decoding. The algorithm is defined purely for a setting where a space-time EC code is present in a MIMO-OFDM system.

A more general method of turbo frequency domain equalization (turbo FDE) is presented in [7]. Here a soft elementary signal estimation (ESE) block works together with a soft-input soft-output (SISO) ECC decoder in an iterative manner so that the estimates of channel symbols are iteratively improved by the results of the decoder. This method uses the CP insertion and removal only to ensure the circulant property of the channel matrix. The iterative application of a complex APP decoding algorithm results in a great increase in computational complexity.

Finally, the third concept is based on the observation that cyclic prefix size defined in *communication standard* is designed for the worst-case scenario therefore in case that the channel delay spread is shorter than the prefix duration, the prefix information can be used for CE without extensive postprocessing [10]; the drawback is limited usage of this method together with modern standards, such as IEEE 802.16e where the size of the prefix varies according to the channel propagation conditions.

### 3. Concatenation of Codes

**3.1. OFDM is a Concatenated Code.** The principle of serial concatenation of codes is well known. As shown in Figure 2, the transmitter simply feeds the output of one encoder to the input of the other through a pseudorandom or rectangular interleaver.

The situation in the receiver is more complicated; usually two SISO decoders cooperate in an iterative manner, exchanging extrinsic information as described in [11]. After a number of iterations, final hard decision is made. This decoding scheme is well known and is also used in turbo code decoding application.

In today's systems, OFDM is always used along with a powerful error correcting code such as turbo or LDPC code as shown in Figure 1. If we understand the cyclic

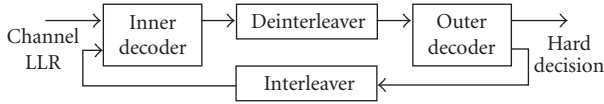


FIGURE 3: Iterative decoding of serially concatenated codes.



FIGURE 4: Modified OFDM decoder principle.

prefix insertion as an inner code and the IDFT as an interleaver, then the coded OFDM transmitter is a serially concatenated encoder system, and the decoder can be redesigned to iterative form. Because the inner code is very weak (partial repetition) and the decoding of the outer code is computationally intensive (It is an iterative process itself.) a simplified noniterative soft-output scheme is suggested.

In Figure 4, only the forward branch of the decoder in Figure 3 is performed, so the weak inner repetition code with very simple decoding is used only to improve the log-likelihood values of samples from the channel, entering the powerful outer code decoder. The first logical step in the analysis of concatenated OFDM decoding is to omit the negative effect of IBI, ISI, and time frequency domain switches. This simplification leads to a very simple model, with transmitter and receiver shown in Figures 2 and 4. As mentioned before, the SISO decoding of the repetition code is very simple. The extrinsic information for any bit is just the copied channel LLR of its repetition [12]. Therefore, if a second copy of part of the bits is available, the first stage (the decoding of the partial repetition code) is done by a simple addition of LLR values. After deinterleaving, these fortified LLRs enter the unmodified outer code decoder.

**3.2. Empiric Error Performance upper Bound.** Before actual prefix redundancy extraction efforts, simplified simulation experiments were done, primarily with the goal to give us a proof of concept. The first round of simulations used a simplified OFDM system model. In the simulations, we used the inner partial repetition code of rate  $R = 8/9$  which corresponds to one of the WiMax defined cyclic prefix sizes for OFDM [1] and convolutional turbo code of rate  $R = 1/3$  such as defined in UMTS standards [13]. We also used a pseudorandom interleaver used in UMTS instead of IDFT. The simulation results are shown in Figure 5. Three curves depict the BER after the 1st, 3rd, and 7th iteration. It is clear that the system with repetition decoder placed in front of the turbo decoder achieves the same reliability at approximately 0.25 dB lower  $E_b/N_0$  than the turbo system alone.

The resulting 0.25 dB improvement can be interpreted as a rough estimate or an upper bound to the actual improvement that can be achieved in real systems. The significant difference between the simplified model and the real system is corruption of the second copy of data used

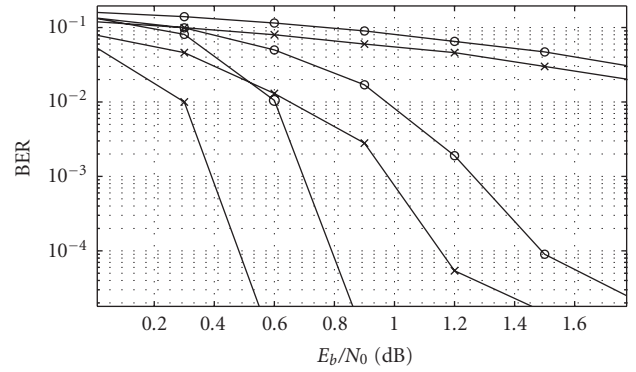


FIGURE 5: BER after 1st, 3rd, and 7th decoder iterations for UMTS turbo code with (x-marker) and without (o-marker) using the prefix redundancy.

in repetition decoder. This corruption is caused by IBI and cannot be fully remedied. Two differently successful solutions addressing this problem are described in the following sections.

#### 4. Cyclic Prefix in OFDM

The following section consists of three parts. First, the principle of prefix insertion (CPI) and removal (CPR) for the purpose of simple frequency domain equalization in context of OFDM multipath-environment transmission is reviewed. Second, a more formal matrix-based description of channel and CPI/CPR processes is presented. Finally, the process of extraction of redundant information from cyclic prefix (CP) is described, based on the formal matrix representation.

**4.1. Frequency Domain Equalization.** The propagation of a signal through a multipath channel with ISI is usually described by convolution with the channel impulse response  $\mathbf{h}$  (in this paper a discrete-time version is assumed). In a block-oriented system, such as OFDM, the discrete-time convolution can be written in matrix form

$$\mathbf{r} = \mathbf{H} \times \mathbf{t}, \quad (1)$$

where  $\mathbf{t}$  is a vector of time samples, produced by the transmitter,  $\mathbf{H}$  is a convolution matrix consisting of channel impulse response values (an example  $\mathbf{H}$  is shown in Figure 7), and  $\mathbf{r}$  is channel output/received sequence. The process of convolution of transmitted data block with channel impulse response prolongs the block from transmitted  $N$  samples to received  $N + \nu - 1$  samples, where  $\nu$  is the length of  $\mathbf{h}_{(n)}$ . If there is no guard interval between two consecutively transmitted blocks, or if the guard interval is shorter than  $\nu - 1$ , IBI occurs on the boundaries of the blocks.

One possible way of coping with IBI is to send an all-zero guard prefix. Another way, used in OFDM, is to use a cyclic prefix—part of the samples from the end of the transmitted block is copied and prepended before the beginning of the block. In the receiver, only the appropriate subblock of the received sequence is selected, redundant samples of the prefix

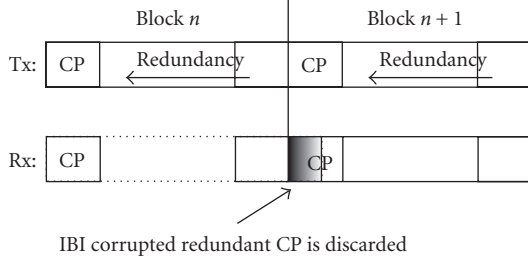


FIGURE 6: Prolonging of blocks in multipath channel results in interblock interference; currently the redundant CP samples are all discarded.

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.9 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.4 & 0.9 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.4 & 0.9 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.4 & 0.9 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.4 & 0.9 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.4 & 0.9 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.4 & 0.9 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.4 & 0.9 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.4 \end{bmatrix}$$

FIGURE 7: Structure of channel convolution matrix  $\mathbf{H}$  for  $\mathbf{h}_{(n)} = \{1, 0.9, 0.4\}$  and block size of 8 samples. The received block is prolonged to 10 samples. For purpose of this paper, the matrix can be divided as indicated into 16 submatrices;  $\mathbf{H}_{11}$  in the upper left corner to  $\mathbf{H}_{44}$  in the bottom right corner. The division to submatrices is dictated by the CP size, here assuming 2, making the payload vector size 6.

are discarded. The motivation for CP insertion and removal is that these operations allow us to understand the channel convolution matrix as a circulant matrix. More precisely a submatrix  $\mathbf{H}_c$  can be found in  $\mathbf{H}$ , so that if the transmitted vector  $\mathbf{t}$  with redundant CP is multiplied by  $\mathbf{H}_c$ , the resulting received vector  $\mathbf{r}_D$  is the same as if the transmitted vector  $\mathbf{t}$  with no CP inserted was multiplied by a circulant matrix  $\mathbf{H}_{\text{circ}}$  (see Figure 8). For a circulant matrix  $\mathbf{H}_{\text{circ}}$ , the following equation

$$\mathbf{D}_h = \mathbf{F} \times \mathbf{H}_{\text{circ}} \times \mathbf{F}^{-1} \quad (2)$$

means that such a matrix, when multiplied by the Fourier transform matrices (which represent the IDFT in transmitter and DFT in receiver), results in a diagonal matrix with nonzero elements only on the main diagonal [14]. The diagonal elements are also equal to the channel frequency response samples. Therefore, the equalization is simply performed by scalar multiplication of each element of the received block with a reciprocal value of channel frequency response (assumed to be known).

**4.2. Channel Convolution Matrix Circularization.** The goal of prefix extraction is to process the currently discarded

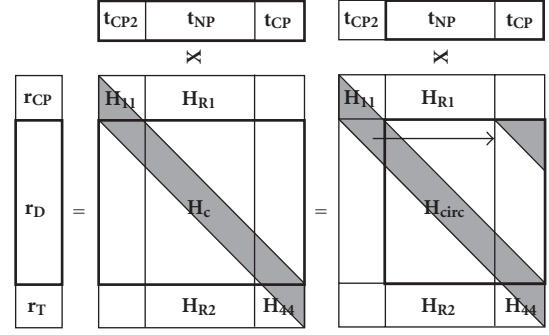


FIGURE 8: Theoretical background for equalization: selecting the correct submatrix  $\mathbf{H}_c$  of channel convolution matrix  $\mathbf{H}$ . The circulant property is a result of cyclic prefix insertion in transmitter and proper subblock selection in the receiver. Vector  $\mathbf{t}$  is transmitter output and  $\mathbf{r}$  receiver input. The bold framing indicates subblock selection. The gray area represents nonzero entries.

received CP subblock to obtain a second copy of the data bits, more specifically a second set of channel LLR values for the data bits, in order to use these values in a repetition decoder and in that way fortify the successive error correcting soft-input decoder.

To completely understand the process, a more formal description of transmission based on (1) is needed. The transmitted vector  $\mathbf{t}$  containing the cyclic prefix can be divided into three subblocks  $\mathbf{t} = \{\mathbf{t}_{\text{CP2}} \parallel \mathbf{t}_{\text{NP}} \parallel \mathbf{t}_{\text{CP}}\}$ , where  $\mathbf{t}_{\text{CP2}}$  is the cyclic prefix with value equal to  $\mathbf{t}_{\text{CP}}$  (repeated samples from the end of the block), and  $\mathbf{t}_{\text{NP}}$  is the data part that is not repeated (“ $\parallel$ ” denotes vector concatenation operator). The payload is the vector  $\mathbf{t}_D = \{\mathbf{t}_{\text{NP}} \parallel \mathbf{t}_{\text{CP}}\}$  while the first  $\mathbf{t}_{\text{CP2}}$  vector is redundant. The received vector can also be divided to subblocks  $\mathbf{r} = \{\mathbf{r}_{\text{CP}} \parallel \mathbf{r}_D \parallel \mathbf{r}_T\}$ , where the  $\mathbf{r}_{\text{CP}}$  part is discarded by design because it is corrupted by IBI—for block number  $n$ , its samples sum up with the  $\mathbf{r}_T$  samples of block  $n - 1$ . Also,  $\mathbf{r}_T$  is the tail subblock—the result of the convolutional prolonging of block in the channel. For the block number  $n$ , it corrupts the samples at the beginning of the block  $n + 1$ . The useful data is the  $\mathbf{r}_D$  part, and in a standard OFDM receiver implementation it is the only subvector further processed. As shown in Figure 8, (1) can be expressed in terms of sub vectors of  $\mathbf{r}$  and  $\mathbf{t}$ ; these subvectors define a partitioning of the convolution matrix  $\mathbf{H}$  dividing it to 3 submatrices  $\mathbf{H}_{R1}$ ,  $\mathbf{H}_c$ , and  $\mathbf{H}_{R2}$ , which can be further partitioned to smaller matrices (mainly for the purpose of finding small, nonzero, and possibly square matrices). (In Figure 8 a partitioning dividing the  $\mathbf{H}$  matrix to  $3 \times 3$  submatrices is shown. However, the lowest rightmost matrix is labeled  $\mathbf{H}_{44}$ . This is intentional and enables compatibility of the labeling with a slightly different partitioning necessary for the prefix extraction defined below. (1) is reformulated as follows:

$$\begin{aligned} \mathbf{r}_{\text{CP}} &= \mathbf{H}_{R1} \times \mathbf{t} = \mathbf{H}_{11} \times \mathbf{t}_{\text{CP2}} = \mathbf{H}_{11} \times \mathbf{t}_{\text{CP}}, \\ \mathbf{r}_D &= \mathbf{H}_c \times \mathbf{t} = \mathbf{H}_{\text{circ}} \times \mathbf{t}_D, \\ \mathbf{r}_T &= \mathbf{H}_{R2} \times \mathbf{t} = \mathbf{H}_{44} \times \mathbf{t}_{\text{CP}}. \end{aligned} \quad (3)$$



**4.3. Prefix Extraction.** Conforming to Figure 8, the interesting information the receiver needs to extract is the redundant information in the transmitted cyclic prefix  $\mathbf{t}_{CP2}$  contained in the received vector  $\mathbf{r}_{CP}$ . So far the redundant information in the received vector  $\mathbf{r}_{CP}$  is discarded because what receiver actually observes is not the vector  $\mathbf{r}_{CP}$  but a combination of samples of two consecutive OFDM symbols (indexed by  $n$ ):

$$\begin{aligned}\mathbf{r}_O &= \mathbf{r}_{CP(n)} + \mathbf{r}_{T(n-1)} \\ &= \mathbf{H}_{11(n)} \times \mathbf{t}_{CP2(n)} + \mathbf{H}_{44(n-1)} \times \mathbf{t}_{CP(n-1)}.\end{aligned}\quad (4)$$

However, if matrices  $\mathbf{H}_{11}$  and  $\mathbf{H}_{44}$  are known, which is assumed true, it is possible to extract the information required using an additive correction and matrix inversion:

$$\mathbf{r}_{cor1(n-1)} = \mathbf{H}_{44(n-1)} \times \mathbf{t}_{CP(n-1)}, \quad (5)$$

$$\mathbf{t}_{CP2(n)} = (\mathbf{H}_{11}^{-1}) \times (\mathbf{r}_O - \mathbf{r}_{cor1(n-1)}). \quad (6)$$

However,  $\mathbf{t}_{CP(n-1)}$  is a reconstruction of transmitter output samples for the block  $n-1$ . It is based on the decoded bits from previous OFDM symbol.

The reconstruction of the transmitter output in receiver is straightforward (as shown in Figure 10). The decoded bits are encoded again, mapped to signal space constellation samples, and transformed to time domain. As most of the iterative decoding algorithms can produce the codeword, along with the decoded bits, the encoding process can be omitted. All other operations necessary for reconstruction are not computationally intensive.

The first obstacle here is the fact that the time domain samples  $\mathbf{t}_{CP2}$  are not very useful for repetition decoding. LLR values have to be created for bits contained in these samples and this computation must be done in frequency domain (similar to standard OFDM operation, see Figure 1). Only such values can be used in repetition decoding.

This can be done by concatenation  $\mathbf{t}_D'' = \{\mathbf{t}_{NP} \parallel \mathbf{t}_{CP2}\}$ , creating a 2nd copy of time block  $\mathbf{t}_D$ , and transformation to frequency domain again. The redundant information contained in a subset of samples  $\mathbf{t}_{CP2}$  in the time domain will be smeared to all samples in frequency domain. Therefore, the repetition decoder will affect all the bits.

A serious drawback of this procedure is the fact that the inversion of channel response submatrix  $\mathbf{H}_{11}$  is a numerically unstable operation and in the presence of noise can lead to noise amplification rendering the practical limited-precision implementation unreliable. This effect can be reduced by an optimized organization of computations, but there is also another method.

The *basic idea is not to compute any inversion*, but instead use the properties of the channel convolution matrix. We assume that an additive correction, that uses an estimate of previous transmitted block (5), is already applied (the vector  $\mathbf{r}_{cor1}$  is subtracted). This correction removes IBI and must be done in time domain.

As the standard branch of processing depends on the circulant property which is a consequence of cyclic prefix insertion in transmitter and appropriate subblock selection in receiver, in case of the 2nd copy extraction, the circulant property will be provided by a correction done in the

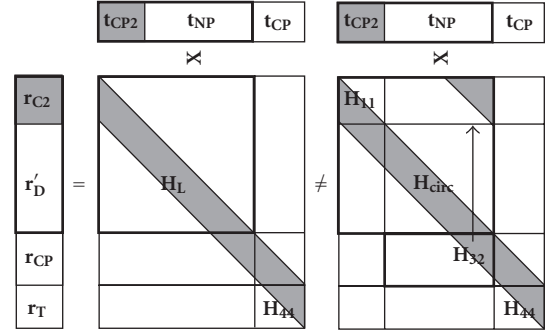


FIGURE 9: The circulant matrix  $\mathbf{H}_{circ}$  is a result of channel convolution and additive correction. Caution:  $\mathbf{r}_{C2}$  is part of  $\mathbf{r}_D'$ .

receiver. The result will be a second copy of frequency domain samples that originates from the otherwise ignored prefix samples.

In Figure 9, the actual situation is shown on the left,  $\mathbf{r}_D'$  is now the important subvector of the received sequence and is related with the interesting block  $\mathbf{t}_D'$  by a simple equation

$$\mathbf{r}_D' = \mathbf{H}_L \times \mathbf{t}_D', \quad (7)$$

( $\mathbf{r}_{C2}$  is considered part of  $\mathbf{r}_D'$ ). Vector  $\mathbf{t}_D' = \{\mathbf{t}_{CP2} \parallel \mathbf{t}_{NP}\}$  is a cyclically shifted version of vector  $\mathbf{t}_D = \{\mathbf{t}_{NP} \parallel \mathbf{t}_{CP}\}$ . Matrix  $\mathbf{H}_L$  is not circulant, but if a second additive correction is applied to the received vector  $\mathbf{r}_D'$  (more specifically to  $\mathbf{r}_{C2}$ ), then the resulting vector is related to the transmitted vector  $\mathbf{t}_D'$  through multiplication with a circulant matrix  $\mathbf{H}_{circ}$ , and therefore can be effectively equalized by a single-tap frequency domain equalizer, just as the  $\mathbf{r}_D$  block in a standard OFDM system. The second additive correction uses the matrix  $\mathbf{H}_{32}$  shown in Figure 9 on the right:

$$\mathbf{r}_{cor2(n)} = \mathbf{H}_{32(n)} \times \mathbf{t}_{NP(n)}. \quad (8)$$

If the correction  $\mathbf{r}_{cor2}$  is applied to  $\mathbf{r}_{C2}$ , then the vector  $\mathbf{r}_D'$  is equalized in the same way and with the same equalizer values as  $\mathbf{r}_D$ .

As apparent in Figure 9, matrix  $\mathbf{H}_{32}$  can be further divided vertically into two submatrices, one of them zero. This division enables reduction of complexity when only a small part of samples of  $\mathbf{t}_{NP}$  needs to be estimated. An example of final  $\mathbf{H}$  matrix partitioning covering both partitionings in Figures 8 and 9 is shown in Figure 7.

## 5. Modified OFDM Receiver

**5.1. Receiver Design.** In Figure 10, the modified OFDM receiver is shown. It consists basically of a standard receiver, fortified with a second copy extraction and simple repetition decoder. An estimate of a subset of samples of transmitter output is computed based on the decoded output of the standard branch. Because a potent ECC is assumed to be present in the system, the reconstruction of transmitter output will be a “good estimate” of the actual values. After the additive corrections  $\mathbf{r}_{cor1}$  and  $\mathbf{r}_{cor2}$  are applied, the resulting time-domain samples (subset of them containing



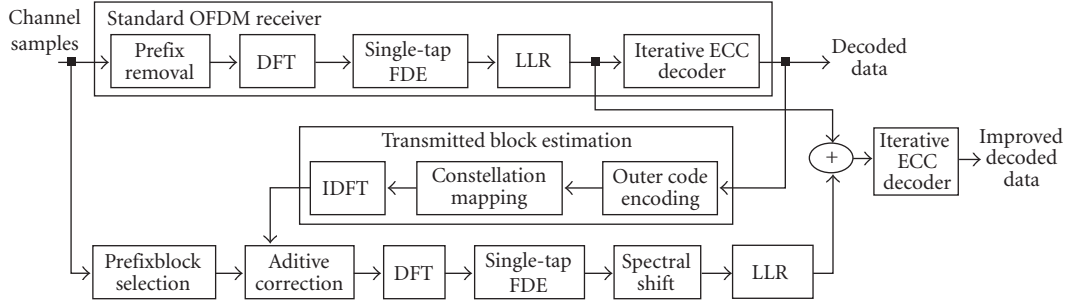


FIGURE 10: Modified OFDM receiver.

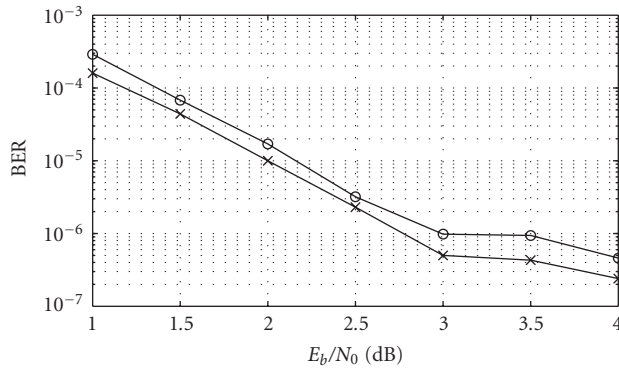


FIGURE 11: Standard (○-marker) and modified (×-marker) receiver error performance after 8 decoder iterations.

new information) are transformed to frequency domain where they can be easily equalized in the same way as shown earlier. Because the 2nd copy of the received block is rotated in time domain by the size of the prefix, another single-tap phase correction must be applied in frequency domain (the “spectral shift” block). This correction is multiplicative and depends only on the size of the shift which is constant for a specific prefix size. A second set of channel LLR values is computed and added to the LLR values from the standard processing branch. This improved sequence then enters the ECC decoder.

As indicated earlier, all of the functional blocks are implemented in software. The key property of the modified branch is that it is built using exactly the same components as the standard processing branch with one exception—the spectral shift block that is implemented as a simple scalar complex multiplication. The development and inclusion of the modification is very straightforward in an SDR receiver. Furthermore, the additional processing can be turned off and on adaptively, depending on the transmission quality requirements and available processing time.

**5.2. Simulation Results.** We simulated a coded OFDM system with an outer RSC turbo code of rate  $R = 1/3$  defined in [13], with cyclic prefix size equal to  $1/8$  of the data block size (as defined in [1]) over a multipath channel with AWGN noise. Each data block of size 1024 bits was after turbo coding mapped to three OFDM symbols of 1024 complex

samples. The values of channel impulse response samples were distributed according to [15].

The error performance of the new system is only slightly better (approx. 0.1 dB) than the basic system. The improvement is most visible in the error floor area (below BER  $10^{-6}$ ) of the suboptimal log-domain iterative decoder of the outer code.

## 6. Conclusion

We have shown that it is possible to exploit the redundancy in a cyclic prefix of OFDM. The modified receiver is fully backward compatible with any existing OFDM-based protocol. The computational complexity is approximately double compared to the standard OFDM receiver. Simulations for specified parameters have shown that a relatively small improvement of 0.1 dB in bit error rate could be achieved thanks to exploitation of the prefix redundancy. Because the modification reuses most of the functional blocks already present in the system, it can be implemented very rapidly in an SDR system using a high-level programming language.

## Acknowledgment

This work was supported by Scientific Grant Agency of Ministry of Education of Slovak Republic and Slovak Academy of Sciences under contract VEGA 1/0376/09, 2009–2012.

## References

- [1] “Mobile WiMax—Part I: A technical Overview and Performance Evaluation,” WiMax Forum, August 2006.
- [2] T. Keller and L. Hanzo, “Adaptive multicarrier modulation: a convenient framework for time-frequency processing in wireless communications,” *Proceedings of the IEEE*, vol. 88, no. 5, pp. 611–640, 2000.
- [3] J. Glossner, M. Moudgill, D. Iancu, et al., “The SandBridge Sandblaster convergence platform,” <http://www.sandbridgetech.com/pdf/sandbridge.white.paper.2005.pdf>.
- [4] P. N. Fletcher, “Iterative decoding for reducing cyclic prefix requirement in OFDM modulation,” *Electronics Letters*, vol. 39, no. 6, pp. 539–541, 2003.
- [5] H.-C. Won and G.-H. Im, “Iterative cyclic prefix reconstruction and channel estimation for a STBC OFDM system,” *IEEE Communications Letters*, vol. 9, no. 4, pp. 307–309, 2005.

- [6] X. Wang and K. J. Ray Liu, "Adaptive channel estimation using cyclic prefix in multicarrier modulation system," *IEEE Communications Letters*, vol. 3, no. 10, pp. 291–293, 1999.
- [7] X. Yuan, Q. Guo, X. Wang, and L. Ping, "Evolution analysis of low-cost iterative equalization in coded linear systems with cyclic prefixes," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 2, pp. 301–310, 2008.
- [8] X. Wang and K. J. Ray Liu, "Performance analysis for adaptive channel estimation exploiting cyclic prefix in multicarrier modulation systems," *IEEE Transactions on Communications*, vol. 51, no. 1, pp. 94–105, 2003.
- [9] B. Devillers, J. Louveaux, and L. Vandendorpe, "Exploiting cyclic prefix for performance improvement in single carrier systems," in *Proceedings of the 7th IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC '06)*, pp. 1–5, Cannes, France, July 2006.
- [10] A. Tarighat and A. H. Sayed, "An optimum OFDM receiver exploiting cyclic prefix for improved data estimation," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '03)*, vol. 4, pp. 217–220, Hong Kong, April 2003.
- [11] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Transactions on Information Theory*, vol. 42, no. 2, pp. 429–445, 1996.
- [12] T. K. Moon, *Error Correction Coding*, John Wiley & Sons, Hoboken, NJ, USA, 2005.
- [13] M. C. Valenti and J. Sun, "The UMTS turbo code and an efficient decoder implementation suitable for software-defined radios," *International Journal of Wireless Information Networks*, vol. 8, no. 4, pp. 203–215, 2001.
- [14] R. Gray, *Toeplitz and Circulant Matrices: A Review*, Stanford University, Stanford, Calif, USA, 1971.
- [15] M. Debbah, "Short introduction to OFDM," [www.supelec.fr/d2ri/flexibleradio/cours/ofdmtutorial.pdf](http://www.supelec.fr/d2ri/flexibleradio/cours/ofdmtutorial.pdf).

## Research Article

# Implementing a DVB-T/H Receiver on a Software-Defined Radio Platform

**Yong Jiang, Wen Xu, and Cyprian Grassmann**

*Infineon Technologies AG, 85579 Neubiberg, Germany*

Correspondence should be addressed to Wen Xu, wen.xu@ieee.org

Received 1 September 2008; Accepted 16 February 2009

Recommended by Daniel Iancu

Digital multimedia broadcasting is available in more and more countries with various forms. One of the most successful forms is Digital Video Broadcasting for Terrestrial (DVB-T), which has been deployed in most countries of the world for years. In order to bring the digital multimedia broadcasting services to battery-powered handheld receivers in a mobile environment, Digital Video Broadcasting for Handheld (DVB-H) has been formally adopted by ETSI. More advanced and complex digital multimedia broadcasting systems are under development, for example, the next generation of DVB-T, a.k.a. DVB-T2. Current commercial DVB-T/H receivers are usually built upon dedicated application-specific integrated circuits (ASICs). However, ASICs are not flexible for incoming evolved standards and less overall-area efficient since they cannot be efficiently reused and shared among different radio standards, when we integrate a DVB-T/H receiver into a mobile phone. This paper presents an example implementation of a DVB-T/H receiver on the prototype of Infineon Technologies' Software-Defined Radio (SDR) platform called MuSIC (Multiple SIMD Cores), which is a DSP-centered and accelerator-assisted architecture and aims at battery-powered mass-market handheld terminals.

Copyright © 2009 Yong Jiang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

## 1. Introduction

The DVB-T system was developed and agreed in 1997 by the DVB Project [1]. In the last ten years, DVB-T systems have been successfully deployed not only in Europe but also in the rest of the world. Now the next generation of DVB-T called DVB-T2 is under development. In order to bring the digital video broadcasting service to battery-powered mobile receivers, for example, mobile phone handheld devices, the DVB-T standard is extended to Digital Video Broadcasting-Handheld (DVB-H) with additional features such as 4K mode, in-depth interleaving, time-slicing, and additional forward error correction [2].

Recently, more radio systems, such as GSM, WCDMA, HSPA, GPS, FM radio, Bluetooth, WiFi, and DVB-H have been integrated into mobile handset terminals because these "all in one" mobile terminals provide an "anytime, anywhere" access to information in an easy way for the end user. In fact, the market already requires this kind of terminals and some manufacturers have promptly reacted, for example, Nokia with its N95 and Apple with its iPhone3G. This

creates new challenges to the semiconductor manufacturers, which should keep the stringent requirements for power consumption, silicon area, and time-to-market together with the increasing requirements for throughput and number of coexisting radio standards in a mobile phone terminal. Infineon Technologies provides its innovative SDR platform MuSIC (Multiple SIMD Cores) to meet the requirements. MuSIC is a DSP-centered and accelerator-assisted architecture [3]. The key attraction of the SDR concept is the ability to support multiple standards on the same chip by changing the software only and the feasibility to share the processing resources among several standards in case of a nonsimultaneous execution. In this work, we present an example implementation of a DVB-T/H receiver on the prototype of Infineon Technologies' Software-Defined Radio platform MuSIC which aims at mass-market handset terminals.

The paper is organized as follows. Section 2 provides a system overview of a DVB-T/H receiver showing the main algorithmic functions comprising the baseband processing chain. The architecture of MuSIC and its programming

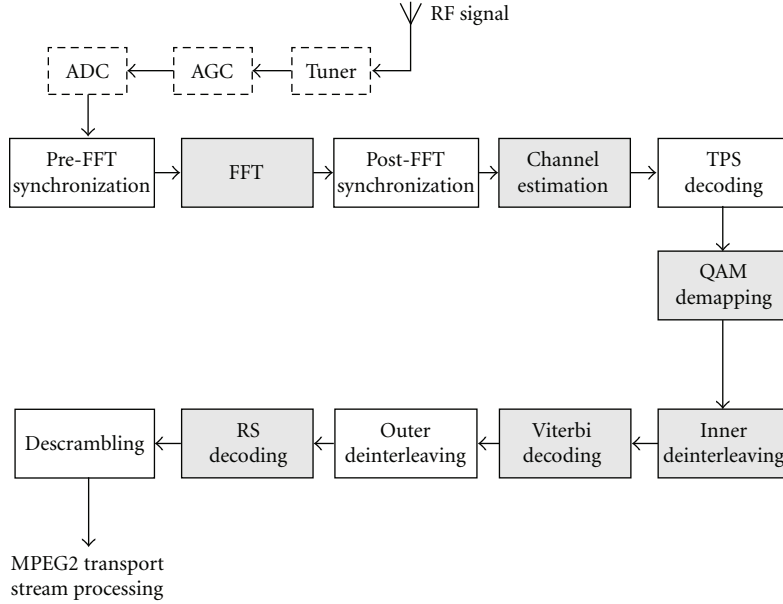


FIGURE 1: Functional block diagram of a DVB-T/H receiver.

model are introduced in Section 3. Section 4 investigates the computational requirements of DVB-T/H and its potential for parallelization on MuSIC. In Section 5, some conclusions and hints for the future work are given.

## 2. DVB-T/H Receiver Algorithms

In this work, we only consider the physical layer processing of the DVB-T/H receiver and omit all analog components, higher layer protocols, and application processing. The functional block diagram in Figure 1 shows a conventional DVB-T/H receiver structure. The RF signal is received by the receiver antenna, downconverted by the tuner circuitry, scaled by an Automatic Gain Control (AGC) circuitry, and then digitized by the Analog-to-Digital Converter (ADC). The baseband processor receives the digitized signal as complex samples from the ADC and delivers the descrambled MPEG transport stream to a higher layer protocol and application processor.

**2.1. Synchronization and Fast Fourier Transformation.** In a typical receiver, a pre-FFT acquisition stage is required to obtain the OFDM symbol timing, the OFDM symbol length, that is, size of the Fast Fourier Transformation (FFT), and the cyclic prefix length, where the latter two parameters are adjustable by the transmitter. The principle of the pre-FFT synchronization is based on the availability of the cyclic prefix in the OFDM symbols [4]. In addition, the carrier frequency offset (CFO) existing between the transmitter and the receiver is also partly estimated in this functional block, that is, only the fractional part of it, and compensated in time domain. Hence, only CFOs which are an integer value of the subcarrier spacing will remain in the OFDM signal. The recovered OFDM symbols are transformed by means

of the FFT, which acts as a matched filter for the OFDM signal. The post-FFT synchronization obtains estimations for the integer part of the CFO and the initial sampling clock frequency offset (SCFO) in frequency domain, which are then compensated in time-domain OFDM signal in order to reduce Inter-Carrier Interference (ICI).

After the acquisition phase, the pre-FFT synchronization is inactive, and the post-FFT synchronization is turned to tracking mode. Due to the instability and drift of the oscillator at receiver the CFO and SCFO will vary during the data receiving phase. Therefore, it is necessary to track the small residual SCFO and CFO to ensure the orthogonality of the OFDM subcarriers and accurate timing after the acquisition phase. The detectors of the residual CFO and SCFO are based on the temporal correlation of OFDM signal in frequency domain. For the investigated DVB-T/H receiver, the continual pilot signals are used to estimate the residual SCFO and CFO. The principle of the tracking methods is described in [5]. Figure 2 shows an example behavior of the implemented SCFO tracking algorithm under additive white Gaussian noise (AWGN) channel with 4 dB channel SNR and an SCFO of 30 ppm. Our simulations show that in the tracking mode, the residual SCFO and CFO, depending on the reception channel conditions, remain very small after a number of OFDM symbols.

A more detailed analysis of the synchronization strategies for an OFDM-based receiver is provided in [6]. It should be noticed that most of the synchronization tasks are only active during the acquisition phase. Because of the use of time-slicing in DVB-H it is more necessary to minimize the whole synchronization time.

**2.2. Channel Estimation, Equalization, and TPS Decoding.** After FFT the impairment of the transmission channel, especially the Doppler frequency drift by high-speed mobility of

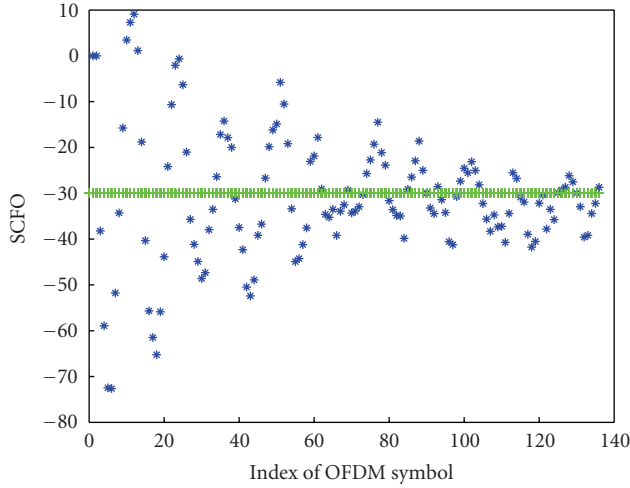


FIGURE 2: Typical behavior of the implemented SCFO tracking algorithm for SCFO = 30 ppm. (“+” = actual SCFO, “\*” = SCFO estimated by the tracking algorithm).

the handheld terminal, will be mitigated through the channel equalization which is characterized as

$$X_{l,k} = \frac{Y_{l,k}}{H_{l,k}}, \quad (1)$$

where  $k$  is the subcarrier index in the OFDM symbols, and  $l$  is the OFDM symbol index in the OFDM frame.  $X$  and  $Y$  are the equalized and received OFDM symbols, respectively.  $H_{l,k}$  are the coefficients of the channel transfer function (CTF). In order to estimate the coefficients  $H_{l,k}$ , two cascaded interpolation filters are used as proposed in [7].

The principle of the channel estimation is depicted in Figure 3. The value of  $X_{l,k}$  and  $Y_{l,k}$  corresponding to the scattered pilots is known beforehand at the receiver. Therefore, at a first stage the CTF coefficients  $H_{l,k}$  for the scattered pilots can be readily estimated using (1). For subcarriers sharing the same index  $k_0$ , a 6-tap time-direction interpolation filter is used in which the six closest scattered pilots with the same index  $k_0$  are used as supporting points, thereby obtaining the coefficients  $H'_{l,k_0}$ . Afterwards, both  $H$  and  $H'$  are used as the supporting points in a 12-tap frequency-direction filter to calculate the CTF for the remaining subcarriers. Thus, those positions for which a coefficient  $H'$  is computed can be seen as “pseudo” pilots, which are marked as  $\otimes$  in Figure 3. The interpolation filters use the scattered pilots in each OFDM symbol as a priori knowledge. Therefore, the scattered pilots have to be determined before the channel estimation can be started.

In a regular DVB-T receiver the exact positions of the scattered pilots are determined after decoding the transmission parameter signaling (TPS) bits. The TPS bits also provide information about other system parameters required to demodulate and decode the received signal. They are organized into blocks of 68 bits, and each bit is transmitted on several subcarriers within one OFDM symbol. Hence, 68 symbols, that is, one OFDM frame in DVB-T/H, are required to transmit a whole TPS block. The TPS bits are modulated

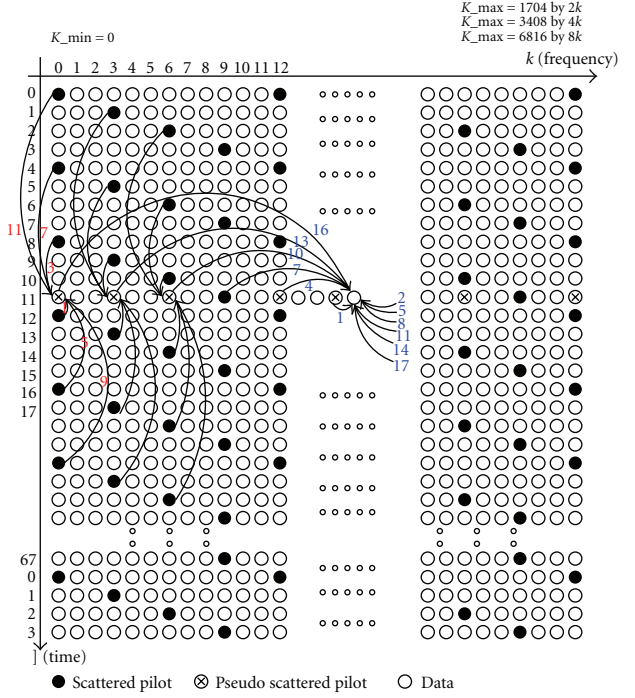


FIGURE 3: DVB-T/H frame structure and illustration of the channel estimation principle.

using Differential Binary Phase Shift Keying (DBPSK) in the time direction. This allows recovery of the bits without channel equalization.

The OFDM frame synchronization using TPS bits implies a synchronization time ranging from 17 up to 84 OFDM symbols depending on the very first received OFDM symbol in the frame. However, such a long synchronization time is not acceptable in a DVB-H receiver due to time slicing, where the receiver is allowed to power off when it is inactive leading to significant power savings. To solve this problem a fast scattered pilot synchronization scheme has been proposed in [8], which needs only one OFDM symbol to identify the scattered pilot and is based on the temporally repetitive structure of the scattered pilots. This method also makes it possible to execute the channel estimation and equalization prior to TPS bit decoding, thereby improving the robustness of the TPS decoding itself under extremely bad transmission conditions [9]. After DBPSK decoding of the TPS bits, the frame synchronization is required to determine the position of the respective TPS bit in one TPS block, as the semantics of each TPS bit defined in [10] depends on its position in one TPS block.

**2.3. Demapping.** With the decoded TPS signal the constellation parameters used by the transmitter for the QAM Gray-mapping are made available to the receiver. The QAM demapper can then recover data bits from the data subcarriers, that is, one QAM symbol demapped onto one  $v$ -bit binary word according to Gray-mapping and the constellation parameters, where  $v$  equals 2 for QPSK, 4 for 16-QAM, and 6 for 64-QAM. Soft decision must be



available to the Viterbi decoder. It improves the error-correction capability and makes the correct interpretation of depunctured bits possible [11]. Every bit of the  $v$ -bit binary word is represented as one  $n$ -bit binary word to represent the signed distance between the constellation point and the decision border, called softbits. A higher resolution of the soft decision increases not only the gain of the Viterbi decoder but also the implementation complexity. A tradeoff between performance and implementation complexity has to be found by means of simulation. Our simulation results show that a 5-bit resolution is a proper choice.

**2.4. Inner Deinterleaving and Viterbi Decoding.** In order to improve the long burst error correction capability, several interleaving stages are specified in the DVB-T/H standard. Because of the bitwise sequent address generation algorithm and the large range of the symbol deinterleaving (e.g., the interleaving over 6048 QAM symbols by 8k mode), a software implementation for the deinterleaving in SIMD core is not efficient. To avoid the sequent address generation a lookup table can be used, but it will be very large for on-chip memory. The inner deinterleaving can be done better with a scalar processor or in hardware accelerator. Following the deinterleaver, a Viterbi decoder with soft decision is used to perform convolutional decoding.

**2.5. Outer Deinterleaving, Reed-Solomon Decoding, and Descrambling.** The bit stream obtained after the convolutional decoding is reorganized as a byte stream and further processed by the outer deinterleaver, thereby improving the long burst correction capabilities of the receiver. Afterwards, the byte stream is processed by Reed-Solomon decoder and descrambled at byte level.

The functions marked in gray in Figure 1 are core functions demanding most of the computing power and storage capacity and are therefore the critical functions for a software implementation on an SDR platform. We have analyzed their computational requirements and investigated the possibilities for a parallel realization on the MuSIC platform, as it will be discussed in Section 4. To achieve best performance under various network conditions and coverage scenarios, the DVB-T/H standard provides the network operators numerous system parameters which are listed in Table 1. The combinations of these parameters derive a net bit rate from 3.11 Mbit/s at 5 MHz channel bandwidth, QPSK, 1/2 code rate, 1/4 guard interval, up to 31.67 Mbit/s at 8 MHz channel bandwidth, 64-QAM, 7/8 code rate, 1/32 guard interval. The receiver must comply with all the possible combinations of parameters the transmitter may use. At this point it is reasonable to analyze only the worst case scenario, that is, a transmission at 31.67 Mbit/s.

The whole DVB-T/H receiver described has a good system performance. Our simulation results show that for AWGN channel with a channel SNR of more than 4 dB, or for typical urban channel with vehicular speed of 6 km/h (TU6) and a channel SNR of more than 10 dB, user data can be decoded with a bit error rate smaller than  $10^{-3}$ .

TABLE 1: DVB-T/H system parameters (see [10]).

Elementary period $T$	7/64, 1/8, 7/48, 7/40 [ $\mu$ s]
Channel bandwidth $B$	8, 7, 6, 5 [MHz]
Number of carrier $N$	8k, 4k, 2k
Number of guard $N_g$	1/32, 1/16, 1/8, 1/4 [in $N$ ]
Number of used carrier $N_u$	6817, 3408, 1705
Number of data carrier $N_d$	6048, 3024, 1512
Code rate $r$	7/8, 5/6, 3/4, 2/3, 1/2
Symbol duration $T_u$	$T_u = T \cdot N$
Carrier spacing $\Delta f$	$1/T_u$
QAM constellation	64, 16, 4

### 3. Architecture and Programming Model

The intensive work carried out at Infineon Technologies resulted in a versatile processor architecture which is able to cope with the performance, power, and area requirements of a multistandard SDR approach [12, 13]. This processor mainly consists of a cluster of four single-instruction multiple-data (SIMD) DSP cores (see Figure 4). Each SIMD core contains four processing elements (PEs) and operates with a clock frequency of 300 MHz. To relax the timing requirements for the memory and to resolve pipeline hazards, each core runs four threads which are switched by a fixed time multiplexing mechanism. This is equivalent to running 16 threads at 75 MHz each.

Long instruction words (LIWs) of the PE array show memory, arithmetic, and communication components. The SIMD core controller is in fact a 32-bit general purpose processor (GP). The GP communicates with the other units via instruction and data FIFOs.

The cluster of the SIMD cores is accompanied by dedicated configurable hardware accelerators for coding/decoding and for filtering operations. In addition, there is an ARM processor for the execution of the protocol stacks. For a more detailed discussion about the MuSIC processor, we refer to [14–16].

The programming model for this architecture is the multithreaded programming in C. Wrapped into functions called by threads, the purely data parallel parts (associated with SIMD cores) are programmed in a data parallel language extension of C.

To support multithreaded programming, the Infineon Lightweight Operating System (ILTOS) has been developed. It provides the means to create and synchronize threads to asynchronously send and query messages between them and to allocate and free shared memory.

Functions to be executed on an SIMD core are written in Data Parallel C Extension (DPCE) language, a superset of the C language [17]. DPCE offers parallel data types and operations on them. A compiler which takes a DPCE source and produces synchronized C code for GP core and DMA transfers (to be translated further by a C compiler for the GP) as well as PE assembly was developed at Infineon. This compiler is not yet optimized, though. To achieve best performance, inline assembly is used for the PE array and

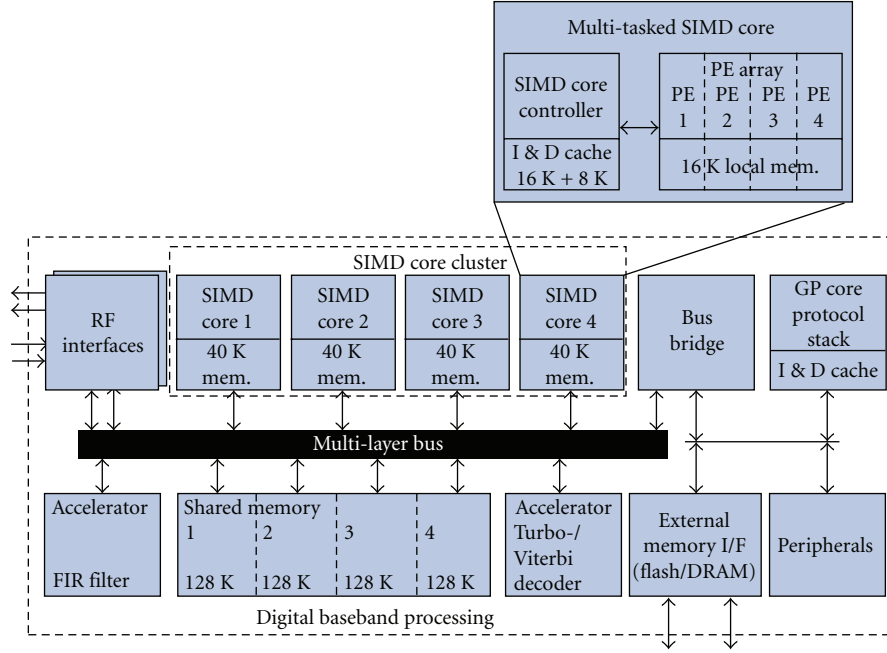


FIGURE 4: The SDR baseband platform MuSIC.

explicit DMA configuration. What remains is mainly the C language with some intrinsic functions for PE and DMA control plus an assembly source code library for the PE. Implementations can be done completely without PE and DMA by writing pure C programs. These will then run on the GP core alone. This feature is of importance for testing assembly implementations.

Moreover, a virtual prototype of the entire MuSIC platform based on SystemC has been developed at Infineon. The virtual prototype is a cycle- and bit-accurate software-based simulator. It contains models of all processors, accelerators, busses, memories, and peripherals which will be available in the real hardware. The same software can be run on both the virtual prototype and the real hardware.

#### 4. Computational Requirements for an Implementation on MuSIC

**4.1. Fast Fourier Transformation (FFT).** FFT is the core function of an OFDM receiver. The most commonly used algorithm for FFT calculation is the well-known butterfly algorithm. The computational requirement of FFT is very high, especially for the large FFT block which in DVB-T/H can be 8192 complex samples. The theoretical computational complexity of FFT with  $N$  complex samples based on radix-2 algorithms is given as follows:

$$\begin{aligned} \frac{N}{2} \cdot \log_2 N, & \text{ complex multiplications,} \\ N \cdot \log_2 N, & \text{ complex additions.} \end{aligned} \quad (2)$$

In the case  $N$  equals 8192, 4 instructions are needed for 1 complex multiplication and 2 instructions are needed for

1 complex addition, it leads to 425984 instructions for 8k mode based on radix-2 without data level parallelism. In the ideal parallel case of 4 data paths, that is, 4 PEs in an SIMD core, the cycle count can be reduced from 425984 to 106496. An example implementation of 8k FFT on MuSIC was measured out. It shows that about 20 percent overhead in cycles is needed to overcome all data transfer and temporary storage. This shows the MuSIC architecture fully exploits the data level parallelism of the FFT algorithms.

**4.2. Channel Estimation and Equalization.** Section 2.2 describes the channel estimation in our DVB-T/H receiver. Its computational complexity is analyzed here. The first step, that is, the interpolation in time direction, is carried out for those subcarriers  $k_0 = 0, 3, 6, \dots$  for which a scattered pilot is available (see Figure 3). It can be described as follows:

$$\begin{aligned} H'_{l,0} &= w_1(1)H_{l-11,0} + w_1(2)H_{l-7,0} + \dots + w_1(6)H_{l+9,0}, \\ H'_{l,3} &= w_2(1)H_{l-10,3} + w_2(2)H_{l-6,3} + \dots + w_2(6)H_{l+10,3}, \\ H'_{l,6} &= w_3(1)H_{l-9,6} + w_3(2)H_{l-5,6} + \dots + w_3(6)H_{l+11,6}. \end{aligned} \quad (3)$$

The second step is based on the scattered pilots  $H$  and the estimations  $H'$  computed in time direction within one OFDM symbol, namely

$$\begin{aligned} H''_k &= v_1(1)H_{k-16} + v_1(2)H_{k-13} + \dots + v_1(12)H_{k+17}, \\ H''_{k+1} &= v_2(1)H_{k-17} + v_2(2)H_{k-14} + \dots + v_2(12)H_{k+16}, \end{aligned} \quad (4)$$

where  $w_n(m)$  and  $v_n(m)$  are the filter coefficients and depend on the distance between the supporting point and the

TABLE 2: Computational requirements.

Functional unit	MOPS
FFT	461
Channel estimation and correction	155
Soft demapping and quantization	170
Viterbi decoding	4868*
RS decoding	24*
Total	5678 (786**)

The worst case is considered here, that is, 8 MHz channel bandwidth, 8k mode, 1/32 guard interval, 64 QAM, 7/8 code rate, and 31.67 Mbit/s.

\*Based on [18] and scaled according to the system parameters.

\*\*Without Viterbi and RS decoding.

estimated point, which is also shown in Figure 3. Therefore, it needs  $1 \times 2 = 2$  real multiplications (RMs) to calculate  $H$ . There are 568 scattered pilots in one 8k OFDM symbol. For each pseudopilot  $H'$ , 12 real multiplication-accumulations (rMACs) are required. There are 1705 pseudopilots in one 8k OFDM symbol. 24 rMACs are required for each of the remaining sub-carriers to calculate its CTF coefficient  $H''$ . There are 4544 such subcarriers in one 8k OFDM symbol. The channel correction requires 2 RMs for every payload subcarrier. Because split of the TPS and continual pilots from the payload subcarriers by means of software is even more expensive than the equalization for the TPS and continual pilots, we equalize all the subcarriers excluding the scattered pilots, that is, 6249 subcarriers need to be corrected.

Together it needs 13634 RMs and 129516 rMACs per OFDM symbol in 8k mode for the channel estimation and correction, which equals 143150 instructions without parallel processing. Because the time direction interpolation is not causal, FIFO buffering is required to delay the input OFDM symbols. In the case of the 6-tap time direction filter, all subcarriers  $Y$  within 12 OFDM symbols and the CTFs  $H$  at all scattered pilot positions within 23 OFDM symbols need to be stored, which leads to the most memory demand of the DVB-T/H receiver.

**4.3. Demapping.** The Gray mapping is used in DVB-T/H. The demodulation of one subcarrier, in the case of 64-QAM, needs 8 operations. The quantization of one soft decision needs 3 operations and in the case of 64-QAM, each payload subcarrier implies 6 soft-decisions. So  $6048 \times (8 + 6 \times 3) = 157248$  operations are needed for the demapping and quantization of one OFDM symbol.

The computational complexity of each function is listed in Table 2 as required million operations per second (MOPS), where one OFDM symbol duration is 924 microseconds, which determines the real-time processing requirement. The Viterbi and Reed-Solomon decoders demand enormous computing power and are therefore implemented as hardware accelerators. In this manner the computational requirements can be reduced from formerly 5678 MOPS down to 786 MOPS. For 4 data paths in data-level parallelism the real-time processing requirement is reduced to 197 million cycles per second. With use of thread level parallelism, this computing requirement can be

affordably met with 3 of 16 threads at 75 MHz on MuSIC in the case of about 15 percent implementation overhead and can be sufficiently met with 4 threads, that is, one SIMD core, in the case of about 50 percent implementation overhead.

## 5. Conclusions

In this paper, we first analyzed the example algorithms of a DVB-T/H receiver in detail, and then gave a brief introduction on the architecture and programming model of the prototype of Infineon SDR platform MuSIC. Based on the algorithms and the hardware architecture, and based on our implementation on MuSIC, we estimated and partly measured the computational requirements of the relevant functions for a DVB-T/H receiver on MuSIC. The results show that it is feasible to implement a DVB-T/H receiver on MuSIC with one of the four SIMD cores. It should be noted that the control functions for the respective functional blocks have not been considered till now. This will be studied in the future work.

## Acknowledgments

The authors gratefully acknowledge fruitful discussions with Mirko Sauermann, Mathias Richter, Dominik Langen, Reinhard Rueckriem, and Professor Ulrich Ramacher. This work has been supported by the German BMBF (Bundesministerium für Bildung und Forschung) project MxMobile.

## References

- [1] "History of the DVB Project," <http://www.dvb.org/>.
- [2] ETSI, EN 302 304 v1.1.1, "Digital Video Broadcasting (DVB); Transmission System for Handheld Terminals (DVB-H)," November 2004.
- [3] U. Ramacher, "Software-defined radio prospects for multi-standard mobile phones," *Computer*, vol. 40, no. 10, pp. 62–69, 2007.
- [4] J.-J. van de Beek, M. Sandell, and P. O. Börjesson, "ML estimation of time and frequency offset in OFDM systems," *IEEE Transactions on Signal Processing*, vol. 45, no. 7, pp. 1800–1805, 1997.
- [5] M. Speth, S. Fechtel, G. Fock, and H. Meyr, "Optimum receiver design for OFDM-based broadband transmission—part II: a case study," *IEEE Transactions on Communications*, vol. 49, no. 4, pp. 571–578, 2001.
- [6] M. Speth, S. A. Fechtel, G. Fock, and H. Meyr, "Optimum receiver design for wireless broad-band systems using OFDM—part I," *IEEE Transactions on Communications*, vol. 47, no. 11, pp. 1668–1677, 1999.
- [7] P. Hoher, S. Kaiser, and P. Robertson, "Two-dimensional pilot-symbol-aided channel estimation by Wiener filtering," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '97)*, vol. 3, pp. 1845–1848, Munich, Germany, April 1997.
- [8] L. Schwoerer, "Fast pilot synchronization schemes for DVB-H," in *Proceedings of the 4th IASTED International Multi-Conference on Wireless and Optical Communications*, pp. 420–424, Banff, Canada, July 2004.

- [9] H. Ye, "TPS decoder in an orthogonal frequency division multiplexing receiver," US patent no. 7123669, 2006.
- [10] ETSI, EN 300744 V1.5.1, "Digital Video Broadcasting (DVB); Framing Structure, Channel Coding and Modulation for Digital Terrestrial Television," November 2004.
- [11] U. Reimers, *DVB the Family of International Standards for Digital Video Broadcasting*, Springer, Berlin, Germany, 2005.
- [12] C. Grassmann, M. Richter, and M. Sauermann, "Mapping the physical layer of radio standards to multiprocessor architectures," in *Proceedings of the Conference on Design, Automation and Test in Europe (DATE '07)*, pp. 1412–1417, Nice, France, April 2007.
- [13] H.-M. Blüthgen, C. Sauer, M. Gries, et al., "Finding the optimum partitioning for multi-standard radio systems," in *Proceedings of the Software Defined Radio Technical Conference (SDR '05)*, pp. 1–6, Orange County, Calif, USA, November 2005.
- [14] W. Raab, H.-M. Blüthgen, and U. Ramacher, "A low-power memory hierarchy for a fully programmable baseband processor," in *Proceedings of the 3rd Workshop on Memory Performance Issues (WMPI '04)*, pp. 102–106, Munich, Germany, June 2004.
- [15] H.-M. Blüthgen, C. Grassmann, W. Raab, U. Ramacher, and J. Hausner, "A programmable baseband platform for software-defined radio," in *Proceedings of the Software Defined Radio Technical Conference (SDR '04)*, Phoenix, Ariz, USA, November 2004.
- [16] H.-M. Blüthgen, C. Grassmann, and U. Ramacher, "A software programmable multiple-standard radio platform," in *Proceedings of the 14th IST Mobile & Wireless Communications Summit*, pp. 1–5, Dresden, Germany, June 2005.
- [17] "Data Parallel C Extensions (DPCE)," <http://www.crescentbaysoftware.com/dpce/index.html>.
- [18] M. Hosemann, G. Cichon, P. Robelly, et al., "Implementing a receiver for terrestrial digital video broadcasting in software on an application-specific DSP," in *Proceedings of IEEE Workshop on Signal Processing Systems (SIPS '04)*, pp. 53–58, Austin, Tex, USA, October 2004.

## Research Article

# Galois Field Instructions in the Sandblaster 2.0 Architecture

Mayan Moudgill,<sup>1</sup> Andrei Iancu,<sup>1</sup> and Daniel Iancu<sup>1,2</sup>

<sup>1</sup> Sandbridge Technologies, Inc., White Plains, NY 10601, USA

<sup>2</sup> Tampere University of Technology, Korkeakoulunkatu 1, 33720 Tampere, Finland

Correspondence should be addressed to Daniel Iancu, diancu@sandbridgetech.com

Received 20 November 2008; Revised 17 May 2009; Accepted 26 August 2009

Recommended by Mihai Sima

This paper presents a novel approach to implementing multiplication of Galois Fields with  $2^N$ . Elements of  $GF(2^N)$  can be represented as polynomials of degree less than  $N$  over  $GF(2)$ . Operations are performed modulo an irreducible polynomial of degree  $n$  over  $GF(2)$ . Our approach splits a Galois Field multiply into two operations, polynomial-multiply and polynomial-remainder over  $GF(2)$ . We show how these two operations can be implemented using the same hardware. Further, we show that in many cases several polynomial-multiply operations can be combined before needing to a polynomial-remainder. The Sandblaster 2.0 is a SIMD architecture. It has SIMD variants of the poly-multiply and poly-remainder instructions. We use a Reed-Solomon encoder and decoder to demonstrate the performance of our approach. Our new approach achieves speedup of 11.5x compared to the standard SIMD processor of 8x.

Copyright © 2009 Mayan Moudgill et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

## 1. Introduction

Galois Field arithmetic is widely used in applications such as error-correcting codes and cryptography. Generally, the Galois Fields used are  $GF(2^N)$  for some  $N$ . Elements of  $GF(2^N)$  may be represented as polynomials of degree less than  $N$  over  $GF(2)$ . Operations are performed modulo some polynomial  $P$ , where  $P$  is an irreducible polynomial of degree  $N$  over  $GF(2)$ .  $P$  is known as the prime polynomial. The multiplication of two elements  $X$  and  $Y$  can be accomplished by multiplying their polynomial representations, and then computing the remainder modulo  $P$ .

Conventionally, these polynomials are represented as binary numbers, where the  $i$ th term is represented by setting the  $i$ th bit to 1 or 0 depending on whether that term is present. Thus, the polynomial  $x^4 + x + 1$  would be represented as 10011.

Addition of  $GF(2^N)$  values in this representation is straightforward; it is simply an exclusive-or (xor) of the two binary numbers. Galois Field multiplication (GFM), however, is much more complicated. It involves the following steps.

- (i) Do a polynomial-multiply of two inputs.

- (ii) Do a polynomial-remainder of the product modulo a third input, the prime polynomial.

In software, GF multiplications are usually performed using Look-up Tables (LUTs). For large  $N$ , the LUT becomes rather large, requiring prohibitive large memory size. The processing time also becomes prohibitive at high data rates.

To further complicate issues, processors need to be able to handle Galois Fields of different lengths. Consequently, several processors have added instructions for Galois Field Multiplication (GFM). The most representative is the TI C64x DSP.

A general purpose GFM instruction needs 4 inputs, at least 3 of which must be in registers, the two inputs and the prime polynomial. Since most instruction-sets do not provide for 4 inputs fields, a GFM instruction generally uses a special-purpose register that provides either the length or the prime, or both. For instance, the GMPY4 operation in the TI C64x DSP uses the GFPGR special purpose register to specify the length and prime polynomial [1].

In this paper, we describe an approach that implements GFM using two instructions, one of which implements polynomial-multiply over  $GF(2)$ , and the other of which implements the polynomial-remainder over  $GF(2)$ .



In the Sandblaster 2.0 architecture [2], these instructions are called `gfmul` and `gfnormi`, respectively. Both of these instructions use two register inputs. The `gfnormi` additionally has a third immediate input, the length of the polynomial.

The Sandblaster 2.0 has a 16-way SIMD unit; consequently, we also have 16 way variants of polynomial-multiply and polynomial-remainder instructions called `rgfmul` and `rgfnorm`. Additionally, the SIMD unit supports polynomial-multiply-and-add and polynomial-multiply-and-reduce instructions called `rgfmac` and `rgfmulred`.

It turns out that it is possible to specify the `gfmul` and `gfnormi` operations in such a fashion that we can use almost identical hardware to implement both functions. Consequently, there is no hardware overhead to split the GFM operation into two operations.

The Galois Field sum of several GFM operations can be simplified to the polynomial sum (i.e., xor) of several polynomial-multiplies, followed by a single polynomial remainder. This is quite common in several algorithms that use Galois Field arithmetic. In those cases, we can implement the sum of  $N$  GFM operations using  $N$  polynomial-multiplies and 1 polynomial-remainder, incurring a 1 instruction overhead because of our split implementation.

Section 2 of this paper describes the GFM instructions in the Sandblaster 2.0 architecture. Section 3 focuses on the implementation of the operations. Section 4 examines the performance of GFM instructions in the context of Reed-Solomon encoding/decoding. We conclude in Section 5.

## 2. Galois Field Multiply Instructions

The Sandblaster scalar unit has 16 32-bit general purpose registers. Like most RISC architectures, at most 2 registers can be read and 1 written per integer operation. An integer operation has fields to specify up to 3 registers. An extended immediate variant of the instruction can additionally provide up to 12 bits of immediate data.

**2.1. Polynomial Representation.** In the customary binary representation of polynomials over  $GF(2)$ , the bits are right-aligned, with the LSB bit representing the coefficient of term  $x^0$ , and bit  $i$  representing the coefficient of term  $x^i$ . By contrast, we left-align the coefficients, so that the coefficient of the largest term of the polynomial is represented by MSB. For a polynomial of degree  $N$ , bit  $31-i$  of the general purpose register represents the coefficient of term  $x^{N-i}$ .

Note that in this representation, without knowing the length of the polynomial, we cannot be sure which polynomial is represented by a specific number. For instance, `0xb000_0000` could be interpreted as  $x^3 + x + 1$  if the polynomials are of degree 3 or  $x^5 + x^3 + x^2$  if the polynomial is of degree 5.

We picked this representation to make it easier to compute the remainder. Since the high-order term of the divisor and dividend is left-aligned, we can start subtraction without requiring any shifting to line up the start of the polynomials.

For correctness, it is assumed that all unused bits in the register are 0. Both polynomial-multiply and -remainder are implemented so that they leave their results left-aligned with unused bits as 0.

There is one wrinkle about the representation. We assume that the polynomial remainder is performed with a left-aligned divisor so that the MSB is always 1. In this case, representing the leading coefficient is redundant. So, we do not represent the leading bit of a divisor polynomial. Instead the MSB represents the coefficient of the second highest term  $x^{N-1}$ . For instance, the divisor polynomial  $x^6 + x^3 + 1$  is represented as `0x2400_0000`; that is, with the leading 6 bits being `001001`.

**2.2. Polynomial Operations.** The poly-multiply instruction in the Sandblaster architecture, `gfmul`, has the following format:

```
gfmul rc,ra,rb
```

It does a polynomial multiplication of the upper-most 8 bits of `ra` and with the upper 8 bits of `rb`, and writes the 15 bit result of the poly-multiply in the upper bits of the target register `rc`. The remaining bits of `rc` are zeroed.

The poly-remainder instruction in the Sandblaster architecture, `gfnormi`, has the format

```
gfnormi rt,rc,rp,J
```

The dividend is the 32-bit number formed by the upper 16 bits of `rc` right padded with 0. The divisor is the 17 bit number formed by prepending a 1 to the upper 16 bits of `rp`. `J` is an immediate operand ranging from 0 to 7. The `gfnormi` instruction performs  $J + 1$  poly-division steps, leaving the remainder in the  $16-(J + 1)$  upper-most bits of the target register `rt`.

**2.3. Galois Field Multiplication.** Implementing a GFM over  $GF(2^K)$  with  $K + 1$  bit prime polynomial  $P$  uses the following setup:

- (i) the product inputs are stored in the upper  $K$  bits of two registers, `ra`, `rb`,
- (ii) the leading bit of  $P$  is dropped and the remaining  $K$  bits are stored in the upper  $K$  bits of a register, `rp`,
- (iii) all unused bits are set to 0.

After executing the following code sequence, the final result of the GFM will be the upper  $K$  bits of `rt`:

```
gfmul rc,ra,rb
gfnormi rt,rc,rp,K-1
```

Table 1 shows an example of Galois Field multiplication over  $GF(2^6)$  of the two numbers `101100` and `011011` with the prime polynomial `1001001`. This results in an intermediate product of `01111010100` with a final remainder of `101010`. In Table 1, the column on the right shows how the values will be stored in the corresponding registers.

TABLE 1: Galois field multiply GF(2<sup>6</sup>).

	10_1100	ra	0xb000_0000
⊗	01_1011	rb	0x6c00_0000
=	011_1101_0100	rc	0x7a80_0000
%	100_1001	rp	0x2400_0000
=	10_1010	rt	0xa800_0000

2.4. *SIMD*. The SIMD unit in Sandbridge 2.0 architecture has eight 16 × 16-bit SIMD registers as well as four accumulator registers. The instruction encoding for SIMD operations allows for 4 input fields. The SIMD unit allows three registers to be read and 2 to be written by an instruction.

The SIMD unit supports GFM through the *rgfmul* and *rgfnorm* instructions, which have the following format:

```
rgfmul vc, va, vb
rgfnorm vt, vc, vp, J
```

These instructions do 16 poly-multiplies/poly-remainders in parallel. Since the SIMD register elements are 16-bit wide, the *rgfmul* uses the upper 8 bits of each element, while the *rgfnorm* uses the entire 16 bits of the element. Other than that, their behavior is identical to the *gfmul*/*gfnormi* instructions.

Upto three SIMD registers can be read per cycle; we use the extra read-port to implement a poly-multiply-and-add instruction with the format:

```
rgfmac vc, va, vb, vs
```

The *rgfmac* instruction 16 poly-multiplies of the 16 elements of *va* and *vb*, and then poly-adds (xor's) the products with the corresponding elements of *vs*.

The SIMD unit has an idiom where the 16 results of element-wise operations (such as *rgfmul*) are combined together to form a single value that is written to the accumulator. The poly-multiply-and-sum-reduce instruction follows this idiom

```
rgfmulred act, va, vb
```

The 16 elements of *va* and *vb* are poly-multiplied together, and the 16 resulting products are poly-summed (xor-ed) together to form a single 16-bit value that is written to the accumulator register.

### 3. Implementation

The *gfnormi* and *gfmul* instructions can be implemented by the same block with very little overhead. As can be seen from the pseudocode in Algorithm 1, the algorithms for the two involve the same computational kernel and differ in their setup and controls. They are described in detail below.

3.1. *gfnormi*. The *gfnormi* instruction computes the remainder using polynomial long division. Since the values are left-aligned, we start the process at bit 31 of the dividend value.

```
gfp (ismul, ra, rb, J)
/*setup*/
if (ismul)
    res = 0 × 00000
    div = 0 × 00.rb [31:24]
    N = 8
else
    res = ra [31:17]
    div = rb [31:17]
    N = J + 1
/*shift/xor stages*/
for (i = 0; i < 8; i++)
    if (ismul)
        isxor = ra [31 - i]
        issh = true
    else
        isxor = res [15] && i < N
        issh = i < N
    if (isxor)
        res = (res << 1) ^ div
    else if (issh)
        res = (res << 1)
```

ALGORITHM 1: Gf-op pseudocode.

The divisor consists of a leading 1 and the upper 16 bits of the divisor register. The immediate argument to the *gfnormi* instruction specifies the number of divide steps executed, *J* + 1. For example, 5 steps of the division of 011.1101.0100 by 100.1001 will proceed as follows:

```
01111010100000000
00000000000000000
11110101000000000
10010010000000000
11001110000000000
10010010000000000
10111000000000000
10010010000000000
01010100000000000
00000000000000000
10101000000000000
```

At each step, the result is xor-ed with 0 or with the divisor, depending on the leading bit being 0/1. The result is then left-shifted by 1 to ensure that the remainder after the division step is left-aligned. Note that xor-ing with 0 is the identity operation; this results in just a left-shift. This is done when the intermediate remainder is smaller than the divisor.

3.2. *gfmul*. Each poly-multiply step needs to follow the same pattern as the poly-remainder so that much of the hardware is common. If we are going to *J* + 1 steps, we do the following:

- (i) the partial result is initialized to all 0s,
- (ii) the “divisor” at each step is one of the multiply inputs prepended with *J* + 1 zeroes,

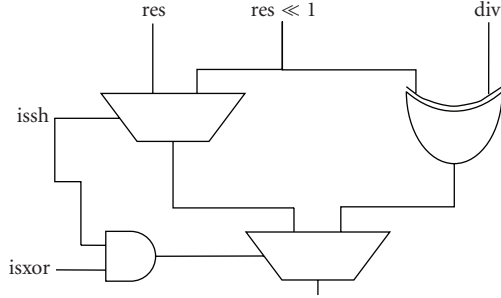


FIGURE 1: xor-select block.

- (iii) the control to select whether the xor is with the divisor or 0s are the bits of the second multiply inputs starting with the upper-most bit.

The example below multiplies 101100 and 011011 using 6 steps. 10110 is used as the control input

```

0000000000000000
0000000110110000
0000001101100000
0000000000000000
0000110110000000
0000000110110000
0001110111000000
0000000110110000
0001110101000000
0000000000000000
0011110101000000
0000000000000000
0111101010000000

```

The gfmul instruction always does 8 steps of multiply. Consequently, in the implementation, the “:divisor” is prepended by 8 zeroes.

**3.3. Results.** The unified block that implements gfnormi and gfmul in the SB3500 consists of some setup followed by 8 stages of a computational kernel. This computation in each stage is an xor-select, as shown in Figure 1.

In the case of the polynomial remainder operation, gfnormi, the result, (res), and divisor (div) values are setup from the value of the ra and rb registers. The count (N) is set to the immediate value specified in the operation + 1. For the first N of the 8 xor-shift stages, if the MSB in the res is 1, the res is shifted and xor-ed with value in div; otherwise it is just shifted by 1.

For the polynomial multiply operation, res is set to 0 and div is set to the value of the rb register prepended by 8 zeroes; the count N is always 8. The top 8 bits of the rb register are used as controls to the 8 xor-shift stage; if the corresponding bit is 1, res is shifted and xor-ed with the value in div; otherwise it is just shifted by 1.

From the diagram shown in Figure 1, it is clear that the critical path for each of the 8 xor-select stages in a naïve implementation uses two 2-1 muxes. Adding in the initial setup, this gives a total delay for the entire block of about 17 2-1 muxes. In the TSMC 65 nm low power TSMC65LP

process the critical path is about 0.9 nanoseconds at an area of  $2856 \mu\text{m}^2$  using regular Vt transistors and typical timing.

The SB3500 implemented is targeted for a 1.6 nanoseconds clock. It has a 2-stage execute pipeline, so the gf-op block is pipelined across 2 stages. This gives the synthesis tool 3.3 nanoseconds to implement the block. The synthesis tool used this relaxed timing to pick a power and area optimized implementation. In this implementation the gf-op block occupies approximately  $2018 \mu\text{m}^2$ , not including the pipeline registers.

It is possible to implement various look-ahead schemes that would reduce the critical path at the expense of extra logic. Since we have ample slack, we did not investigate any area/speed tradeoffs.

## 4. Reed-Solomon

We have implemented a RS encoder/decoder that is designed to be implemented on a SIMD architecture. The numbers presented in this section are tuned for the DVB (digital video broadcasting) standard. This standard uses RS(204,188) encoding; that is, it adds 16 check symbols to a 188-byte packet resulting in a total code word of length 204 bytes.

**4.1. Algorithm.** The RS encoder used in this study does the following steps [3]:

- (i) append N zeroes to a data block,
- (ii) perform successive Horner reduction of the polynomial whose coefficients are the data block plus zeroes to obtain remainders,
- (iii) multiply remainders by pre-calculated coefficients and sum.

All operations are over  $\text{GF}(2^8)$ .

The RS decoder [4] starts with syndrome computation, which computes the dot-product of the received code-word with precomputed syndrome vectors [4]. If the syndromes are all zero, then there are no errors.

Our implementation combines several techniques to improve the error decoding capability.

- (i) Correct codeword using Peterson-Gorenstein-Zieler (PGZ) [5] algorithm.
- (ii) If that fails to correct the errors, successively apply 2, 4, 6, 8 erasures, deriving an error locator polynomial, until an error locator polynomial of correct degree is derived [6, 7].
- (iii) If an error locator polynomial is identified, attempt to decode the word using the Forney-Messey-Berlekamp (FMB) method [8, 9].

Again, all operations are over  $\text{GF}(2^8)$ . The details of this approach have been published previously [6].

**4.2. Results.** We started off with an original version of the code that was designed to use Galois Field operations. This base code was then rewritten to use SIMD forms of polynomial-multiply and -remainder operations.

TABLE 2: Polynomial operations in RS(208,188) encoder/decoder. In the decoder, by contrast, the number of poly remainder operations is 25% of the number of poly-multiplies. This allows us to get a 11.5x speed up.

Operation		Encoder	Decoder PGZ)	Decoder FMB one iteration
Galois field multiply		3384	8864	9267
Polynomial multiply	gfmul			
	rgfmul	203	134	134
	rgfmac	16	56	0
	rgfmulred		423	529
Total		219	613	663
Polynomial remainder	gfnormi		2	2
	rgfnorm	204	152	152
	Total	204	154	154
polynomial operations		423	767	817

The experiments that were run encode one RS(204,188) packet artificially introduce enough errors to require 8 erasures and then decode the packet. Note that this is the worst-case decode situation; in practice 98% of all packets have all syndromes equal to zero, so no error decoding is required.

Table 2 gives the details of the results of our experiments. In the encoder, the number of poly-remainder operations is almost the same as the poly-multiplies. Consequently, SIMD only achieves a 8x speed up, even though the SIMD variants of the polynomial instructions perform 16 poly-operations in parallel.

**4.3. End-to-end Simulation Results.** For the DVB-T case, for the highest bitrate of 31.67 Mbps, the decoder is called 21 763 times per second. The total number of cycles spent by the processor in vector mode for the GF operations only is less than 18 MHz (a fraction of the SBX processor capabilities) compared to 277 MHz in scalar mode. The iterative decoding algorithm was tested in the end-to-end DVB-T/H simulated system, specified by ETSI EN 744 V1.4.1 (2001-01). The simulations were performed by using the SBX simulation tools. Using our GF instructions, the total number of cycles per second consumed in the SBX processor, for the highest bit rates specified in the standards and assuming that every packet has eight errors and eight erasures, is the following: 29 MHz for the 31.67 Mbps DVB-T, 9 MHz for the 4.4 Mbps DVB-H including the optional second RS decoder at the link level.

## 5. Conclusions

The method for implementing GFM we have described, that implements a GFM using 2 instructions a poly-multiply and a poly-remainder, allows the addition of GFM to a standard architecture without the need to introduce a special purpose register for the GFM. Further, both of these instructions can be implemented using the same hardware block.

We have shown that, for some applications, it is not necessary to execute both a poly-multiply and a poly-remainder for each GFM. In the cases where the results of several GFM are added together, the products of the corresponding poly-multiplies are summed and then a single poly-remainder is used. In one specific case, only 25% of the poly-multiplies required a poly-remainder. Our simulation results indicate a speedup of 11.5x of the extended processor versus the standard processor.

## References

- [1] Texas Instruments, Inc, "TMS320C64x/C64x+ DSP CPU and Instruction Set Reference Guide," February 2008.
- [2] M. Moudgill, J. Glossner, S. Agrawal, and G. Nacer, "The sandblaster 2.0 architecture and SB3500 implementation," in *Proceedings of the Software Defined Radio Technical Forum (SDR Forum '08)*, Washington, DC, USA, October 2008.
- [3] D. Iancu, J. Glossner, and M. Moudgill, "A Method of Reed-Solomon Encoding and Decoding," European Patent EP1704647.
- [4] S. G. Wilson, *Digital Modulation and Coding*, Prentice-Hall, Upper Saddle River, NJ, USA, 1996.
- [5] S. B. Wicker, *Error Control Systems for Digital Communication and Storage*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1995.
- [6] D. Iancu, M. Moudgill, J. Glossner, and J. Takala, "Efficient Reed-Solomon iterative decoder using galois field instruction set," in *Proceedings of the 8th Workshop on Embedded Computer Systems: Architectures, Modeling, and Simulation (SAMOS '08)*, vol. 5114 of *Lecture Notes in Computer Science*, pp. 126–135, Samos, Greece, July 2008.
- [7] D. Iancu, H. Ye, J. Glossner, M. J. Schulte, S. Mamidi, and J. Takala, "Improved spectral efficiency through iterative concatenated convolutional reed-solomon software decoding," in *Proceedings of the Joint 1st Workshop on Sensor Networks & Symposium on Trends in Communications (SymptoTIC '06)*, pp. 1–5, Bratislava, Slovakia, June 2006.
- [8] J. L. Massey, "Shift register synthesis and BCH decoding," *IEEE Transactions on Information Theory*, vol. 15, no. 1, pp. 122–127, 1969.
- [9] G. Forney Jr., "On decoding BCH codes," *IEEE Transaction on Information Theory*, vol. 11, no. 4, pp. 549–557, 1965.

## Research Article

# Low-Cost Transceiver Architectures for 60 GHz Ultra Wideband WLANs

**S. O. Tatu, E. Moldovan, and S. Affes**

*Laboratoire LRF, Center Énergie, Matériaux et Télécommunications, Institut National de la Recherche Scientifique, 800 de la Gauchetière Ouest, Montréal, R6900, Quebec, Canada H5A 1K6*

Correspondence should be addressed to S. O. Tatu, [tatu@emt.inrs.ca](mailto:tatu@emt.inrs.ca)

Received 23 November 2008; Accepted 2 February 2009

Recommended by Daniel Iancu

Millimeter-wave multiport transceiver architectures dedicated to 60 GHz UWB short-range communications are proposed in this paper. Multi-port circuits based on 90° hybrid couplers are intensively used for phased antenna array, millimeter-wave modulation and down-conversion, as a low-cost alternative to the conventional architecture. This allows complete integration of circuits including antennas, in planar technology, on the same substrate, improving the overall transceiver performances.

Copyright © 2009 S. O. Tatu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

## 1. Introduction

With the rapid growth of wireless technologies, ubiquitous and always-on wireless systems in homes and enterprises are expected to emerge in the near future. Facilitating these ubiquitous wireless systems is one of the ultimate goals of the next generations (4G and beyond) wireless technologies being discussed worldwide today. This increasing interest for wireless connectivity has pushed the regulatory agencies to provide new opportunities for unlicensed spectrum usage with fewer restrictions on radio parameters. In order to provide more flexibility in spectrum sharing, the FCC introduced (i) an underlay approach with severe restrictions on transmitted power levels with a requirement to operate over the occupied and the unoccupied spectrum across the 3.1–10.6 GHz band and (ii) an opening of 7 GHz unlicensed spectrum at millimeter-wave frequencies around 60 GHz, where oxygen absorption limits long-distance interference. Both spectrums are suitable for ultra-wideband (UWB) short-range wireless local area networks (WLANs) dedicated to high data rate applications, which could be high-speed home or office wireless networking and entertainment, such as high-definition television (HDTV).

Conventional microwave UWB technology (3.1–10.6 GHz band) is one of the most active focus areas in academia, industry, and regulatory circles. Because of the power spectral density limitations, (−41 dBm/MHz)

the microwave UWB overlays existing wireless services (GPS, PCS, Bluetooth, and IEEE 802.11 WLANs) without significant interferences.

Compared to the conventional UWB technology, 60 GHz mm-wave communications will operate in the currently unlicensed spectrum (57–64 GHz) and will easily provide data rates up to several Gb/s [1–3]. In the case of comparable bandwidths and data rates, an important advantage of using millimeter-wave bands is the reduced ratio between the bandwidth and the central frequency, leading the way to transceiver simplicity. However, the 60 GHz indoor channel presents a challenging environment for UWB wireless communications. Therefore, future UWB WLANs will certainly be achieved using smart antenna arrays to reduce the multipath effects together with the link budget and the power consumption. The inherent power limitations and wireless channel propagation characteristics dictate the short-range capability of 60 GHz communications. The strong signal attenuation allows an efficient reuse of the spectrum. This helps to create small indoor cells for hot spot secure wireless communications.

## 2. Fabrication Technologies

The successful integration of circuits and modules into the same substrate is the key to a significant cost reduction.



In particular, GaAs technology has been developed to the point where 60 GHz monolithic microwave integrated circuits (MMIC) are production ready [1, 3]. An alternative technology based on silicon germanium (SiGe) promises to provide low-cost millimeter-wave front-end MMICs, simultaneously maintaining the performances of GaAs [1]. While complementary metal oxide semiconductor (CMOS) technology is considered as an ideal solution in terms of cost and circuit integration, RF CMOS for 60 GHz frequency requires more performance improvement [3].

A very promising technology for small-scale integration is represented by monolithic hybrid microwave integrated circuits (MHMIC) on very thin ceramic substrate. It is to be noted that coplanar wire-bond interconnections between chips and the flip chip assembly (direct electrical connection of face-down, hence “flipped”) could be low loss at 60 GHz, whereas multichip module technologies could well accommodate millimeter-wave components along with IF and baseband circuits.

A further improvement would be the monolithic integration of antennas with MMIC or MHMIC chips in order to avoid significant interconnection losses. We note that antennas are rarely integrated into the proposed front-ends, decreasing the overall performances in terms of noise figure and gain. The integration of circuits and antennas forming a single front-end module offers several benefits, such as compactness, low-power consumption, and multifunctionality.

### 3. MultiPort Millimeter-Wave Circuits and Modules for UWB Applications

The transceivers for 60 GHz WLANs must be composed by low-cost components having excellent electrical performances, designed according to the UWB transceiver requirements.

Millimeter-wave multiport circuits and modules represent a low-cost unconventional approach.

A complete multiport quadrature down-conversion theory, validated by various simulations and measurements of a Ka-band direct conversion receiver, has already been published [4]. Simulations have also been performed for a V-band multiport heterodyne receiver [5]. The multiport technology allows improved results in terms of conversion loss and requires a reduced LO power compared to the conventional methods (as low as  $-20$  dBm to perform an efficient frequency conversion), as it has recently been demonstrated in [6]. As known, a conventional diode mixer using antiparallel diodes acting at LO-driven switches requires around  $+10$  dBm LO power for the same conversion loss. The excellent isolation between the multiport RF inputs is another important advantage versus the conventional approach. Another recent study [7] proves that the multiport mixer exhibits very good suppression of harmonic and spurious products, due to the symmetry and specific multiport properties.

Figure 1 shows the block diagram of the implementation of multiport circuit for down-conversion purposes.

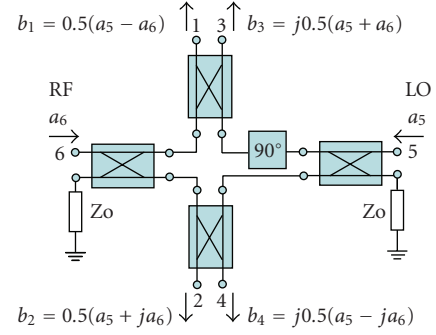


FIGURE 1: The simplified block diagram of a multiport circuit for down-conversion.

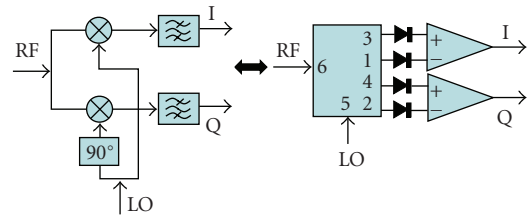


FIGURE 2: The equivalence between conventional I/Q mixer—the multiport down-converter.

The circuit uses four  $90^\circ$  hybrid couplers and a  $90^\circ$  phase-shifter. The normalized output waves are represented on the same figure. As seen, the output waves are linear combination of phase-shifted input signals. The insertion loss is equal to 6 dB because each signal passes through two 3 dB couplers.

Figure 2 shows the equivalence with conventional I/Q mixer architecture. The millimeter-wave multiport in conjunction with a local oscillator and two IF differential amplifiers (IFDA) will be used in order to obtain two quadrature IF signals, avoiding the use of two costly millimeter-wave mixers and reducing the power consumption [5]. It is to be noted that these IF circuits will realize additional filtering functions.

The millimeter-wave modulator is an essential element for the transmitter. Depending on specific modulation scheme, a multiport-based modulator or a millimeter-wave switching network are proposed for low-cost UWB applications.

The multiport modulator is composed by four  $90^\circ$  hybrid couplers, a  $90^\circ$  phase shifter and two pairs of monoports having a controllable reflection coefficient, as seen in Figure 3. The phase and the amplitude of the normalized output signal are related to the monoport return loss values. Therefore, the direct modulation of a millimeter-wave signal can be easily obtained. As an example, using short or open circuits, the return loss values are equal to  $-1$  and  $+1$ , respectively, and a direct QPSK multiport modulator is obtained [8].

In addition, “II” or “T” switch circuits using diodes (having appropriate return and insertion losses at 60 GHz, and a bandwidth of several GHz) can be designed and implemented for the transposition of the microwave UWB at 60 GHz.

Intelligent antennas take advantage of both antenna technology and propagation characteristics [9, 10]. The use of such antennas in millimeter-wave front-ends has the potential to reduce multipath interference and to increase the output signal to noise ratio.

The advantages of the electronic beam scanning have been extensively discussed in literature. The primary reason for using antenna arrays is to produce a directive beam that can be electronically repositioned (scanned). The antenna arrays consist of multiple stationary antenna elements (which are fed coherently) and can use controlled phase shifters at each element to scan a beam to given angles in space.

In order to avoid the use of controlled phase shifters, for low-cost millimeter-wave applications, the use of antenna arrays based on Butler matrices is proposed. This is a multiple beam array, each input port exciting an individual beam in space. The Butler matrix is a circuit implementation of the FFT, radiating orthogonal sets of beams with uniform aperture illumination. The proposed choice is a  $4 \times 4$  Butler matrix antenna array with four orthogonal beams, spaced by  $30^\circ$ . If more discrete beams are needed, the number of antennas and the complexity of the Butler matrix increase rapidly.

Figure 4 shows the block diagram of a four element antenna array. Its architecture is based on a  $4 \times 4$  Butler matrix having an original topology adapted to millimeter-wave frequencies, which avoids any cross-line. The four patch antennas are connected to a multiport circuit having four inputs and four outputs. This circuit is composed of four  $90^\circ$  hybrid couplers and two  $45^\circ$  phase shifters, implemented using  $\lambda/8$  transmission lines. Due to their small dimensions, the patch antennas are integrated on the same substrate.

For each multiport output signal ( $b_5$ – $b_8$ ), an individual maximum is obtained by shifting the angle of arrival in the  $180^\circ$  range. The side-lobes are at least 8 dB below the main lobe and the angles of arrival corresponding to maximum signals are around  $-45^\circ$ ,  $-15^\circ$ ,  $15^\circ$ , and  $45^\circ$ . Therefore, the main lobe of the antenna array can be shifted by  $30^\circ$  multiplies. A gain of around 10 dBi can be obtained using this multiport antenna array [11].

#### 4. Proposed Transceiver Architectures

UWB technologies using short-pulse signals have been applied to radar systems since the 1960s, and their communication applications have stimulated industry and academia interest since 1990s. A network capacity of many hundreds of Mbps may be required by some specific applications such as WLAN Bridge for interconnecting Giga-Ethernet LANs, wireless virtual reality allowing free body movements, wireless TV high-resolution recording camera, or wireless Internet download of lengthy files. It is to be noted that, in addition to communication applications, the UWB devices can be used for imaging, measurements, sensors, and vehicle radars. In principle, there are two ways to achieve such a significant network capacity: (i) by increasing the spectral efficiency and/or (ii) by using an extended bandwidth.

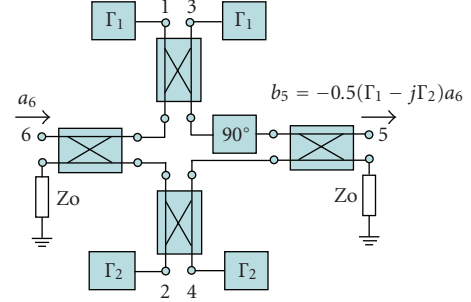


FIGURE 3: The simplified block diagram of a multiport modulator.

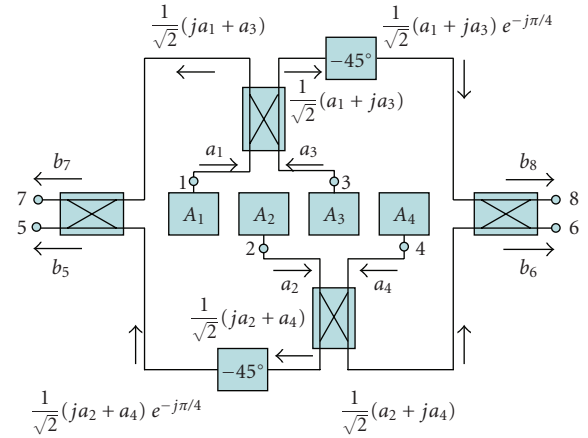


FIGURE 4: The simplified block diagram of a multiport antenna array.

According to FCC definition, the transmission bandwidth of UWB signals should be greater than 500 MHz or larger than 20% of the central frequency. This open definition does not specify any air interface or modulation for UWB. In the early stages, time-domain impulse radio (IR) has dominated UWB technology and still plays a crucial role. However, driven by the standardization activities, conventional modulation schemes, such as orthogonal frequency division multiplexing (OFDM) or single carrier (SC), have also appeared [3].

As known, the OFDM technique partitions a UWB channel to a group of nonselective narrowband channels (using a simple modulation technique such as QPSK), which makes it robust against large delay spreads, by preserving orthogonality in the frequency domain. In order to meet at least the “500 MHz requirement” of the UWB systems, two approaches can be used: (i) a high number of carriers (16, 32, 64, or 128) with a corresponding relatively low bit-rate/carrier, or (ii) a small number of carriers (2, 4, or 8) with a corresponding higher bit-rate/carrier. However, at 60 GHz, phase noise and carrier offset will degrade the OFDM performances. Due to the complexity of the OFDM architecture, in our opinion, only the second approach can be suitable for low-cost 60 GHz UWB WLANs.

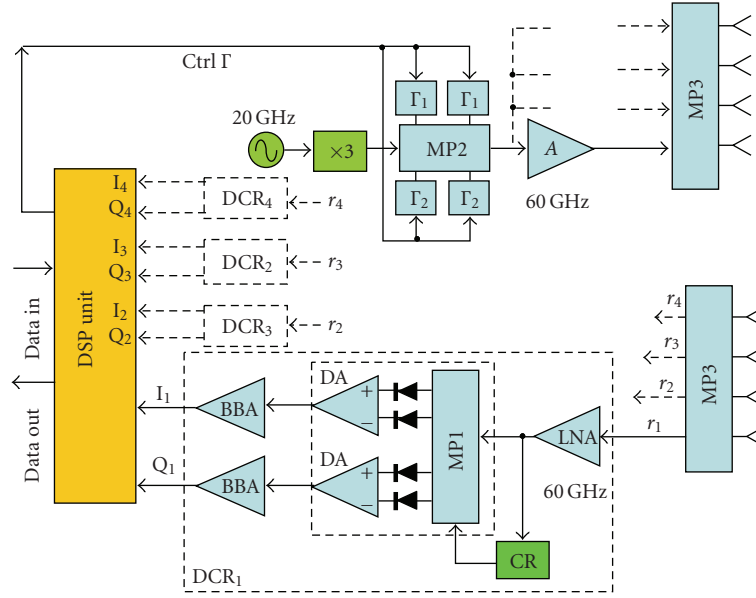


FIGURE 5: The simplified block diagram of a multiport direct conversion transceiver.

The available bandwidth, the efficient reuse of spectrum (due to strong signal attenuation at 60 GHz) makes flexibility, simplicity, and cost the most critical points. It was demonstrated that transmissions using SC advanced modulation techniques and directive antennas can achieve comparable performances with OFDM for a 60 GHz indoor channel [3]. These modulations, such as  $M$ -ary quadrature amplitude modulation (QAM) and  $M$ -ary phase shift keying (PSK), will increase the spectral efficiency. Simulations of a 60 GHz high-speed multiport heterodyne receiver have been recently published [5]. A bit-rate up to 400 Mbps has been achieved using a 16 QAM modulation for an IF of 900 MHz. The proposed architecture enables the design of compact and low-cost wireless millimeter-wave communication receivers for future high-speed WLANs, according to the IEEE 802.15.3c standard. However, the millimeter-wave circuits' bandwidth must be increased to few GHz and the IF must be at least 2.45 GHz, to cope with Gb/s bit-rates.

In order to use such a modulation, a UWB transmitter must be also designed. In the case of SC-AM, the use of the directive antenna arrays based on multiports, of the heterodyne multiport I/Q down-converter, and of the multiport modulator, is considered as very promising for indoor low-cost UWB communications.

Figure 5 shows a simplified block diagram of a millimeter-wave multiport direct conversion transceiver. A multiple input multiple output (MIMO) architecture is proposed. Two-phased array based on Butler matrices are used. This solution appears optimal because few discrete beam directions are generally sufficient in indoor WLANs. A 20 GHz microwave oscillator and a frequency multiplier generate the 60 GHz signal. The DSP unit modulates the carrier using a multiport direct modulator MP2. According to 60 GHz proposed standard, the required amplifier output

power is +10 dBm. In order to obtain the control of transmitted beams, the DSP will activate one of millimeter-wave amplifiers to feed the transmitter multiport antenna array MP3. The corresponding direct conversion receiver (DCR<sub>i</sub>) is composed by a low-noise amplifier (LNA), a carrier recovery (CR) circuit, a multiport down-converter (MP1, power detectors and differential amplifiers (DA)), and two baseband amplifiers (BBA).

The access to the inphase (I) and quadrature (Q) signals will enable significant additional capabilities, increasing the phase measurement accuracy and offering a straightforward correspondence between the baseband phasor rotation frequency and the Doppler shift if the same oscillator is used in the receiver part. The carrier recovery circuit is used as reference signal and will compensate the Doppler shift in a hardware approach.

Figure 6 shows the proposed heterodyne architecture of a multiport transceiver. Due to the increased gain of the receiver, omnidirectional antennas can be also used. An additional millimeter-wave oscillator must be used in the receiver part due to the nonzero IF. The Doppler effects and the inherent frequency shift between millimeter-wave oscillators are compensated using a PLL circuit operating at IF. The second down-conversion can use conventional means due to the microwave operating frequency. To cope with data rates of 500 Mbps, the IF of the heterodyne receiver is chosen at 900 MHz. If the data rate is increased to 1 Gb/s, the IF can be chosen at 2.45 GHz.

Direct Sequence (DS) UWB is often referred to as impulse, baseband, or zero carrier technique. It operates by sending Gaussian low-power-shaped pulses, coherently received by the receiver. In view of the fact that the system operates using pulses, the transmission spreads out over a wide bandwidth, typically many hundreds of MHz or even several GHz. To enable data to be carried, DS UWB

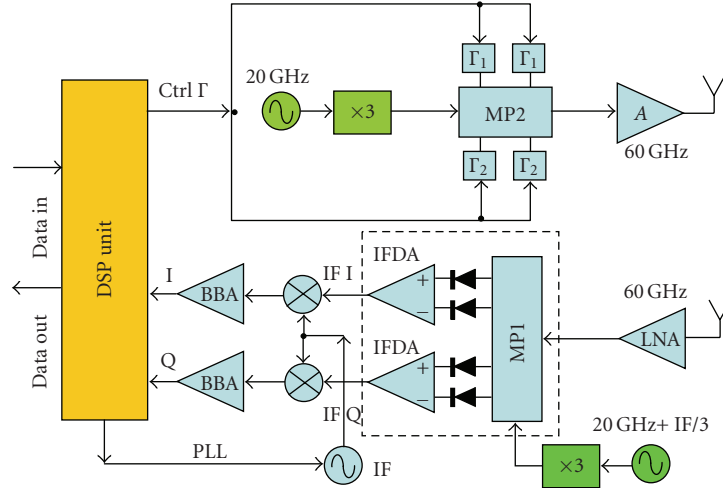


FIGURE 6: The simplified block diagram of a multiport heterodyne transceiver.

transmissions can be modulated in multiple ways. For example, pulse position modulation (PPM) encodes the information by modifying the time interval and, hence, the position of the pulses; binary phase shift keying (BPSK) reverses the phase of the pulse to signify the data to be transmitted.

Therefore, in order to use a larger bandwidth with reduced power consumption, a new method based on the transposition of impulse radio ultra wideband (IR UWB) signals at 60 GHz-band can also be taken into account [12–14].

As a low-cost 60 GHz IR UWB proposal, the transmitter part can be implemented using an oscillator, a millimeter-wave switch and an amplifier. A pulse generator (1st PG) generates subnanoseconds pulses (e.g., pulse width around 350 ps, in order to reach 3 GHz bandwidth). The 60 GHz carrier will be digitally pulse position modulated (PPM) using a millimeter-wave switch. After amplification, Gaussian pulses are emitted over several GHz bandwidth centered into the 60 GHz band. In order to implement the receiver, either a mixer or a detector can be used. If a mixer is used, a millimeter-wave oscillator is needed. The mixer can be implemented using the low-cost, low-power consumption multiport down-converter. The oscillator is not required when a topology with a detector is chosen, as presented in Figure 7. Therefore, the receiver is composed of three main modules: a low-noise amplifier, a 60 GHz detector, and a correlator. A pulse generator (2nd PG) is used to control the sample and hold (S/H) circuit. The main advantage of this architecture is that no phase information is needed, and thus, no sophisticated coherent stable sources or carrier recovery circuits are involved. It is to be noted that the pulses can also be modulated using the BPSK with minimal architectural changes in both transmitter and receiver. In order to transmit data information, instead of modifying the position of pulses as for PPM, the phase of subnanosecond pulses will be reversed at the transmitter. The receiver must be able to observe these 180° phase changes; therefore, a multiport based phase-detector can be successfully used. It

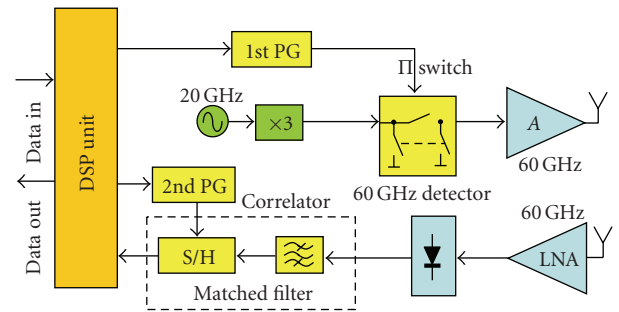


FIGURE 7: The simplified block diagram of an impulse-radio transceiver.

is to be noted that in the block diagrams, the DSP include required A/D and D/A converters.

Transmission on ranges up to 10 meters can be expected, due to the high free-space loss at the carrier frequency. The Friis path loss equation shows that, for equal antenna gains, path loss increases with the square of the carrier frequency. Therefore, 60 GHz communications have an additional 22 dB of path loss when compared to an equivalent 5 GHz system. However, antenna dimensions are inversely proportional to carrier frequency. Therefore, more antennas can be placed within a fixed area and the resulting antenna array will improve the overall antenna gain. The directive antenna pattern of a beam forming antenna array improves the channel multipath profile by limiting the spatial extent of the transmitting and receiving antenna patterns to the dominant transmission path. This aspect opens up new opportunities for wireless system design. The use of smart antennas will also improve the link budget and will reduce the transmitter power.

A consequence of the confinement to smaller cells is that the channel dispersion is smaller than the values encountered at lower frequencies because echo paths are shorter on average. However, movements of the portable stations, as well as the movement of objects in the environment, cause

Doppler effects, relatively severe at 60 GHz, because they are proportional to the carrier frequency. For example, if persons move at a walking speed of 1.5 mps, the Doppler spread result is 1200 Hz.

Simulated BER results are excellent. The BER values are less than  $10^{-5}$  for energy per bit to noise power spectral density (Eb/No) ratio of 10 dB [11]. These results prove that simple multiport architectures can be suitable for low-cost millimeter-wave transceivers for future 60 GHz WLANs.

## 5. Conclusion

Several millimeter-wave multiport transceiver architectures have been presented in this paper. The millimeter-wave frequency conversion is obtained using the proposed I/Q multiport down-converter, avoiding the use of a costly active mixer or a high-power millimeter-wave LO (in the case of the conventional diode mixers). A multiport direct modulator or a millimeter-wave switching network can be used to modulate the millimeter-wave carrier. In addition, an antenna array based on a multiport circuit provides four output signals corresponding to four optimal directions of arrival.

The proposed multiport architectures enable the design of compact and low-cost wireless millimeter-wave transceivers for future UWB wireless communication systems.

## Acknowledgments

The financial support of the National Science Engineering Research Council (NSERC) of Canada is gratefully accepted.

## References

- [1] P. Smulders, "Exploiting the 60 GHz band for local wireless multimedia access: prospects and future directions," *IEEE Communications Magazine*, vol. 40, no. 1, pp. 140–147, 2002.
- [2] D. Cabric, M. S. W. Chen, D. A. Sobel, S. Wang, J. Yang, and R. W. Brodersen, "Novel radio architectures for UWB, 60 GHz, and cognitive wireless systems," *EURASIP Journal on Wireless Communications and Networking*, vol. 2006, Article ID 17957, 18 pages, 2006.
- [3] C. Park and T. S. Rappaport, "Short-range wireless communications for next-generation networks: UWB 60 GHz millimeter-wave WPAN, and ZigBee," *IEEE Wireless Communications*, vol. 14, no. 4, pp. 70–78, 2007.
- [4] S. O. Tatu, E. Moldovan, K. Wu, R. G. Bosisio, and T. A. Denidni, "Ka-band analog front-end for software-defined direct conversion receiver," *IEEE Transactions on Microwave Theory and Techniques*, vol. 53, no. 9, pp. 2768–2776, 2005.
- [5] S. O. Tatu and E. Moldovan, "V-band multiport heterodyne receiver for high-speed communication systems," *EURASIP Journal on Wireless Communications and Networking*, vol. 2007, Article ID 34358, 7 pages, 2007.
- [6] D. Hammou, N. Khaddaj Mallat, E. Moldovan, S. Affes, K. Wu, and S. O. Tatu, "V-band six-port down-conversion techniques," in *Proceedings of the International Symposium on Signals, Systems and Electronics (ISSSE '07)*, pp. 379–382, Montreal, Canada, July-August 2007.
- [7] B. Boukari, E. Moldovan, S. Affes, K. Wu, R. G. Bosisio, and S. O. Tatu, "A low-cost millimeter-wave six-port double-balanced mixer," in *Proceedings of the International Symposium on Signals, Systems and Electronics (ISSSE '07)*, pp. 513–516, Montreal, Canada, July-August 2007.
- [8] R. G. Bosisio, Y. Y. Zhao, X. Y. Xu, et al., "New wave radio," *IEEE Microwave Magazine*, vol. 9, no. 1, pp. 89–100, 2008.
- [9] K. M. K. H. Leong and T. Itoh, "Advanced and intelligent RF front-end technology," in *Proceedings of the IEEE Topical Conference on Wireless Communication Technology (WCT '03)*, pp. 190–193, Honolulu, Hawaii, USA, October 2003.
- [10] J.-Y. Park, S.-S. Jeon, Y. Wang, and T. Itoh, "Integrated antenna with direct conversion circuitry for broad-band millimeter-wave communications," *IEEE Transactions on Microwave Theory and Techniques*, vol. 51, no. 5, pp. 1482–1488, 2003.
- [11] E. Moldovan, S. O. Tatu, and S. Affes, "A 60 GHz multi-port front-end architecture with integrated phased antenna array," *Microwave and Optical Technology Letters*, vol. 50, no. 5, pp. 1371–1376, 2008.
- [12] M. Devulder, N. Deparis, I. Telliez, et al., "60 GHz UWB transmitter for use in WLAN communication," in *Proceedings of the International Symposium on Signals, Systems and Electronics (ISSSE '07)*, pp. 371–374, Montreal, Canada, July-August 2007.
- [13] N. Deparis, A. Boé, C. Loyez, N. Rolland, and P.-A. Rolland, "UWB-IR transceiver for millimeter wave WLAN," in *Proceedings of the 32nd Annual Conference on IEEE Industrial Electronics (IECON '06)*, pp. 4785–4789, Paris, France, November 2006.
- [14] N. Deparis, A. Bendjaballah, A. Boe, et al., "Transposition of a baseband UWB signal at 60 GHz for high data rate indoor WLAN," *IEEE Microwave and Wireless Components Letters*, vol. 15, no. 10, pp. 609–611, 2005.



## Research Article

# Multiband Antennas for SDR Applications

**E. Surducan,<sup>1,2</sup> V. Surducan,<sup>1,2</sup> D. Iancu,<sup>1,3</sup> and J. Glossner<sup>1</sup>**

<sup>1</sup> Sandbridge Technologies, 120 White Plains Road, 4th floor Tarrytown, NY 10591, USA

<sup>2</sup> Department of Molecular and Biomolecular Physics, National Institute for Research and Development of Isotopic and Molecular Technologies, 65-103 Donath Street, 400293 Cluj Napoca, Romania

<sup>3</sup> Department of Computer Systems, Tampere University of Technology, Korkeakoulunkatu, 3, 33720 Tampere, Finland

Correspondence should be addressed to E. Surducan, esurducan@gmail.com

Received 30 September 2008; Accepted 12 February 2009

Recommended by Mihai Sima

We present multiband antennas configurations for SDR applications. Using a composite folded dipole structure as starting point, we derived more complex antenna configurations to support multiple communication protocols for mobile application with linear and circular polarizations. Prototypes as single antenna with circular polarization, tunable single antenna with PIN diode and MIMO systems with three and four antennas, all derivatives of the same basic structure, were produced in an iterative fashion until the desired parameters were achieved. These antennas are suitable for microstrip circuit realizations and can be included in the printed circuit board (PCB) of the device, or used as stand alone. The shapes and measurement results are presented throughout the paper. From the illustrated graphs it can be seen that the stand-alone antennas exhibit positive gain for all the frequency bands of interest while the separation between antennas, for the multiple-input multiple-output (MIMO) case, is better than 15 dB.

Copyright © 2009 E. Surducan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

## 1. Introduction

To comply with the requirements of portable multiple-communication protocol enabled hand-held devices, a multiband antenna solution is highly desired. Most of the existing antennas used in current hand-held devices have negative gain of several dBs while base-band processing struggles to recover the last tenth of dB gain through more complex algorithms. However, considering antenna performance requirements and the tradeoffs associated with operation over multiple disparate frequency bands, constraints on size and limitations of commercial low-cost materials, typically required in handset designs, this task becomes very challenging and highly difficult. The multiband antennas configurations presented in this paper are part of the SDR platform developed by Sandbridge Technologies [1].

Ultra-wideband antennas and diversity techniques were intensively investigated and presented in the literature with not only the goal of optimizing the gain and bandwidth [2], but also by optimizing the number and position of antennas in stationary as well as in mobile devices. Multiple-input multiple-output (MIMO) antenna systems and diversity

techniques are used successfully in today's wireless communications as part of the effort of increasing the overall data throughput and link reliability [3, 4]. However, for space diversity, one requirement is to place the antennas at least one half wavelength apart. This restriction raises significant challenges when two or more antennas are to be placed on today's hand-held devices. Released and future release standards such as high-speed packed access (HSPA), long-term evolution (LTE), and worldwide interoperability microwave access (WiMAX), all specify open and/or close loop MIMO systems.

In our approach, we started the antenna design with a modified printed dipole antenna as miniature dipoles with split arms, targeting wireless local area network (WLAN) at 2.45 and 5 GHz frequencies bands [5–7]. A dual mode microstrip antenna [8] using a switch for selecting one of the two antennas is described in [9]. To avoid the extra components cost and the associated power consumption, in our solution, we choose two dipoles working on different frequencies and fed by microstrip lines at the same location. This configuration looks like miniature microstrip dipoles with split arms. We applied this technique first to the digital video broadcasting terrestrial/hand-held (DVB-T/H)

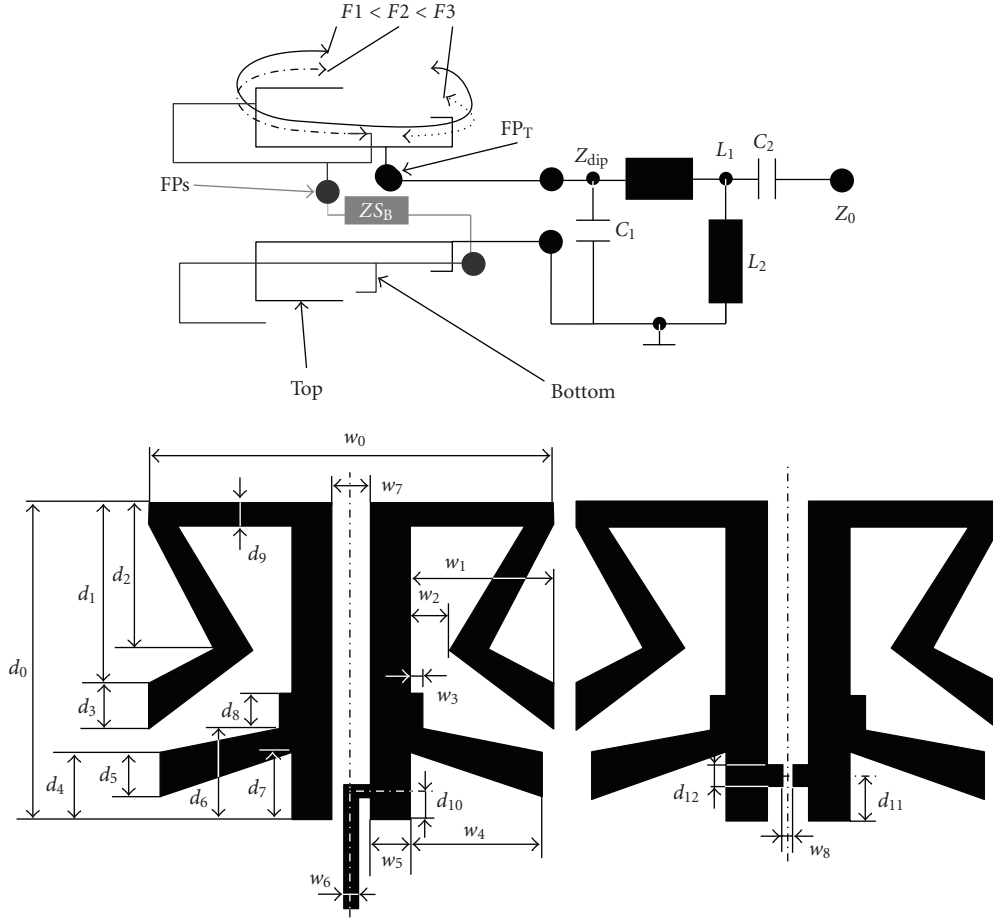


FIGURE 1: Schematic of the basic antenna configurations.  $L_1$  and  $C_1$  are distributed values due to the PCB and  $L_2$ ,  $C_2$ , and  $Z_{S_B}$  are surface mount components;  $Z_0$  is the line impedance (50 Ohm) and  $Z_{dip} \sim 200$  Ohm. The  $d_i$ ) and  $w_i$ ) dimensions figured on the PCB top and bottom side are specific to the particular antenna design.

frequency bands by using a composite structure, miniature directive antenna with high gain as shown in [10, 11], similar to metamaterial dielectric support antennas [12, 13] as a solution to enlarge the frequency bands. As a second approach, similar to a Yagi antenna applied to microstrip structures [14], we used a selective reflector placed behind the folded dipole on multilayer dielectric substrate in order to enlarge the frequency bands. We have redesigned the composite antenna for multiple communication standard hand-held devices, supporting frequency bands specified by DVB-H, GSM, and WCDMA and we incorporated the antenna to the printed circuit board (PCB) of the device [15]. Early efforts to implement MIMO antenna systems are described in automotive [2], and hand-held devices [10, 11, 16]. Our form factor WLAN and WiMAX antenna design, with two receive and one transmit antennas, is described in [17].

A common solution to tune the multiband antenna is the PIN diode switching technique [18, 19]. Our first effort in adding a PIN diode to our antenna is presented in [20]. The antenna polarization can also be controlled by PIN diodes, as shown later the paper, servicing this way the GPS L2 band.

The paper is structured as follows. Section 2 describes the approach we took for the antennas design, the measurement approach is shown in Section 3 while Section 4 presents practical realization and measurement results for different antennas. Section 5 concludes the paper.

## 2. Antenna Design

The new antenna configurations are based on our previous work on multiband composite antennas supporting DVB and WLAN frequency bands. The basic idea is to start with the composite antenna and add a selective reflector in the near field location of the active antenna (active in the sense of main radiator not in the sense of active devices on the layer). Constructively, it is an antenna with two conductive layers of similar printed patterns, top and bottom, and a composite dielectric substrate of one or more dielectric layers with different permittivity [10, 11]. On the top side is placed the matching circuit to the microwave line. The equivalent schematic of the matching circuit, and the top and bottom side PCB of the antenna (with related dimensions) are illustrated in Figure 1.  $L_1$  and  $C_1$  are distributed values

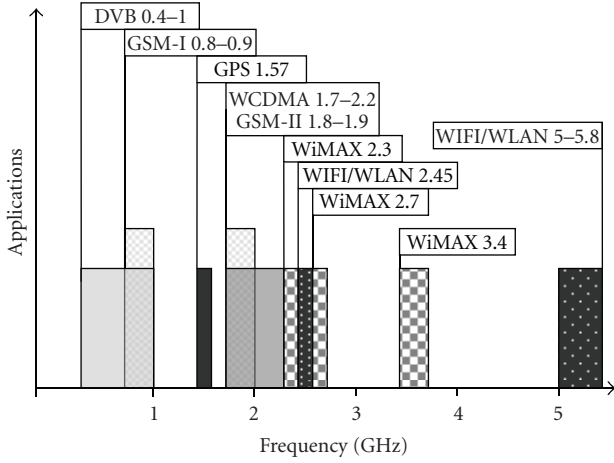


FIGURE 2: Frequency bands covered with composite antennas.

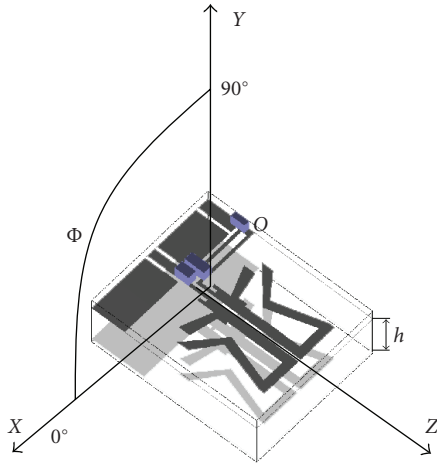


FIGURE 3: The XOY plan where the directive gain was measured for the experimental antenna.

due to the PCB,  $L_2$ ,  $C_2$ , and  $ZS_B$  are lumped surface mount components. The line impedance  $Z_0$  and the dipole impedance  $Z_{dip}$  are 50 and 200 Ohms, respectively.

In fact, the antenna is an inductive multifolded dipole structure [5], exhibiting multiple resonant frequencies  $F1 < F2 < F3$ , and having a common feed point located at zero voltage and maximum current. In Figure 1, the feed point on the top side is  $FP_T$  and the related feed point on the bottom side is  $FP_B$ . The load impedance at the feed point on the reflector side  $FP_B$  may be short or open circuit or, an impedance load,  $ZS_B$ . The position ( $d_{10}$ ) of the feed on the top side  $FP_T$  is crucial for the multiresonance frequencies of the antenna. The electrical properties of the composite dielectric between the two conductive layers are directly related to the gain as well as the width of the frequency bands covered by the antenna. With this antenna configuration it is possible to cover all frequency bands required by a specific application, with little change of the basic pattern. A large variety of antennas have been designed with dimensions ranging from 10 to 25 mm width, 20 to 50 mm length,

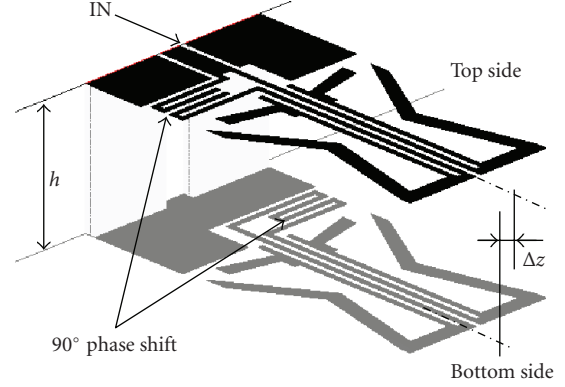


FIGURE 4: GPS omnidirectional antenna.

and PCB thickness from 0.8 to 2.54 mm. The dielectric permittivity  $\epsilon_r$  was also varied from 2.8 to 10. In the following we present only the most representatives of the produced antennas. The range of the frequency bands for different communication standards is illustrated in Figure 2.

### 3. Measurements

The most important parameters characteristic to antennas are the return loss ( $S_{11}$ ) or the standing wave ratio (SWR), gain and directivity. All measurements were performed using the Agilent N5230A (PNA-L) Vector Network Analyzer (VNA). The  $S_{11}$  (or SWR) was measured directly with the VNA. Between  $S_{11}$  and SWR there is the following well known conversion formula [8]:

$$SWR = \frac{1 + |S_{11}|}{1 - |S_{11}|}. \quad (1)$$

For a reasonably good antenna the SWR must be less than 3.

For the gain calculation, the Friis transmission equation was considered with the far-field condition as described in [21, 22]:

$$P_R = \left( \frac{\lambda}{4\pi R} \right)^2 G^2 P_T, \quad \text{with } R \geq \frac{2d^2}{\lambda}. \quad (2)$$

In the above equation,  $P_R$  is the power measured at the receive antenna output port,  $P_T$  is the power measured at the transmit antenna input port,  $G$  is the gain for both transmit and receive antennas, considered identical,  $\lambda$  is the wavelength,  $R$  is the separation between antennas, and  $d$  is the largest physical dimension of the antenna. From (2), knowing that  $S_{12} = S_{21}$ , through simple algebraic manipulation, the gain follows:

$$G = -3.779 + 10 \log(RF) + 0.5(S_{21}) + 2.15, \quad (3)$$

where  $G$  is measured in dBi, the transmission parameter  $S_{12}$  in dB,  $R$  in cm, and  $F$  in GHz.

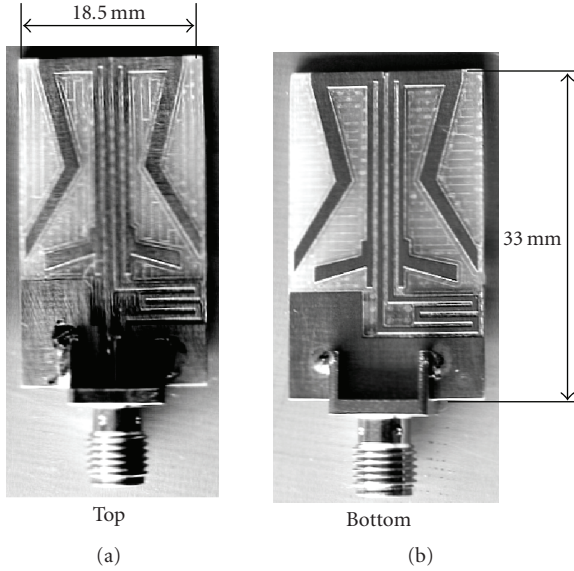


FIGURE 5: Pictures of GPS antenna, top and bottom sides.

The directivity  $D$  is the measure of the directional properties of the antenna related to the isotropic antenna. The directivity gain  $G$ , related to the isotropic dipole is also defined in [8] as

$$G = eD, \quad \text{with } 0 < e < 1, \quad (4)$$

where  $e$  is the radiation efficiency of the antenna.

The directive gain was calculated with relation (3). The  $S_{21}$  parameter was measured using two identical antennas or an antennas system. To obtain the directive gain diagram one antenna was rotated around  $OZ$  direction as shown in Figure 3. In the case of the antennas system the isolation between antennas was measured directly with the VNA through the transmission parameter the between two antennas considering that  $S_{ij} = S_{ji}$ .

#### 4. Antennas

**4.1. GPS Antenna.** The GPS satellites transmit right-hand circularly polarized (RHCP)  $L$ -band signal known as  $L1$  at 1575.42 MHz. The minimum signal power level is  $-160$  dBw at the Earth's surface. The very low power level of the GPS signal requires passive antennas of a few dB gains in the immediate proximity of the receiver or, an active antenna of minimum 15 dB gain if the antenna is placed on the roof of a car, building, or boat. To achieve circular polarization from our antenna configuration, we introduced a  $90^\circ$  delay line between the two branches of the folded dipole in both sides, top and bottom, as shown in Figure 4.

The top feed is through a microstrip while the bottom feed is inductive. A shift of  $\Delta z$  between the top and bottom layer was added, as shown in Figure 4, to compensate for the phase shift between the two sides.

We built two-prototype antennas, first on RT-6010LM substrate: 2.45 mm thick, 15 mm wide, 27 mm high, permittivity  $\epsilon_r = 10.7$ , and dissipation factor  $\tan \delta = 0.0028$ , and the

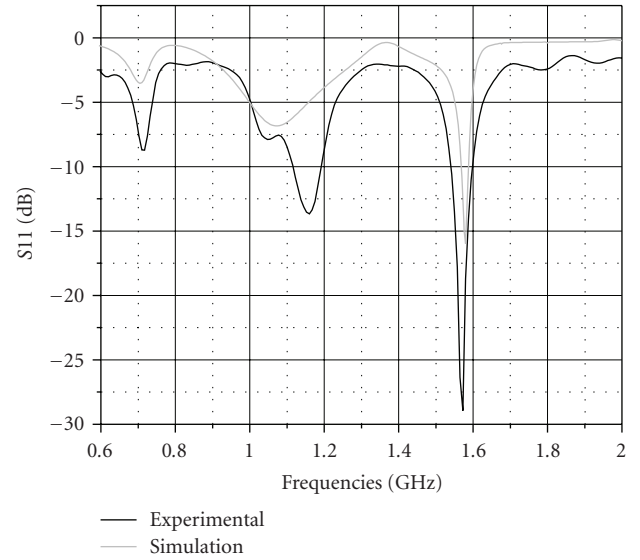


FIGURE 6: Simulation and experiment for the GPS antenna return loss.

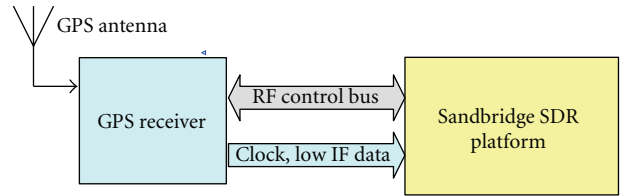


FIGURE 7: GPS test platform.

second on common PCB substrate: 1.5 mm thick, 18.5 mm wide, 33 mm high, and  $\epsilon_r = 4.6$ . For the antenna shown in Figure 5 the values of  $(d_i)$  and  $(w_i)$ , measured in mm, (as shown in Figure 1) are presented in Table 1.

We tested the antenna performance first in the lab. After the lab performance test the overall performance was evaluated on real GPS receiver. Figure 6 illustrates both the simulation and test results for the PCB substrate GPS antenna. From Figure 6 it can be seen that the return loss  $S_{11}$  is  $-25.77$  dB at 1.5753 GHz. The measured bandwidth at 1.5753 GHz is around 4 MHz while at  $S_{11} = -10$  dB the bandwidth is 67 MHz.

The field test for the antenna was performed on the Sandbridge Technology SDR platform as described in Figure 7. The total sensitivity of the GPS receiver with the passive antenna was estimated at  $-132$  dBm [23] while adding a 27 dB signal booster right after the antenna, the total sensitivity came down to  $-159$  dBm.

**4.2. PIN Diode Multiband Antenna.** The composite folded dipole antenna can be electronically tuned in frequency by adding a PIN diode to the matching circuit on the active layer as shown in Figure 8. For the prototype antenna we used the same RT-6010LM substrate and physical dimensions  $17 \times 22 \times 2.45$  mm. The antenna configuration is presented in Figure 8.

TABLE 1:  $d_i$  and  $w_i$  values, measured in mm, for GPS antenna.

$i$	0	1	2	3	4	5	6	7	8	9	10	11	12
$d_i$	22	17.6	10.6	2.0	2.0	1.5	3.1	2.0	1.5	1.5	20.8	20.8	1.0
$w_i$	15	7.1	1.4	0.5	5.8	0.5	0.5	1.5	2.5	—	—	—	—

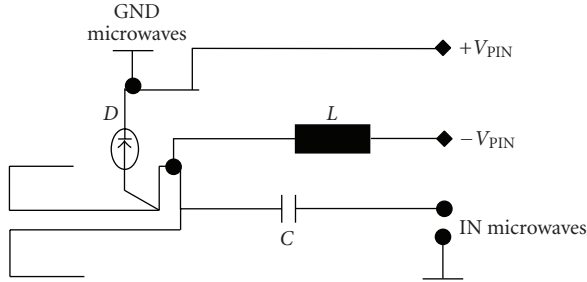
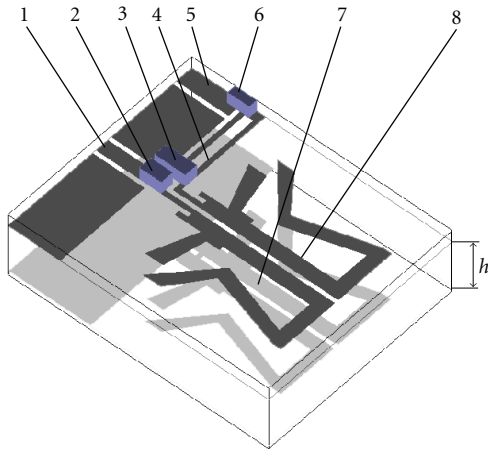
FIGURE 8: PIN diode antenna diagram.  $D$  is the PIN diode,  $L = 0.2$  mH, and  $C = 40$  pF.

FIGURE 9: Antenna configuration with PIN diode.

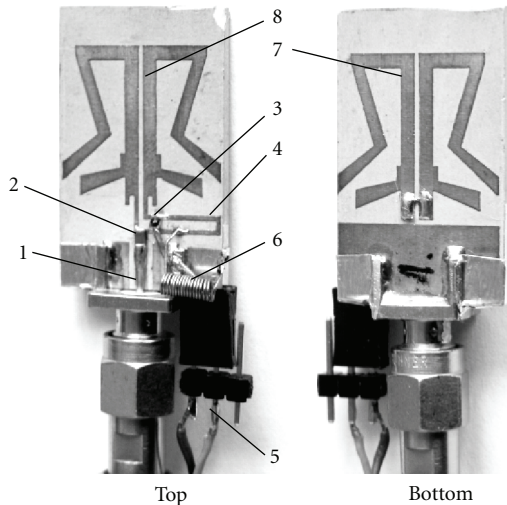
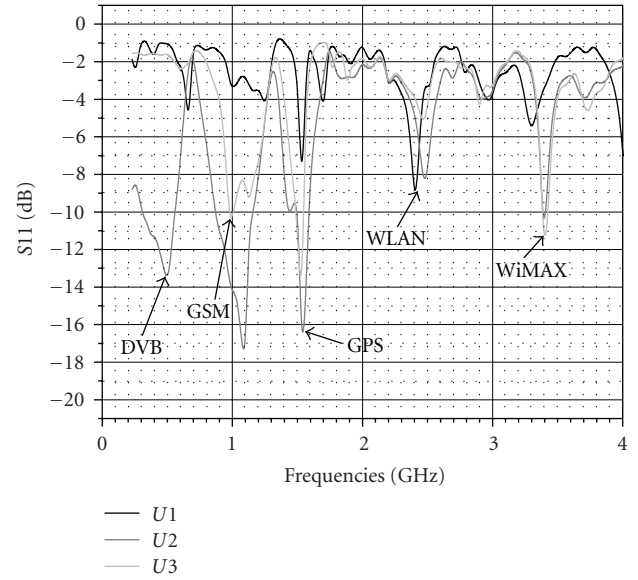


FIGURE 10: Picture of the PIN diode antenna, top and bottom.

FIGURE 11: Return loss measurement  $U1 = 0.0$  V,  $U2 = 0.65$  V,  $U3 = 2.00$  V are the PIN diode polarization voltages.

The input RF line (1) is DC decoupled (2) from the antenna top layer (8). The two branches of the dipole are phase shifted by a delay line (4). The PIN diode (3) does the impedance matching, frequency band selection, and it also changes the dipole polarization. The diode is tuned through the DC line (5) and the inductance (6). The bottom layer (7) of the antenna is totally separated from the top by the dielectric material of thickness  $h$  as shown in Figure 9. A picture of the antenna with the PIN-DC connector is presented in Figure 10.

For the antenna shown in Figure 10 the values of ( $d_i$ ) and ( $w_i$ ), measured in mm, (as shown in Figure 1) are presented in Table 2.

For this design configuration the return loss measurement  $S_{11}$  is the main feedback parameter. Figure 11 depicts the return loss of the antenna as a function of frequency for different PIN diode polarizations,  $U1 < U2 < U3$ , where  $U1 = 0$  V,  $U2 = 0.65$  V, and  $U3 = 2.00$  V. It can be seen that the return loss for different frequency bands varies with the PIN diode polarization. This property will be exploited to tune the antenna for various frequency bands required by specific communication standards.

The next diagrams illustrate the measurements performed to characterize the electric field pattern as a function of the PIN diode polarization in the frequency bands of interest. The directive gain of the antenna was measured in the XOY plan, normal to the antenna layers, as shown in Figure 12.



TABLE 2:  $d_i$  and  $w_i$  values, measured in mm, for the antenna with PIN diode.

$i$	0	1	2	3	4	5	6	7	8	9	10	11	12
$d_i$	17	12.8	9.5	1.9	2.9	2.0	5.5	3.0	1.5	1.4	1.0	1.25	1.0
$w_i$	14.8	7.1	1.4	0.5	5.8	0.5	0.5	1.5	2.5	—	—	—	—

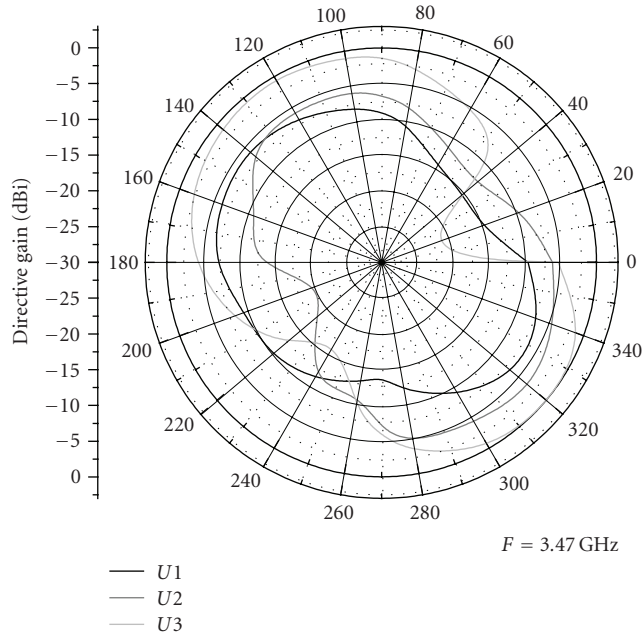


FIGURE 12: Directive gain for WiMAX at 3.47 GHz.

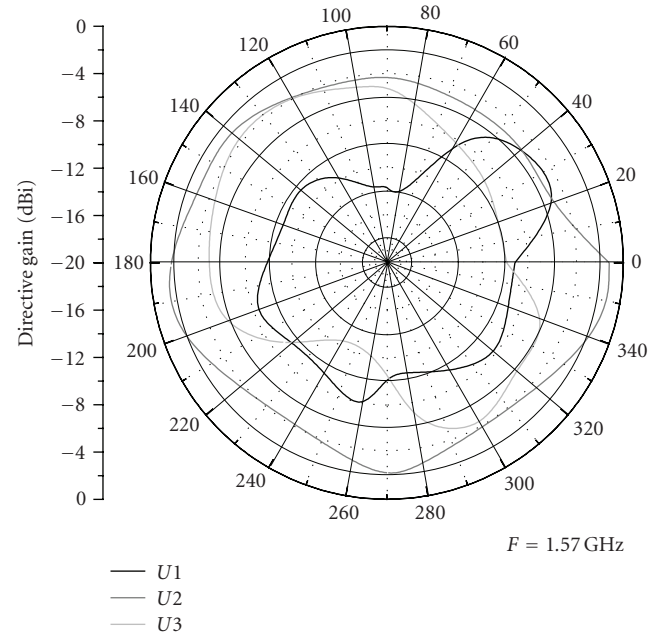


FIGURE 14: Directive gain for GPS at 1.57 GHz.

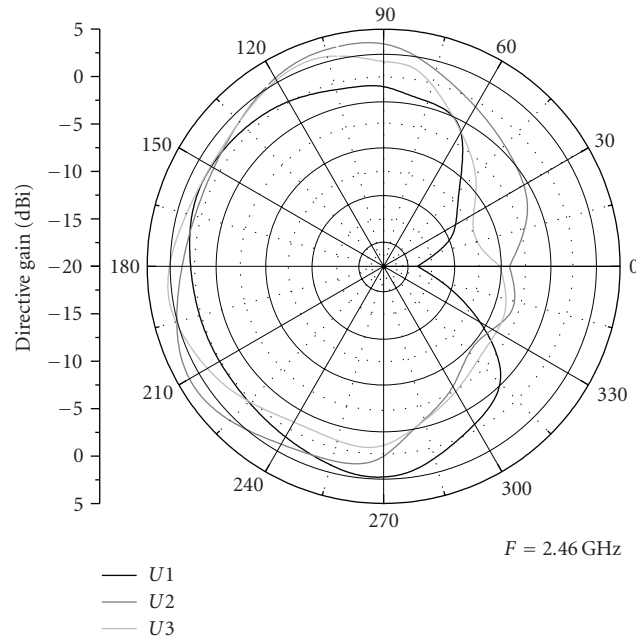


FIGURE 13: Directive gain for WLAN at 2.45 GHz.

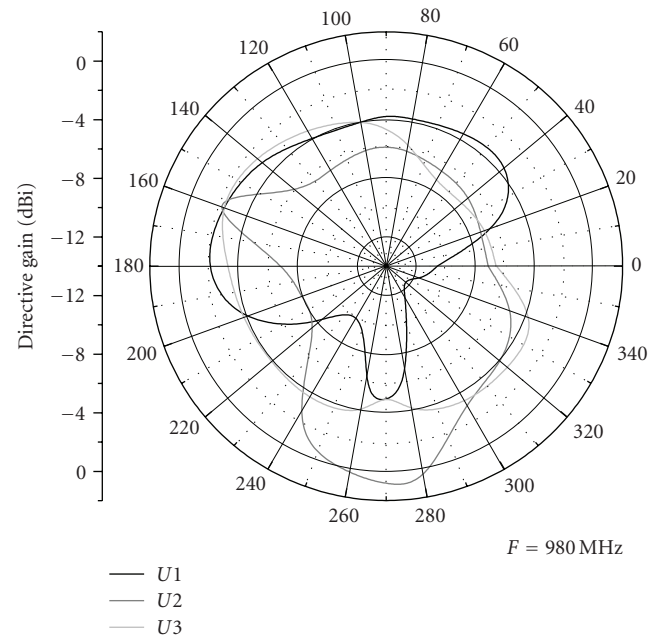


FIGURE 15: Directive gain for GSM at 980 MHz.

In the WiMAX frequency band, Figure 12 shows a gain improvement of 6 dB corresponding to the switching of the

PIN polarization from  $U1$  to  $U3$ . The main radiative direction remains unchanged at  $135^\circ$  ( $45^\circ$  to the  $OY$  normal).

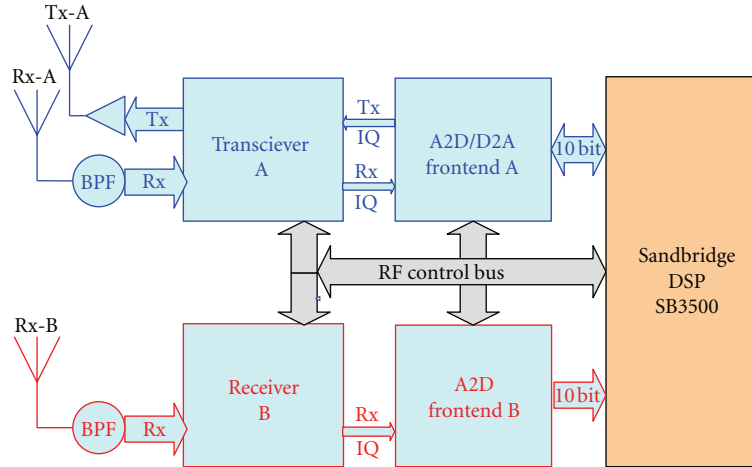


FIGURE 16: SDR platform using three antennas.

In the WLAN frequency band, the PIN diode polarization, switched from  $U1$  to  $U2$ , caused an increase in the normal direction gain ( $\Phi = 90^\circ$ ) from  $-0.9$  dBi to  $3.7$  dBi as shown in Figure 13.

In the GPS frequency band the changes with the PIN polarization voltage are more dramatic. As the polarization voltage changes from  $U1$  to  $U2$  the directivity pattern will change from directional at  $35^\circ$  to omnidirectional. The gain rests unchanged at  $35^\circ$ , but increases by an average of  $5$  dB in the other directions (Figure 14). If the voltage is changed from  $U1$  to  $U3$  the directional pattern stays unchanged but the radiation direction will change with  $90^\circ$  from  $35^\circ$  to  $125^\circ$  with a gain improvement of  $1.5$  dB.

In the GSM band, a gain improvement of  $6$  dB at  $270^\circ$  is observed when the polarization changes from  $U1$  to  $U2$  (practically a reverse in the radiation direction). Also a change from directional (at  $U1$ ) to omnidirectional pattern (at  $U3$ ) can be observed (see Figure 15).

**4.3. MIMO Systems.** Multiple folded dipole composite antenna can be placed on the same PCB to give way to a MIMO system. Three such antennas, two on the receive side and one on the transmit side were used on the Sandbridge Technology form factor SDR platform capable of executing both WiMAX and WLAN [17].

The block diagram of the SDR platform is illustrated in Figure 16 while the picture of the card with the antennas is shown in Figure 17. The TOP layer includes the microstrip feeding point and the impedance matching circuit of each antenna in the system to  $50$  Ohms line impedance. The bottom layer acts as frequency selective reflector, having similar print pattern as the top layer. In order to reduce the mutual coupling between antennas, a free space cut of  $2.5$  mm was added to the dielectric substrate near the sides of the middle antenna.

The antennas in the MIMO system were positioned such a way to maximize the spatial diversity. In order to increase the isolation between antennas, the middle antenna in the

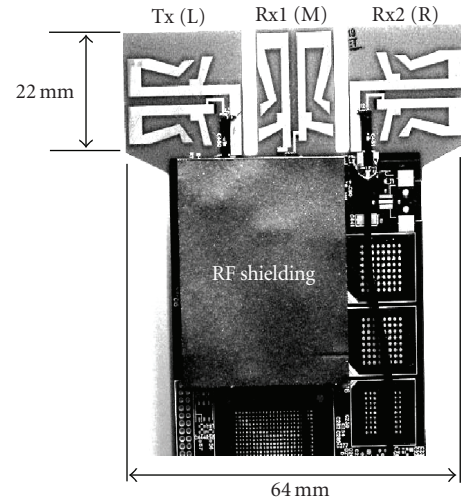


FIGURE 17: Picture of the SDR platform with three antennas.

system was positioned to have the polarization perpendicular to the lateral antennas polarization.

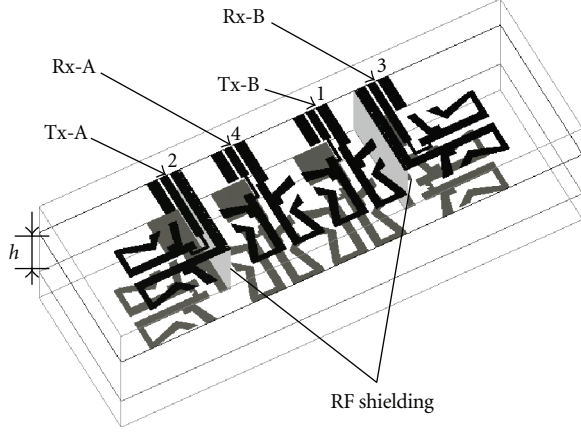
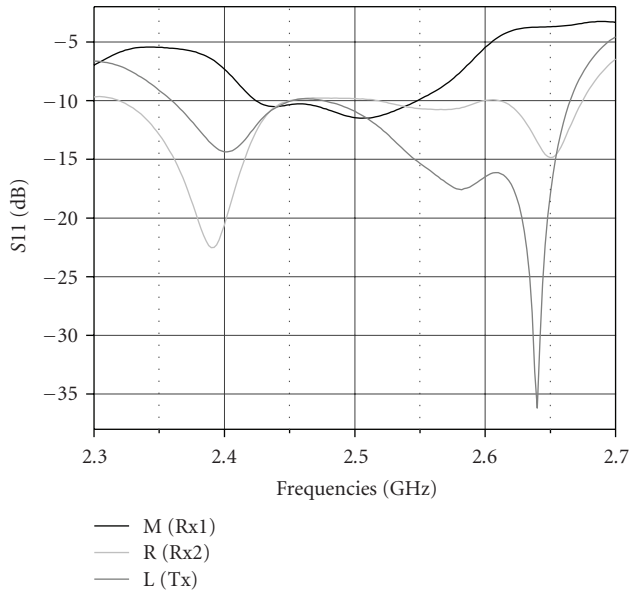
The antennas system was built on 12 layers on Isola FR 406BC PCB with  $1.2$  mm total thickness, global permittivity  $\epsilon_r = 3.8$  and dissipation factor less than  $tg\delta = 0.0140$ . The PCB gap between antennas is  $2.5$  mm wide.

For the MIMO antennas system described in Figure 17, the values of  $(d_i)$  and  $(w_i)$ , measured in mm, (as shown in Figure 1) are presented in Table 3.  $Z_{SB}$  is open circuit for Rx1 antenna and short circuit ( $Z_{SB} = 0$  Ohm) for Tx and Rx2 antennas.

A  $2 \times 2$  MIMO system, capable of both FDD and TDD communication modes, can be achieved by adding the forth antenna, as shown in Figure 18. The four-antenna system is an improved version of the previous three-antenna design, where Rx1 antenna has been symmetrically duplicated. In this case the complexity of the design will increase due to mutual coupling between the antennas. To reduce the mutual coupling a metal shield is added as shown in Figure 18.

TABLE 3:  $d_i$  and  $w_i$  values, measured in mm, for the three antennas system.

$i$	0	1	2	3	4	5	6	7	8	9	10	11	12
$d_i$	17	11.0	10.5	1.2	2.8	2.0	3.8	2.4	1.5	2.4	1.25	0.7	1.0
$w_i$	14	4.5	1.5	1.0	5.0	1.5	0.5	1.5	0.5	—	—	—	—

FIGURE 18: Four antennas in  $2 \times 2$  MIMO configuration.FIGURE 19: Return loss  $S_{11}$  measurements for MIMO system.

Two types of measurement were performed for the MIMO antennas system characterizations: return loss  $S_{11}$  and isolation  $S_{ij}$  between antennas. The return loss measurements are illustrated in Figure 19 for the MIMO system of three antennas.

In Figure 19 it can be seen that the return loss  $S_{11}$  for the M and R (Figure 17) antennas is less than  $-5$  dB in the 2.3–2.6 GHz band. The best antenna in the system is the Tx (L in Figure 17) antenna with  $S_{11} > -10$  dB in the 2.3–2.67 GHz band. In the WLAN band the standing wave ratio is less than  $-10$  dB for all antennas.

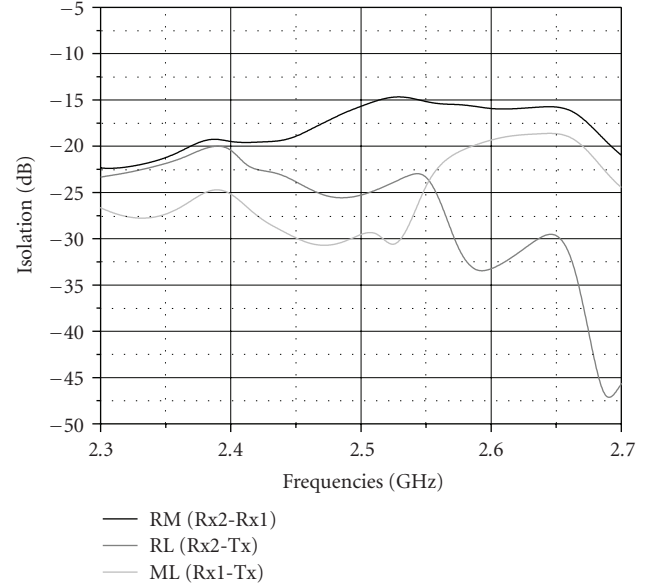


FIGURE 20: Isolation between antennas for the three-antenna configuration.

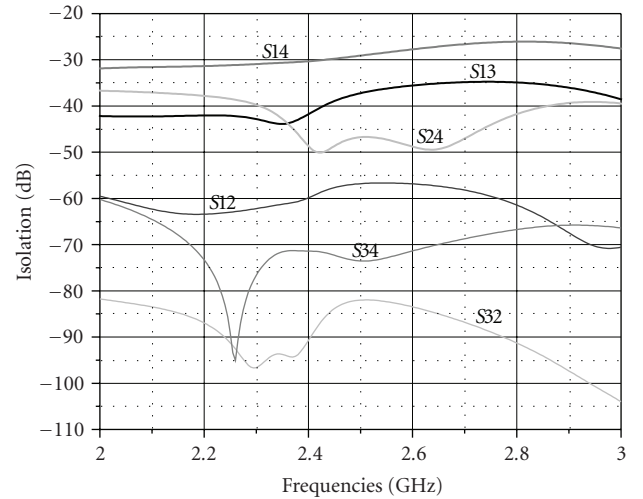


FIGURE 21: Isolation between antennas for the four-antenna configuration.

We estimated the mutual coupling by using one of the antennas as a transmit antenna and the other two as receive antennas. The isolation measurements between antennas are presented in Figure 20. It can be seen an isolation better than  $-15$  dB between all antennas.

Theoretical analysis of the  $2 \times 2$ -antenna configuration regarding the isolation between antennas show that the worst isolation is between antennas 1 and 4 and, antennas 1 and 3 as shown in Figure 21.

## 5. Conclusions

We have shown that starting from a basic antenna design there are many possibilities for practical configurations targeting mobile applications. A single composite antenna can be configured to be a directional or omnidirectional GPS antenna. With a few minor changes in the shape of the print and adding an adaptive/tuning circuit with a PIN diode can significantly improve the antenna gain, polarization, and selectivity properties. A shift and enlargement of the frequencies bands has also been observed. From directivity diagrams follows that the antenna directive gain depends on the PIN diode polarization. It can also be observed an improvement in the antenna gain and a change in the antenna radiation pattern distribution. We proved that by changing the PIN diode DC voltage it is possible to select a specific directivity and also to select a number of simultaneous frequency bands or to enlarge one or more frequency bands.

We also showed that starting from the single antenna configuration a system of MIMO antennas can be also obtained with very good performance.

## References

- [1] J. Glossner, M. Schulte, M. Moudgill, et al., "Sandblaster low-power multithreaded SDR baseband processor," in *Proceedings of the 3rd Workshop on Applications Specific Processors (WASP '04)*, pp. 53–58, Stockholm, Sweden, September 2004.
- [2] C. Sturm, M. Porebska, J. Timmermann, and W. Wiesbeck, "Investigations on the applicability of diversity techniques in ultra wideband radio," in *Proceedings of the International Conference on Electromagnetics in Advanced Applications (ICEAA '07)*, pp. 899–902, Torino, Italy, September 2007.
- [3] A. Lambrecht, S. Schulteis, and W. Wiesbeck, "Diversity antenna system for radio reception in automotive applications," in *Proceedings of the International Conference on Electromagnetics in Advanced Applications (ICEAA '07)*, pp. 21–24, Torino, Italy, September 2007.
- [4] G. Jue and Agilent Technologies, "Addressing the design and verification challenges of long term evolution," *Wireless Design & Development*, no. 9, pp. 24–30, 2008.
- [5] E. Surducan, D. Iancu, and J. Glossner, "Modified printed dipole antenna for wireless multi-band communication devices," in *Proceedings of the URSI International Symposium on Electromagnetic Theory (EMTS '04)*, pp. 1161–1163, Pisa, Italy, May 2004.
- [6] E. Surducan, D. Iancu, and J. Glossner, "Modified printed dipole antennas for wireless multi-band communication systems," US patent no. 7034769, April 2006.
- [7] E. Surducan, D. Iancu, and J. Glossner, "Modified printed dipole antennas for wireless multi-band communication systems," US patent no. 7095382, April 2006.
- [8] R. Garg, P. Bhartia, I. Bahl, and A. Ittipibonon, *Microstrip Antenna Design Handbook*, Artech House, London, UK, 2001.
- [9] E. L. Krenz and D. J. Tammen, "Single compact dual mode antenna," US patent no. 5532708, July 1996.
- [10] E. Surducan, D. Iancu, V. Surducan, and J. Glossner, "Microstrip composite antenna for multiple communication protocols," *International Journal of Microwave and Optical Technology*, vol. 1, no. 2, pp. 772–775, 2006.
- [11] E. Surducan, D. Iancu, and J. Glossner, "Microstrip multi-band composite antenna," patent pending WO/2006/086194, August 2006.
- [12] R. Gonzalo, G. Nagore, and P. de Maagt, "Simulated and measured performance of a patch antenna on a 2-dimensional photonic crystals substrate," *Progress in Electromagnetics Research*, vol. 41, pp. 257–269, 2003.
- [13] H. Mosallaei and K. Sarabandi, "Engineered meta-substrates for antenna miniaturization," in *Proceedings of the URSI International Symposium on Electromagnetic Theory (EMTS '04)*, vol. 1, pp. 191–193, Pisa, Italy, May 2004.
- [14] N. Kaneda, W. R. Deal, Y. Qian, R. Waterhouse, and T. Itoh, "A broad-band planar quasi-Yagi antenna," *IEEE Transactions on Antennas and Propagation*, vol. 50, no. 8, pp. 1158–1160, 2002.
- [15] E. Surducan, D. Iancu, V. Surducan, and S. Stanley, "Multi-band antennas for SDR wireless handset application," in *Proceedings of the International Conference on Electromagnetics in Advanced Applications (ICEAA '07)*, pp. 523–526, Torino, Italy, September 2007.
- [16] C. B. Dietrich Jr., R. M. Barts, W. L. Stutzman, and W. A. Davis, "Trends in antennas for wireless communications," *Microwave Journal*, vol. 46, no. 1, pp. 22–44, 2003.
- [17] E. Surducan, D. Iancu, V. Surducan, and J. Glossner, "Miniature multiband antennas for hand held WiMAX and WiFi application," in *Proceedings of the International Conference on Electromagnetics in Advanced Applications (ICEAA '07)*, pp. 13–16, Torino, Italy, September 2007.
- [18] F. Yang and Y. Rahmat-Samii, "A reconfigurable patch antenna using switchable slots for circular polarization diversity," *IEEE Microwave and Wireless Components Letters*, vol. 12, no. 3, pp. 96–98, 2002.
- [19] H.-R. Chuang, L.-C. Kuo, C.-C. Lin, and W.-T. Chen, "A 2.4 GHz polarization-diversity planar printed dipole antenna for WLAN and wireless communication applications," *Microwave Journal*, vol. 45, no. 6, pp. 50–62, 2002.
- [20] E. Surducan, V. Surducan, D. Iancu, and J. Glosner, "Multi-band antenna with adaptive circuit," in *Proceedings of the 11th International Symposium on Microwave and Optical Technology (ISMOT '07)*, pp. 633–637, Villa Mondragone, Italy, December 2007.
- [21] M. D. Foegelle, "Antenna pattern measurement: concepts and techniques," *Compliance Engineering*, vol. 19, no. 3, pp. 22–33, 2002.
- [22] M. D. Foegelle, "Antenna pattern measurement: theory and equations," *Compliance Engineering*, vol. 19, no. 3, pp. 34–43, 2002.
- [23] M/A-COM, "Selection of antenna and cable for optimum automotive GPS performance," Application note M541, V4.0.

## Research Article

# A Geometry-Inclusive Analysis for Single-Relay Systems

Meng Yu,<sup>1</sup> Jing (Tiffany) Li,<sup>1</sup> and Hamid Sadjadpour<sup>2</sup>

<sup>1</sup> Department of Electrical and Computer Engineering, Lehigh University, Bethlehem, PA 18015, USA

<sup>2</sup> Baskin School of Engineering, University of California, Santa Cruz, CA 95064, USA

Correspondence should be addressed to Meng Yu, mey3@lehigh.edu

Received 14 January 2009; Accepted 31 March 2009

Recommended by Daniel Iancu

Successful message relay, or the quality of the interuser channel, is critical to fully realize the cooperative benefits promised by the theory. This in turn points out the importance of the geometry of cooperative system. This paper investigates the impact of the relay's location on the system capacity and outage probability for both amplify-forward (AF) and decode-forward (DF) schemes. Signal attenuation is modeled using power laws, and capacity is evaluated using the max-flow min-cut theory. A capacity contour for DF, the more popular mode of the two, is provided to facilitate the derivation of engineering rules. Finally, a selective single-relay system, which selects the best relay node among a host of candidates according to their locations, is analyzed. The average system capacity and outage, averaged over all possible candidates' locations, are evaluated. The result shows that the availability of a small candidate pool of 3 to 5 nodes suffices to reap most of the cooperative gains promised by a selective single-relay system.

Copyright © 2009 Meng Yu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

## 1. Introduction

Aside from temporal and frequency diversities, spatial diversity is another technique to mitigate the deterioration caused by fading. Due to the limitation on the size of mobile terminals, multiple antennas are not always practical. As a remedy to this, user cooperation has been proposed [1, 2], where multiple users share antennas to form a virtual antenna array and obtain spatial diversity.

Aiming at increasing the channel capacity or decreasing the outage probability or both, several interesting cooperative protocols have been proposed (e.g., [2, 3]). Among them, amplify-forward (AF) and decode-forward (DF) are the two fundamental forwarding modes. Their qualities have been studied by many researchers both from the information theoretic aspects and the practical aspects (e.g., [2, 4]). Reference [5] evaluated their performances in practical wireless scenarios in general, and the interuser outage case in particular. (By an interuser outage, we mean that the relay is unable to extract a clean copy of the source data.) It was shown in [5] that (1) the interuser outage happens at a nonnegligible probability even with decent channel code protection; for example, on block Rayleigh fading channels with an interuser signal-to-noise ratio (SNR) of 10–22 dB, the interuser outage happens at a probability of

10.4%–1.06% even with the protection of a (3000,2000) random low-density parity-check (LDPC) code; (2) when an interuser outage happens, both AF and DF perform badly with an effective diversity order of only 1; (3) the overall system performance is to a large extent limited by this worst-case scenario. These results revealed that a high-quality interuser channel is one key to realize the great benefits that user cooperation may offer.

Prior results in turn point out the importance of the location of the relay. On one hand, if the system cannot choose its relay's location, what is the average performance of a cooperative system when the relay can move to any place (bad location or good location) in a region around the source and the destination? Or how much does a relay node picked up randomly help the system? On the other hand, if the system can judiciously choose its relay partner among a host of candidate nodes, what would be the desired location for the relay? How much benefits if many candidate nodes are available?

Intuitively, when the relay gets close to the source, the interuser outage tends to diminish and the cooperative system tends to resemble a 2-by-1 multiple-input single-output (MISO) system, but how close is close? Further, because 2-by-1 and 1-by-2 systems are capacity comparable, does this suggest that a relay close to the destination would



also work in favor? More generally, is there a symmetry or duality property in the relay system?

The purpose of this paper is to answer the above questions and to understand the effect caused by the location of the relay. Reference [6] analyzed the performance of relay networks based on achievable rate region. This paper focuses on capacity evaluation, including the ergodic channel capacity and the outage probability. Capacity by definition establishes limits on the performance of practical communication systems. These limits provide system benchmarks and reveal how much improvement is theoretically possible. Several researchers have studied the information-theoretic aspects of the two-transmitter one-destination wireless cooperative system but only for a few samples of fixed channel qualities. In this work, we also study the capacity as a function of geometry. A similar study was conducted for the Gaussian channels in [7]. We consider both DF and AF modes for the single-relay cooperative system on Rayleigh fading channels using power law air propagation models. The system limits are first analyzed using the max-flow min-cut theory for different relay locations in several topologies. The results show that, in AF, system achieves high performance when the relay is near the median line between the source and the destination in most cases; in DF, system achieves high capacity when the relay is at the source side. It is worth mentioning that our analysis reveals a symmetry property in the capacity for AF, but not in DF. The reason behind it, particularly in terms of why DF does not mimic multiple-input multiple-output (MIMO) systems, is provided. A capacity contour for DF, the more practical and useful mode of the two, is subsequently computed. The capacity contour clearly provides motivation and guidelines for choosing good partners in practical situations. Following these results, a selective single-relay system, which selects the best relay node among a host of candidate nodes, is analyzed and evaluated. Different from [8], we consider effects of the number of potential relays for a single-relay system and use ergodic capacity or outage probability as selection criteria. We consider uniform random distributions of the candidates and compute the average system capacity and outage, where the average is taken among all possible candidates' locations. The results show that when the signal attenuation due to path loss is severe, or when there are at least two candidates available (so that the better one is selected as the relay), the gain achieved through user cooperation always outweighs the rate loss due to relaying. Further, it is sufficient to keep track of only a small pool of 3–5 candidates to reap most of the gain available to a single-relay system.

In the body of this paper, Section 2 briefs on the system model and cooperative modes. Section 3 analyzes the ergodic capacity and outage probability affected by the relay at different locations, and gives a capacity contour. Section 4 briefs the selective single-relay system and analyzes its performance. Section 5 presents our conclusion.

## 2. User Cooperation

**2.1. System Model.** In this paper we consider single-relay wireless system. Let “home channel”, “interuser channel”, and

“relay channel” denote the channels between the source and the destination, the source and the relay, and the relay and the destination, respectively. Let  $h_{SD}$ ,  $h_{SR}$ , and  $h_{RD}$  denote the respective channel gains. Since user cooperation is most useful when channels are varying very slowly (i.e., hard to obtain time diversity in a single user channel), the channels are modeled as Rayleigh block fading channels, where channels remain constant for the duration of one round of user cooperation (2 consecutive time slots). Between channels and cooperation rounds, the channel gains are independent. The general form of a signal received over a specific channel at time  $t$  is given by

$$y(t) = \sqrt{E_s}h(t)x(t) + n(t), \quad (1)$$

where  $E_s$  is the signal energy,  $h(t)$  is the channel gain, and  $n(t)$  is the additive white Gaussian noise (AWGN) with a power spectrum density  $N_0$ . The channel gain follows

$$h(t) = \alpha\phi(t), \quad (2)$$

where channel fading coefficient  $\phi(t)$  is a random variable that follows the Rayleigh distribution, and  $\alpha$  is the pathloss. We assume that the square of the pathloss is inversely proportional to some power of the distance, that is,  $\alpha^2 = l^{-\delta}$ , where  $l$  is the distance between the transmitter and the receiver, and  $\delta$ , an integer between [2, 4], is the pathloss exponent. (To ease the evaluation, we consider isotropic signal propagation model within each topological setup. In reality, however, the propagation model also depends on the environment and the transmission distance.)

Among the various possible strategies of user cooperation (e.g., [2, 3]), we consider half-duplex systems, the simplest type, where after the source transmits a package in the first time slot, the relay forwards the message in the second time slot. We assume that the channel side information (i.e., channel gain) is known to the respective receivers, and the power is equally allocated in the two time slots.

**2.2. Fundamental Cooperative Modes.** At the first time slot, the signal received at the destination is

$$y_{D,1} = \sqrt{E_s}h_{SD}x_1 + n_0, \quad (3)$$

where  $n_0$  denotes the zero-mean complex AWGN.

**2.2.1. AF Mode.** We assume that the power of the signal retransmitted at the relay node is scaled uniformly with respect to all the bits in the package, such that the average (re)transmission energy per signal equals  $E_s$ .

In the second time slot, the signal received at the destination is

$$y_{D,2}^{\text{AF}} = h_{RD}|h_{SR}|\sqrt{\frac{E_s^2}{E_s|h_{SR}|^2 + N_0}}x_1 + \tilde{n}, \quad (4)$$

where  $\tilde{n}$  is a zero mean complex Gaussian noise with a variance of  $(N_0/2 + N_0|h_{RD}|^2E_s/2(E_s|h_{SR}|^2 + N_0))$  per

dimension [4]. The destination combines  $y_{D,1}$  and  $y_{D,2}^{\text{AF}}$  using the maximal ratio combination (MRC) rule before decoding.

Let

$$C(\gamma) \triangleq \frac{1}{2} \log_2(1 + \gamma) \text{ bit/s/Hz} \quad (5)$$

denote the capacity of a single Gaussian channel in a cooperative system with a signal-to-noise ratio  $\gamma$ . The factor  $1/2$  is introduced to account for the fact that two consecutive time slots are used for each package.

For AF, it is easy to see that the achievable (instantaneous) information rate is upper bounded by the (instantaneous) mutual information of the compound channel [2]:

$$R^{\text{AF}} \leq I^{\text{AF}} = C(\|\gamma^{\text{AF}}\|_1), \quad (6)$$

where

$$\gamma^{\text{AF}} = \left[ \underbrace{\frac{E_s |h_{SD}|^2}{N_0}}_{\text{1st time slot}}, \underbrace{\frac{E_s^2 |h_{RD} h_{SR}|^2}{(E_s |h_{RD}|^2 + E_s |h_{SR}|^2 + N_0) N_0}}_{\text{2nd time slot}} \right], \quad (7)$$

$$\|\mathbf{a}\|_1 = \sum_i |a_i|. \quad (8)$$

That  $h_{SR}$  and  $h_{RD}$  are interchangeable in the above SNR formulation suggests a capacity symmetry with respect to the position of the source and the destination in AF.

**2.2.2. DF Mode.** In DF, the relay demodulates and decodes the packet and forwards *part or all* of the information possibly using a different (compression or error control) code. We note that the decode-forward strategy we consider here has certain flavor of the *compression-forward* (CF) mode discussed in [9]. The difference, however, is that the relay in CF needs not to decode the message and, rather, forwards compressed versions of its observations. With different channel states, different coding strategies may outperform each other [10, 11]. We assume that the system can switch between coding strategies to exploit the channel capacity. From network information theory, one realizes that the achievable information rate of DF is determined by the max-flow min-cut of the system. The cut set around the source forms a broadcast channel, and the cut set around the destination forms a parallel channel (because of the orthogonality in time). The system capacity is the minimum value of the two cut sets' capacity. Within each time slot, the channels can be treated as Gaussian channels. Because a Gaussian broadcast channel is a degraded broadcast channel, the better channel can always carry the information intended for the worse channel also. The capacity of the broadcast cut set is therefore the one with better signal-to-noise ratio:

$$C_{\text{cut1}} = \max\{C_{SD}, C_{SR}\}, \quad (9)$$

where

$$\begin{aligned} C_{SD} &\triangleq C\left(\frac{E_s}{N_0} |h_{SD}|^2\right), \\ C_{SR} &\triangleq C\left(\frac{E_s}{N_0} |h_{SR}|^2\right). \end{aligned} \quad (10)$$

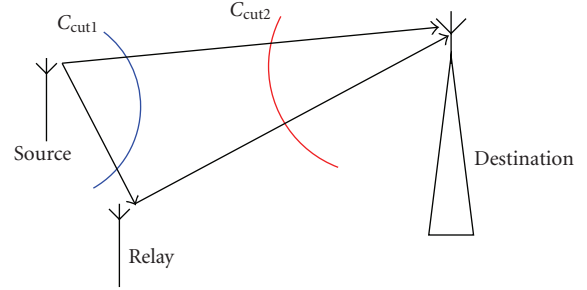


FIGURE 1: Cut sets of relay system.

On the other hand, the capacity of the cut set around the destination is the sum rate of the two paralleled channels:

$$\begin{aligned} C_{\text{cut2}} &= C_{SD} + C_{RD}, \\ C_{RD} &\triangleq C\left(\frac{E_s}{N_0} |h_{RD}|^2\right). \end{aligned} \quad (11)$$

Hence, the system's instantaneous achievable rate is

$$R^{\text{DF}} \leq \min\{C_{\text{cut1}}, C_{\text{cut2}}\}. \quad (12)$$

Based on the relative value of  $C_{SD}$ ,  $C_{SR}$ , and  $C_{RD}$ , the detailed (instantaneous) information rate for DF is upper bounded by

$$R^{\text{DF}} \leq \begin{cases} C_{SD}, & \text{if } C_{SR} \leq C_{SD}, \\ C_{SR}, & \text{if } C_{SD} < C_{SR} \leq C_{SD} + C_{RD}, \\ C_{SD} + C_{RD}, & \text{if } C_{SD} + C_{RD} < C_{SR}. \end{cases} \quad (13)$$

Now that we have the instantaneous rates for the AF and DF system in (6) and (13), respectively, we can average them over the distribution of the channel gain (a Rayleigh distribution whose mean is some power of the distance) to account for the signal attenuation caused by the channel fading and the geometry of the terminals. These results are plotted in Figures 3 and 6 and are discussed in the succeeding sections.

### 3. Capacity and Outage at Different Locations

**3.1. Ergodic Capacity.** Ergodic capacity, more commonly known as the Shannon limit, determines the maximum achievable information rate averaged over all fading states. Under the assumption that the system can adopt appropriate strategy to exploit the capacity, we have the ergodic capacity as

$$\begin{aligned} C_{\text{erg}}^{\text{AF}} &= \iiint R^{\text{AF}} dh_{SR} dh_{SD} dh_{RD}, \\ C_{\text{erg}}^{\text{DF}} &= \iiint R^{\text{DF}} dh_{SR} dh_{SD} dh_{RD}. \end{aligned} \quad (14)$$

As shown in Figure 2, the source and the destination are distributed horizontally with a distance  $l_{SD}$ , which is

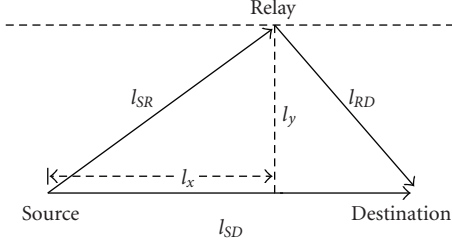


FIGURE 2: Geometry model.

normalized to unity. The horizontal distance and vertical distance from the relay to the source are  $l_x$  and  $l_y$ . Let us begin evaluation with a fixed  $l_y$  and a varying  $l_x$  first.

Figure 3 shows the ergodic capacity where  $l_{SD} = 1$ ,  $E_s = 1$ ,  $N_0 = 1$ , and  $l_y = 0.5$ . Solid curves represent AF, dashed curves represent DF, and power law propagation models of  $\delta = 2, 3, 4$  are evaluated. From the curves, we can see that regardless of the value of  $\delta$ , the capacity of the AF system exhibits a symmetry property, and for the cases we tested, the maximum value is achieved at the median point. The former observation confirms that the positions of the source and the destination are interchangeable, as is implied in (7).

For the cases we tested, to analyze the effect of the relay's location, we take the model with  $\delta = 2$  as an example. First, note that the effective SNR of the AF system is the sum SNRs of two spatially independent channels: the direct channel between the source and the destination, and the cascade channel consisting of the source-relay channel and the relay-destination channel. The SNR of the former is irrespective to the relay, and the SNR of the latter varies with the location of the relay. The average SNR of the cascade channel is an integral over all states of  $h_{SR}$  and  $h_{RD}$ . We analyze the value when  $h_{SR}$  and  $h_{RD}$  are around their mean values, which plays a major part in integral because of its high probability. At this time, the SNR of the cascade channel can be transformed into the following expression:

$$\begin{aligned} \gamma_{\text{cas}} &\propto \frac{1}{l_{SR}^2 + l_{RD}^2 + l_{SR}^2 l_{RD}^2 (N_0/E_s)} \\ &= \frac{1}{(N_0/E_s)(l_{SR}^2 + E_s/N_0)(l_{RD}^2 + E_s/N_0) - E_s/N_0} \\ &= \frac{1}{(N_0/E_s) \left( l_x^2 + l_y^2 + E_s/N_0 \right) \left[ (l_{SD} - l_x)^2 + l_y^2 + E_s/N_0 \right] - E_s/N_0}, \end{aligned} \quad (15)$$

where  $l_{SD}$ ,  $l_{RD}$ , and  $l_{SR}$  are the distances between the respective nodes. The main part of the denominator can be written as

$$\begin{aligned} &\left( l_x^2 + l_y^2 + \frac{E_s}{N_0} \right) \left[ (l_{SD} - l_x)^2 + l_y^2 + \frac{E_s}{N_0} \right] \\ &= (l_x^2 + A^2) \left[ (l_{SD} - l_x)^2 + A^2 \right] \\ &= l_{SD}^2 A^2 + \left[ \frac{1}{4} (l_{SD}^2 - 4A^2) - \left( \frac{1}{2} l_{SD} - l_x \right)^2 \right]^2, \end{aligned} \quad (16)$$

where  $A^2 = l_y^2 + E_s/N_0$  is a constant value. If  $l_{SD}^2 \leq 4A^2$ , the denominator achieves its minimum when  $l_x = 0.5l_{SD}$ . Otherwise, the denominator has two minimum values achieved when

$$l_x = 0.5l_{SD} \pm \sqrt{0.25l_{SD}^2 - A^2}. \quad (17)$$

Accordingly, the SNR of the cascade channel, and subsequently the effective SNR of the the AF system, will reach either one maximum value on the median line between the source and the destination or two maximum values located symmetrically on either side of the median line.

Figure 4 demonstrates an example of the latter case. A relatively low transmit power of  $E_s = 0.24$  is used, and both the ergodic capacity and the outage probability are plotted as functions of the relay location. It should be noted that the two optimal locations specified in (17) are for Gaussian channels. To evaluate fading channels, one needs to average over all the instantaneous Gaussian channel realizations (pertaining to the fading distribution). With a small transmit power such as  $E_s = 0.24$ , the case of  $l_{SD}^2 > 4A^2$  dominates; so there exit two optimal locations (which are functions of the fading coefficient) most of the time and one optimal location (at the median point). The best locations are selected by evaluating the average information rate over all the possible channel realizations. In terms of evaluating the outage probability, the curve may look slightly different with respect to different threshold values. In the Figure, the optimal locations obtained by the capacity and the outage results do not coincide exactly, but they are quite close. Unlike one may expect, these optimal locations are actually closer to the source and the destination than to their median position. We also observe that an arbitrary position between the two optimal locations suffers but only a small performance degradation. This suggests that the median position is nonetheless a convenient and safe choice for the relay.

The symmetry property, however, is not observable in DF. The capacity of DF peaks out when the relay sits at some position near the source, but unlike one would expect from a 2-by-1 MISO system, the optimal location is not very close to the source but appears to be between the 1 : 9 and 3 : 7 sections from the source to the destination for different propagation models. Recall that the capacity of DF is  $\min\{C_{\text{cut1}}, C_{\text{cut2}}\}$ . As the relay moves from the source to the destination, the increase of  $C_{\text{cut1}}$  will cause the decrease of  $C_{\text{cut2}}$  and vice versa. Hence, the maximum value is achieved when  $C_{\text{cut1}} = C_{\text{cut2}}$ . When  $\delta$  is large, that is, high-order signal attenuation,  $C_{SR}$  and  $C_{RD}$  tend to dominate  $C_{\text{cut1}}$  and  $C_{\text{cut2}}$ , making the capacity curve closer to symmetric and the optimal relay location closer to the median line. Additionally, we see that each capacity curve consists of three segments. The first segment, spanning from the source to the optimal relay location, represents the case when the cut set around the destination ( $C_{\text{cut2}}$ ) is the bottleneck for information flow. In that case, the capacity increases as the relay moves toward the destination. The second segment represents the case when the cut set around the source ( $C_{\text{cut1}}$ ) dominates,

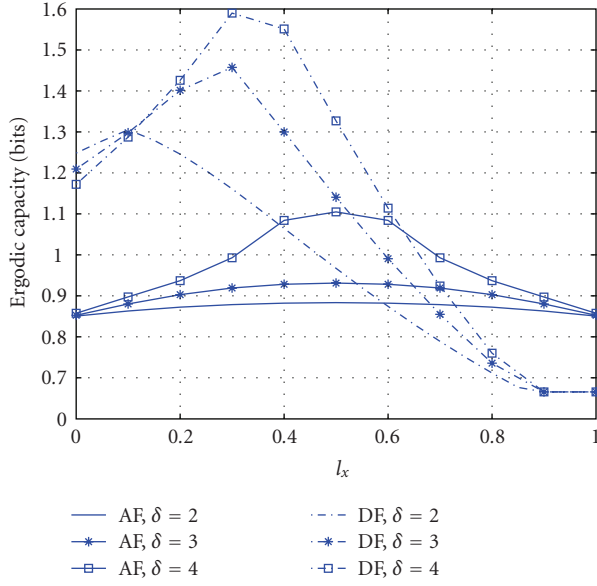


FIGURE 3: Ergodic capacity versus the location of the relay,  $E_s = 1$ ,  $N_0 = 1$ ,  $l_y = 0.5$ , and  $l_{SD} = 1$ .

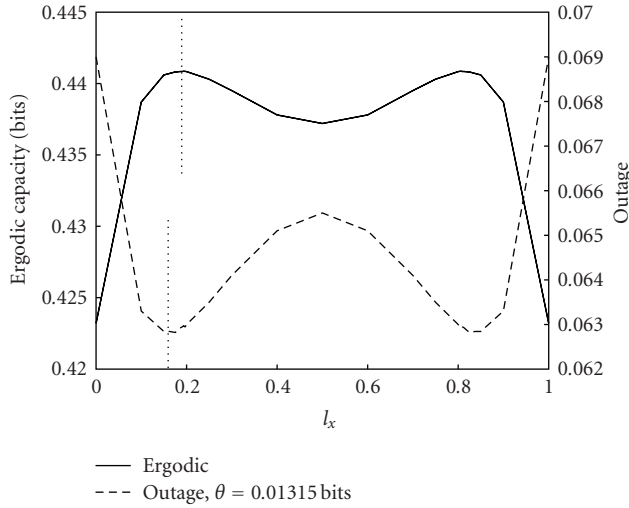


FIGURE 4: Ergodic capacity and outage ( $\theta = 0.01315$ ) of AF versus the location of the relay,  $\delta = 2$ ,  $E_s = 0.24$ ,  $N_0 = 1$ ,  $l_y = 0.05$ , and  $l_{SD} = 1$ .

and consequently the capacity decreases as the relay moves away from the source. Finally the capacity reaches a floor that is irrelevant to the relay location. This happens when the quality of the interuser channel is worse than the home channel, that is,  $l_{SR} > l_{SD}$ , and the relay system reverts to the noncooperative mode.

**3.2. Outage Probability.** Outage probability, aka outage capacity or simply, outage, is another important statistical measure for the quality of a fading channel especially in slow fading cases. Outage specifies the probability that the instantaneous channel quality fails to meet a satisfactory

threshold  $\theta$ . Using information rate as the measure for channel quality, the outage probability for a single channel can be computed using

$$P_{\text{out}}(\theta) \triangleq \Pr(R < \theta) = \int_0^\theta f_C(c)dc, \quad (18)$$

where  $f_C(c)$  is the probability density function of the instantaneous information rate of that channel.

As shown in (7), the AF system can be viewed as a single channel with an effective SNR  $\|\gamma^{\text{AF}}\|_1$ ; hence, the outage probability can be evaluated numerically using (18).

For DF, the outage needs to be evaluated with respect to the cases when the cut set around the source or around the destination dominates. The former is a degraded broadcast channel where outage happens when  $C_{\text{cut1}} = \max(C_{SR}, C_{SD}) < \theta$ . The latter is a parallel channel where outage happens when  $C_{\text{cut2}} = C_{SD} + C_{RD} < \theta$ . Overall the outage for DF system can be computed as

$$\begin{aligned} P_{\text{out}} &= 1 - \Pr(C_{\text{cut1}} > \theta)\Pr(C_{\text{cut2}} > \theta) \\ &= \Pr(\max(C_{SR}, C_{SD}) < \theta) \\ &\quad \cdot \Pr(\max(C_{SR}, C_{SD}) < C_{SD} + C_{RD}) \\ &\quad + \Pr(C_{SD} + C_{RD} < \theta) \\ &\quad \cdot \Pr(\max(C_{SR}, C_{SD}) > C_{SD} + C_{RD}) \\ &= \Pr(C_{SR} < \theta)\Pr(C_{SD} < \theta) \\ &\quad + (1 - \Pr(C_{SR} < \theta))\Pr(C_{SD} + C_{RD} < \theta). \end{aligned} \quad (19)$$

Figure 5 shows the outage probabilities of the model with a threshold  $\theta = 0.35$  bits per channel use,  $N_0 = 0$ , and  $E_s = 1$ . The outage results appear quite consistent with the capacity results. For AF, the outage curve is also symmetric, and the lowest outage is achieved when the relay resides in equal distances between the source and the destination when the transmission power is high. For DF, the optimal relay position in terms of the least outage is somewhere between the 3 : 7 sections from the source to the destination.

Figure 4 is a case when the transmission power is low ( $E_s = 0.24$ ). The dashed curve represents the outage with a threshold  $\theta = 0.01315$  bits. There are two symmetric optimal locations; however they are not the same ones of the ergodic capacity. Though not shown here, different threshold also has different optimal locations. This is because different instantaneous channel fades have the similar effects as different transmission powers, and different transmission powers may correspond to different optimal locations. The optimal locations for the outage threshold  $\theta$  are averaged from 0 to  $\theta$ . This means that different criteria (ergodic capacity or outage, outages with different thresholds) may have different optimal locations.

When the transmission power is low, the optimal location is not on the median line (the numerical results are not shown here).



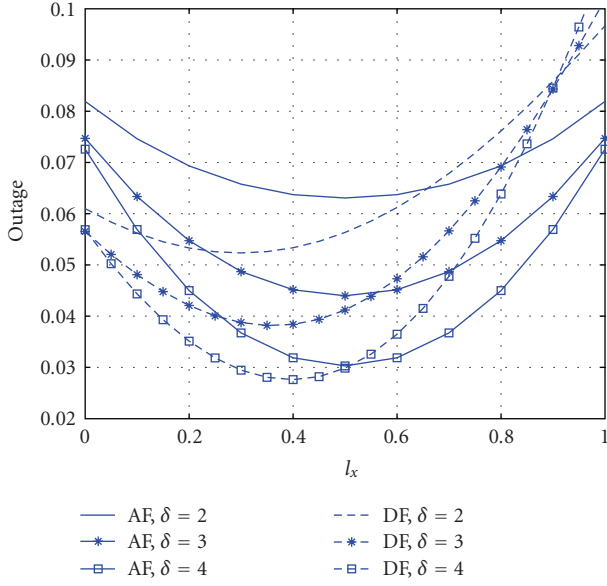


FIGURE 5: Outage probability versus the location of the relay,  $E_s = 1$ ,  $N_0 = 1$ ,  $\gamma = 0.5$ , and  $l_{SD} = 1$ .

**3.3. Capacity Contour.** To cast a complete view of how the system capacity relates to the geometry of the terminals and to provide engineering guidelines for choosing the optimal relay location, we plot in Figure 6 the capacity contour for DF when the relay is at different positions. To ease analysis, the source and the destination are placed at positions  $(0, 0)$  and  $(0, 1)$  with a normalized distance of 1. We take the case when the signal follows the cubic law attenuation ( $\delta = 3$ ) and  $E_s = 1$ . It is interesting to observe that the contour curves are completed by two sets of arcs, cocentered at the source and the destination terminals, respectively. These arcs correspond to the capacities of the two cut sets around the source and the destination. We see that the capacity is maximized by choosing a relay that sits at the 4 : 6 section between the source and the destination. The contour is denser at the destination side than at the source side. When the relay moves farther beyond the destination, the capacity of the home channel will have exceeded that of the interuser channel, that is,  $C_{SR} < C_{SD}$ . Hence the relay node will stop message forwarding, and the cooperative system degenerates to a single-channel system with a capacity of  $C_{SD}$ .

The relation between the cooperative systems, that is, virtual antenna arrays, with the true multi-antenna MIMO systems, has been the interest for a while. Work of Kramer et al. [7] and Xie and Kumar [12] reveals that decode-forward (CF) is akin to multi-antenna transmission, and compress-forward (CF) is akin to multi-antenna reception. It has further been suggested that DF will achieve the maximal capacity when the relay moves toward the source, and CF will achieve the maximal capacity when the relay moves toward the destination. Our results about DF are quite consistent with the multi-antenna interpretation. However, using practical signal attenuation models, we have found that the optimal

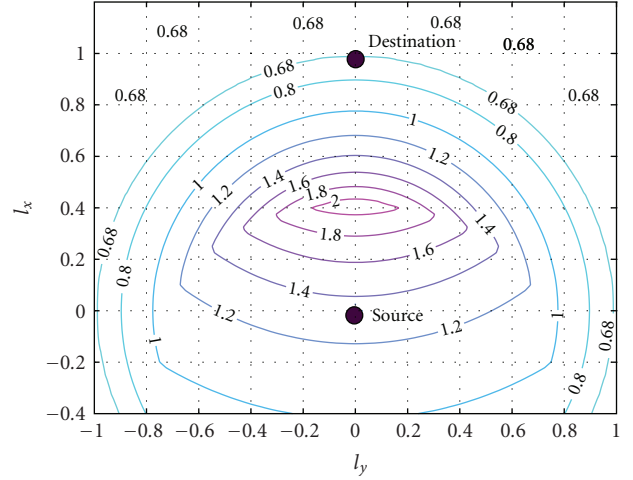


FIGURE 6: Ergodic capacity versus the location of the relay.

location for the relay needs not to be extremely close to the source.

To see why the DF relay system does not perform nearly as well as a 1-by-2 single-input multioutput (SIMO) system even when the relay gets very close to the destination (i.e., making the relay-destination channel near-perfect), consider the difference in the decoding strategies. In the SIMO system, the signals received by the multiple antennas are optimally combined and *jointly* decoded; whereas in the DF relay system, the signals received by the virtual antenna array are *separately* decoded (i.e., the relay demodulates and decodes its received signals and passes hard-decisions to the destination). In this sense, compression-forward appears to be the dual of decode-forward. If the compression of the received (analog) signals at the relay is near-lossless, then the destination will attain undistorted copies of all the signals received at the virtual antenna array and can therefore perform optimal combining and joint decoding.

#### 4. Relay Selection

In previous sections, we have calculated the ergodic capacity and outage probability when the relay is at different locations. What is the expected capacity of a single-relay system whose relay is moving in a given region around the source and the destination? If there are many relay candidate nodes randomly distributed in the region and a selective single-relay system that chooses the candidate node with the best location as its relay, what is the performance of this system? We assume that the system knows the position of each node by using GPS technology or a localization algorithm.

Assume that the distribution of the relay's location takes a certain probability distribution function  $p_{X,Y}(x, y)$ , where  $x, y$  are coordinates of the relay's location. Let us rewrite the ergodic capacity and outage probability of DF at different  $x, y$  as functions of  $(x, y)$

$$\begin{aligned} C_{\text{erg}}^{\text{DF}} &= C_{\text{erg}}(x, y), \\ P_{\text{out}}^{\text{DF}}(\theta) &= P_{\text{out}}(x, y, \theta). \end{aligned} \quad (20)$$



Let  $F_C(c)$  and  $F_P(p, \theta)$  denote the cumulative distribution function of the ergodic capacity and the outage probability at threshold  $\theta$  of the region based on location statistics, respectively,

$$\begin{aligned} F_C(c) &= \iint_{x,y} C_{\text{erg}}(x, y) dx dy, \\ F_P(p, \theta) &= \iint_{x,y} P_{\text{out}}(x, y, \theta) dx dy. \end{aligned} \quad (21)$$

Now assume that we have  $K$  candidate nodes with independent identical distribution in that region. We also assume the system can find locations of relays through some pilot signals. The system selects the candidate node with the best performance to be the relay node, while others keep silent. In other words, the system selects the node with the lowest outage probability or the highest ergodic capacity. The cumulative distribution function of this system's ergodic capacity is

$$\begin{aligned} F_{C,\text{sel}}(c) &= P\left(\max\{C_{\text{erg},i}^{\text{DF}} \mid i = 1, 2, \dots, K\} < c\right) \\ &= F_C(c)^K, \end{aligned} \quad (22)$$

where  $C_{\text{erg},i}^{\text{DF}}$  is the ergodic capacity when the  $i$ th node behaves as a relay.

The cumulative distribution function of the system outage probability is

$$\begin{aligned} F_{P,\text{sel}}(p, \theta) &= P\left(\min\{P_{\text{out},i}^{\text{DF}}(\theta) \mid i = 1, 2, \dots, K\} < p\right) \\ &= 1 - P\left(\min\{P_{\text{out},i}^{\text{DF}}(\theta) \mid i = 1, 2, \dots, K\} > p\right) \\ &= 1 - (1 - F_P(p, \theta))^K, \end{aligned} \quad (23)$$

where  $P_{\text{out},i}^{\text{DF}}(\theta)$  is the outage probability when the  $i$ th node works as a relay.

Because it is difficult to get a closed form expression of  $C_{\text{region}}^{\text{DF}}$  and  $P_{\text{region}}^{\text{DF}}(\theta)$ , we use a discretized numerical method to calculate it.

(1) Divide the total region into tiny grids ( $\Delta$ ) of equal area. Assume that the system performance (either capacity or outage) remains invariant when the relay moves within a grid.

(2) Calculate the ergodic capacity  $C_{\Delta,i}^{\text{DF}}$  and the outage probability  $P_{\Delta,i}^{\text{DF}}(\theta)$  for each outage threshold  $\theta$  in each grid (where  $i$  is the index of the grid).

(3) Divide the range of the capacity or outage values into many equally spaced bins (i.e., uniform quantization of the capacity/outage values), and count the numbers of  $C_{\Delta,i}^{\text{DF}}$  and  $P_{\Delta,i}^{\text{DF}}(\theta)$  falling into each bin to form the respective histograms.

(4) Properly normalize these histograms. These normalized histograms, denoted as  $\hat{p}_C(c)$  and  $\hat{p}_P(p, \theta)$ , serve as

the discretized approximations of the capacity distribution and outage distribution (which are functions of the location distribution of the relay).

When both the area of the grid and the step size of the bin approach zero,  $\hat{p}_C(c)$  and  $\hat{p}_P(p, \theta)$  approach the probability distribution function of the ergodic capacity  $p_C(c)$  and the outage probability  $p_P(p, \theta)$  of the region, respectively. From  $\hat{p}_C(c)$  and  $\hat{p}_P(p)$ , we can compute the respective cumulative distribution functions (cdf),  $\hat{F}_C(c)$ , and  $\hat{F}_P(p, \theta)$ , which are used as approximation to the true cdf's  $F_C(c)$  and  $F_P(p)$ .

As an example, we consider the case when candidate relay nodes are confined to a square region around the source and the destination. Following the setup in Figure 2, we place the source and the destination at locations  $(l_x, l_y) = (0, 0)$  and  $(1, 0)$ , respectively, and choose both the length and the width of the relay region to be unity:  $l_x \in [0, 1]$  and  $l_y \in [-0.5, 0.5]$ . Without loss of generality, assume that the candidate relay nodes are all independent and moves around in the region uniformly and randomly.

When no candidate relay node is available, the system degrades into a noncooperative system, and the source keeps transmitting different messages in both time slots. When there are more than one candidate nodes, the system becomes a best selective single-relay system, where a message is transmitted by the source in the first time slot and relayed by the best candidate (i.e., in the best location) in the second time slot.

In Figure 7, solid curves represent the average ergodic capacity of the best selective single-relay system with the number of candidates from 0 to 40, where the average is taken among all the possible relay locations. In the figure, when the pass loss is relatively small, that is,  $\delta = 2, 3$ , the availability of only one or two candidate nodes may not provide any gain in terms of the average ergodic capacity. As the number of candidate nodes increases, the benefit of user cooperation begins to outweigh the loss in bandwidth efficiency caused by cooperation. For the case of  $\delta = 4$ , because of the severe path loss, the diversity gain provided by the relay system becomes crucial, and the relay system appears to unanimously outperform the noncooperative system even though there is only one candidate to select from. As the number of the candidates increases without bound, the average system capacity approaches a limit, which is achieved by positioning (at least) one relay candidate at the optimal location at all times.

Figure 8 demonstrates the average outage probability of the best selective single-selection system (solid curves). The performances are similar to the case of the capacity, except that, regardless of what value of  $\delta$  or how many candidate nodes are used, cooperation always elevates the outage performance. Further, the outage probability drops at a faster rate than the increase in the capacity (Figure 7) as the size of the candidate pool increases. As soon as the size of the candidate pool reaches 3 to 5, the outage performance get quite close to the limit (dashed lines). This suggests that, in the best selective single-relay system, it is sufficient for the system to keep track of only 3 to 5 candidates in order to reap most of the gain promised by the theory.

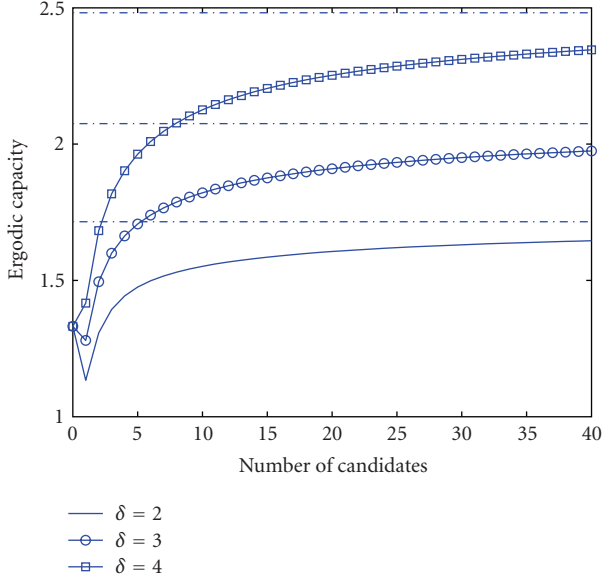


FIGURE 7: Ergodic capacity of single-relay selection system with up to 40 candidate nodes.

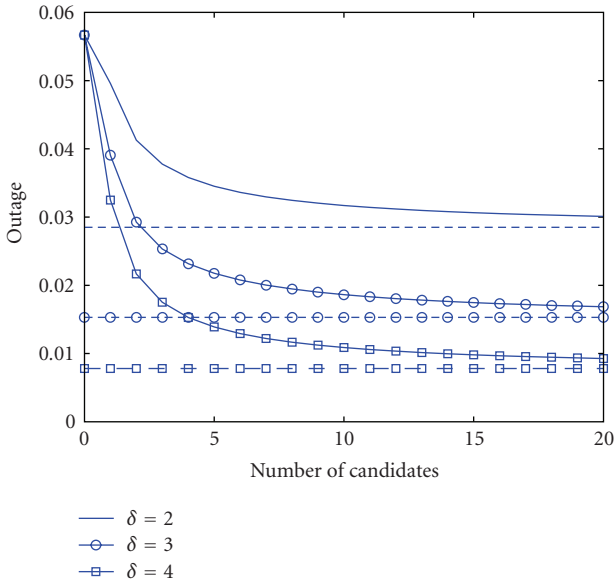


FIGURE 8: Outage of single-relay selection system with up to 20 candidate nodes.

## 5. Conclusion

We have analyzed the performances of amplify-forward and decode-forward, the two basic relaying modes, for a single-relay system in Rayleigh fading environment. The max-flow min-cut theory is used as the base approach, and the performance measure is quantified by the ergodic capacity and the outage probability. We have explicitly taken into account the geometry of the nodes, the distances between them, and the resulting attenuation of radio signals and weighted the information as a function of the transmission distances.

In the case of amplify-forward, we have demonstrated an interesting symmetry property in both the capacity and the outage results. We have shown that the maximum value is achieved when the relay sits on the median line between the source and the destination in most cases, with the exception when the power is very low (below a certain threshold). In the latter case, two symmetric optimal locations on either side of the median line are observed, but our numerical evaluation shows that for the relay to locate anywhere between these two optimal locations incurs only a small performance degradation. Hence, the median point remains a convenient and good choice.

In the case of decode-forward, our capacity and outage results confirm that the system operates much like a multi-antenna transmission system [12]. Using practical signal attenuation models and the max-flow min-cut theorem, we have found that the optimal relay location is somewhere around, but not extremely close to, the source. To provide a complete picture of the system performance as a function of the geometry, a capacity contour plot is computed for the DF system. We note that a similarly contour plot can also be computed for the case of AF, but the computational complexity is much higher. (Each point for AF requires a 3-dimensional integration for the case of AF, but only a 2-dimensional integration for the case of DF.) Further, the contour plot would be more useful for DF than for AF, since the symmetry condition of AF and the “median point rule” make it easier to choose the relay location.

Following this geometry-inclusive analysis, a best selective single-relay system is proposed and analyzed. We consider the case where multiple candidates may be available, but only the one at the best location will be chosen each time to relay the message. Using a discretized numerical method, we have demonstrated the average system capacity and outage as a function of the size of the candidate pool. We observe that when the pass loss is relatively small, the availability of only one candidate node may not render an average system capacity higher than a noncooperative system (due to the loss in bandwidth efficiency). When more candidate nodes are available or when the pass loss is severe, the system performance of the relay system quickly picks up and soon outperforms that of the noncooperative system. We have further demonstrated that the source needs only keep track of some 3 to 5 candidate relays in order to harness most of the “geometric” benefits available to a selective relay system.

## Acknowledgments

The work is supported in part by the National Science Foundation under Grant no. CCF-0430634 and by the Commonwealth of Pennsylvania, Department of Community and Economic Development, through the Pennsylvania Infrastructure Technology Alliance (PITA).

## References

- [1] A. Sendonaris, E. Erkip, and B. Aazhang, “User cooperation diversity—part I: system description,” *IEEE Transactions on Communications*, vol. 51, no. 11, pp. 1927–1938, 2003.

- [2] J. N. Laneman, D. N. C. Tse, and G. W. Wornell, "Cooperative diversity in wireless networks: efficient protocols and outage behavior," *IEEE Transactions on Information Theory*, vol. 50, no. 12, pp. 3062–3080, 2004.
- [3] M. Janani, A. Hedayat, T. E. Hunter, and A. Nosratinia, "Coded cooperation in wireless communications: space-time transmission and iterative decoding," *IEEE Transactions on Signal Processing*, vol. 52, no. 2, pp. 362–371, 2004.
- [4] R. U. Nabar, H. Bolcskei, and F. W. Kneubuhler, "Fading relay channels: performance limits and space-time signal design," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 6, pp. 1099–1109, 2004.
- [5] M. Yu and J. Li, "Is amplify-and-forward practically better than decode-and-forward or vice versa?" in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05)*, vol. 3, pp. 365–368, Philadelphia, Pa, USA, March 2005.
- [6] Z. Lin, E. Erkip, and A. Stefanov, "Cooperative regions for coded cooperative systems," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '04)*, vol. 1, pp. 21–25, Dallas, Tex, USA, November–December 2004.
- [7] G. Kramer, M. Gastpar, and P. Gupta, "Cooperative strategies and capacity theorems for relay networks," *IEEE Transactions on Information Theory*, vol. 51, no. 9, pp. 3037–3063, 2005.
- [8] B. Zhao and M. C. Valenti, "Practical relay networks: a generalization of hybrid-ARQ," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 1, pp. 7–18, 2005.
- [9] T. Cover and A. El Gamal, "Capacity theorems for the relay channel," *IEEE Transactions on Information Theory*, vol. 25, no. 5, pp. 572–584, 1979.
- [10] M. A. Khojastepour, A. Sabharwal, and B. Aazhang, "Lower bounds on the capacity of Gaussian relay channel," in *Proceedings of the 38th Annual Conference on Information Sciences and Systems (CISS '04)*, pp. 597–602, Princeton, NJ, USA, March 2004.
- [11] A. Host-Madsen and J. Zhang, "Capacity bounds and power allocation for wireless relay channels," *IEEE Transactions on Information Theory*, vol. 51, no. 6, pp. 2020–2040, 2005.
- [12] L.-L. Xie and P. R. Kumar, "An achievable rate for the multiple level relay channel," *IEEE Transactions on Information Theory*, vol. 51, no. 4, pp. 1348–1358, 2005.