A Real-Time FPGA-Based Implementation of a High-Performance MIMO-OFDM Mobile WiMAX Transmitter*

Oriol Font-Bach[1], Nikolaos Bartzoudis[1],
Antonio Pascual-Iserte[1,2], and David López Bueno[1]

[1] Centre Tecnològic de Telecomunicacions de Catalunya (CTTC),
Parc Mediterrani de la Tecnologia (PMT), Av. Carl Friedrich Gauss 7,
08860 Castelldefels, Barcelona, Spain
{ofont,nbartzoudis,dlopez}@cttc.cat
[2] Department of Signal Theory and Communications,
Universitat Politècnica de Catalunya (UPC), Campus Nord, Jordi Girona 1-3,
08034 Barcelona, Spain
antonio.pascual@upc.edu

Abstract. The Multiple Input Multiple Output (MIMO)-Orthogonal Frequency Division Multiplexing (OFDM) is considered a key technology in modern wireless-access communication systems. The IEEE 802.16e standard, also denoted as mobile WiMAX, utilizes the MIMO-OFDM technology and it was one of the first initiatives towards the roadmap of fourth generation systems. This paper presents the PHY-layer design, implementation and validation of a high-performance real-time 2x2 MIMO mobile WiMAX transmitter that accounts for low-level deployment issues and signal impairments. The focus is mainly laid on the impact of the selected high bandwidth, which scales the implementation complexity of the baseband signal processing algorithms. The latter also requires an advanced pipelined memory architecture to timely address the datapath operations that involve high memory utilization. We present in this paper a first evaluation of the extracted results that demonstrate the performance of the system using a 2x2 MIMO channel emulation.

Keywords: MIMO, testbeds, IEEE 802.16e, real-time systems, space-time coding, FPGAs, DSP.

1 Introduction

The most innovating PHYsical layer (PHY) algorithms for Multiple Input Multiple Output (MIMO) systems supporting Orthogonal Frequency-Division Multiplexing (OFDM) are usually deployed using testbeds that operate in off-line

* This work was partially supported by the European Commission under projects NEWCOM++ (216715) and BuNGee (248267); by the Catalan Government under grants 2009 SGR 891 and 2010 VALOR 198; and by the Spanish Government under project TEC2008-06327-C03 (MULTI-ADAPTIVE) and Torres Quevedo grants PTQ-08-01-06441, PTQ06-02-0540, PTQ06-2-0553.

J. Del Ser et al. (Eds.): MOBILIGHT 2011, LNICST 81, pp. 48–66, 2012.

mode. The PHY layer algorithms of such experimental MIMO testbeds are commonly designed with the help of system-modelling tools such as Matlab, whereas the rest of the testbed comprises Commercial-Off-The-Shelf (COTS) equipment that are responsible for the Radio Frequency (RF) signal up and down conversion and the signal synthesis and acquisition (i.e., Digital to Analog Conversion, DAC, and Analog to Digital Conversion, ADC). Although this implementation approach enables the rapid prototyping, it is lacking an insight to the implementation complexity and limitations that characterize the deployment of a real-time MIMO-OFDM system. Moreover, the focus of the offline baseband implementations is limited to the analysis of static testing scenarios (i.e., real-time system adaptivity is not feasible), which is commonly constrained to processing short data frames. Hence, offline MIMO testbeds are not suitable for the implementation of high capacity systems whose response has to be adapted according to dynamically changeable parameters or features. This consequently renders the MIMO testbeds that operate in offline mode unsuitable for exploring systems employing Adaptive Modulation and Coding (AMC) or closed-loop communication schemes. On the other hand, real-time testbeds allow the validation and analysis of such systems in realistic environments accounting for both hardware limitations and software or low-level programming constraints. The massive parallelism required for the baseband signal processing in real-time MIMO-OFDM testbeds, makes the Field Programmable Gate Array (FPGA) devices an appropriate candidate for implementing such systems.

This paper presents the PHY-layer design, implementation and validation of a high performance real-time MIMO-OFDM transmitter that was deployed using FPGA technology (based on the IEEE 802.16e-2005[1] standard). The transmitter is able to host two transmit antennas, features a 20 MHz bandwidth and uses matrix A encoding based on Alamouti's Space-Time Block Code (STBC)[2] on a per carrier basis. The PHY-layer algorithms of the transmitter were modelled in Matlab, designed in VHDL and implemented using a real-time FPGA platform. The GEDOMIS® (GEneric hardware DemOnstrator for MIMO Systems) testbed has facilitated the functional validation of the transmitter under realistic operating conditions. GEDOMIS® is a multi-antenna wireless communication testbed that enables the prototyping and evaluation of MIMO PHY algorithms. In the past, it has been used for the PHY-layer implementation of a real-time MIMO-OFDM mobile WiMAX receiver [3]. The transmitter in that case was emulated by using two Vector Signal Generators (VSG; containing an arbitrary waveform generator), each one of which was loaded with a Matlab-based vector file. These files are the output of a Matlab model that emulates the baseband algorithms of the MIMO-OFDM mobile WiMAX transmitter.

The principal issue that this paper aims to address is the absence in the literature of high bandwidth real-time implementations of MIMO-OFDM mobile WiMAX systems that account for low-level implementation issues and signal impairments. Thus, the work presented herein underlines the complexity of such an undertaking in the design, verification and debugging stages. At the same time, our work offers an insight into the critical design issues that should be accounted in similar real-time FPGA-based deployments. Particular focus is given on the impact of the selected high bandwidth in the implementation of the baseband signal processing algorithms in terms of memory utilization and the interfacing of the baseband processing components with the DACs. In fact, the 20 MHz channel bandwidth is one of the key changes that is introduced in the WiMAX 2 standard (IEEE 802.16m), to provide compliance with the prerequisites of fourth generation (4G) wireless communication systems.

2 Related Work

A great number of MIMO testbeds that were encountered in the literature are exclusively assembled by COTS equipment [4,5,6]. Nevertheless, these otherwise excellent development and testing environments do not allow the in-depth evaluation of the PHY-layer algorithms. This is due to the fact that their baseband part is a black box that cannot be modified or extended, limiting by this way their overall scope (i.e., conducting measurement campaigns and performance benchmarking). The PHY-layer implementation of our MIMO-OFDM mobile WiMAX transmitter can be easily expanded to include future WiMAX features or apply experimental PHY-layer concepts.

In the case of MIMO testbeds having custom PHY-layer implementations, it is important to make a distinction between the baseband signal processing deployments that operate in real-time and those that feature off-line functionality. The latter are software-based (e.g., Matlab) baseband implementations [7,8,9] which, as already mentioned before, are not suitable for validating certain testing scenarios that require adaptive signal processing. Also, the assessment of their deployment feasibility is impaired by the absence of real-world baseband signal processing implementation. On the other side, our MIMO-OFDM mobile WiMAX transmitter goes beyond the modelling of PHY-layer algorithms in Matlab; that is for, apart from the challenging task of mapping the algorithms to custom VHDL code, the system-deployment included the board-level code integration (e.g., interpolation filters, DAC ICs programming) and the real-time debugging under realistic laboratory conditions. The latter implied the use of heterogeneous equipment with critical system integration features such as a MIMO-enabled channel emulator.

Various authors have presented papers regarding real-time PHY-layer implementations of WLAN and LTE MIMO tranmsitters [15,16,17,18]. However, we have encountered relative few mobile WiMAX-related PHY-layer implementations available in the literature. In most of the cases the work described in such papers mainly focuses on partial FPGA implementations of specific mobile WiMAX PHY-layer algorithms, that either lack real-time operation or do not evaluate mobile MIMO channels (i.e., absence of either field testing results or laboratory results using a channel emulator). In [10] a low bandwidth FPGA-based 2x2 MIMO fixed WiMAX testbed is used to carry out a channel measurement campaign in order to analyse its capacity. The authors in [11] implemented in a FPGA a 2x2 MIMO Fixed Sphere Decoder (FSD), while the rest of the

PHY-layer was developed in a Matlab model, allowing by this way rapid co-simulations with different FFT sizes. Another FPGA-based implementation of a sphere detector and a channel matrix pre-processor for a 4x4 MIMO system is presented in [12]. In [13] a 2x2 MIMO system with 10 MHz bandwidth is implemented using a cell processor, with the aim to provide a single chip baseband implementation of the IEEE 802.16e OFDMA PHY for a base station transceiver. This system cannot be accurately validated, since it is not accounting for an appropriately applied channel model. Finally, a 2x2 MIMO system targeting a FPGA implementation is presented in [14]. The performance of the baseband transceiver implementation cannot be assessed in the absence of important system characteristics since it mainly presents partial simulation or emulation results.

3 Design of the Real-Time Transmitter

3.1 System Features

The IEEE 802.16e-2005 standard specifies a system for combined fixed and mobile broadband wireless access. The specified PHY supports scalable OFDMA architectures, AMC, various subchannelization permutation techniques and MIMO-aided transmit/receive diversity. However, our system is only adopting a fixed subset of this flexible configuration. In short, the main system parameters are as follows: 2 transmit and 2 receive antennas, RF frequency at 2.595 GHz, Intermediate Frequency (IF) at 67.2 MHz, fixed channel bandwidth of 20 MHz and QPSK modulation. It has to be noted that although this paper presents a MIMO-OFDM mobile WiMAX transmitter that supports only the Partial Usage of Subchannels (PUSC) permutation scheme, the AMC permutation scheme has also been implemented and validated for an equivalent SISO system, but it is not presented herein. It is also useful to mention that channel coding has not been included in our system. The WiMAX signal comprises various frames, which encapsulate user data, separated by equivalent silence periods. The OFDM frame used in our testbed is composed of a single burst with a fixed predefined format (i.e., the FCH and DL-MAP are not used).

3.2 Design, Implementation and Validation Methodology

The challenging inter-operability issues related to setting-up, calibrating and configuring a high-capacity real-time MIMO testbed require an efficient design, implementation and system-validation strategy. Hence, we present in this section a development-flow that was iteratively improved during the deployment of the present system.

The first vital step was to define the specifications of the transmitted signal and the respective channel models. After the completion of this stage, the initial selection of the MIMO signal processing algorithms was conducted, together with the definition of the basic system architecture. Another important

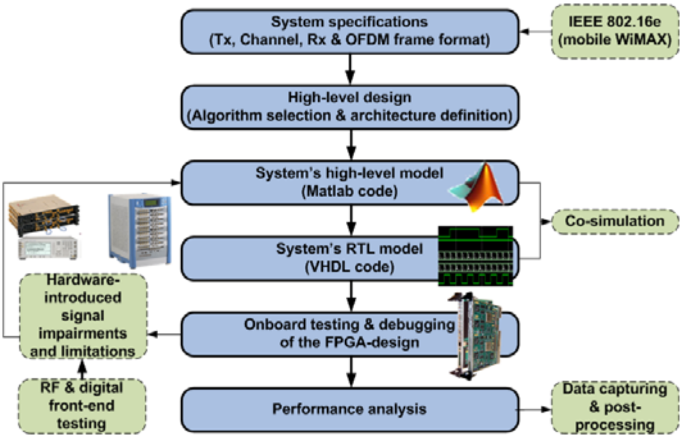

System specifications
(Tx, Channel, Rx & OFDM frame format)
IEEE 802.16e
(mobile WiMAX)
High-level design
(Algorithm selection & architecture definition)
System's high-level model
(Matlab code)
Co-simulation
System's RTL model
(VHDL code)
Hardware-introduced signal impairments and limitations
Onboard testing & debugging of the FPGA-design
RF & digital front-end testing
Performance analysis
Data capturing & post-processing

Fig. 1. Proposed development flow for a real-time MIMO testbed

step involved the estimation of the computational complexity of the PHY-layer algorithms (i.e., having as a reference the target FPGA device). The result of this estimation was used to re-adjust the scalability of certain system features.

A: Emulated transmitter (high-level system modelling)
This first evaluation stage was significant because it allowed us to start developing a precise Matlab model of the system, where the fixed-point logic was emulated by applying specific quantizations. This model was a vehicle to rapidly verify the behaviour of the designed architecture. The selected algorithms were optimized according to their estimated computational complexity. The GEDOMIS® testbed facilitated the rapid validation of the Matlab model. In more detail, the output of the baseband Matlab model of the MIMO-OFDM mobile WiMAX transmitter was stored in two files and uploaded in two VSGs, appropriately configured for MIMO signal transmission. The VSGs were then used to play back in real-time the MIMO signal and up-convert it to the desired RF frequency (i.e., 2.595 GHz). Next, the MIMO signal was fed to the channel emulator and finally it was demodulated using a multichannel oscilloscope that is remotely accessed by a certified mobile WiMAX software package (i.e., Agilent VSA with the IEEE 802.16e-2005 option). After validating the conformance with the mobile WiMAX standard, the receiver's data acquisition board was used to capture again the received signal (i.e., higher ADC resolution compared to the oscilloscope data-captures).

B: Real-time FPGA-based MIMO-OFDM signal transmission
Once the Matlab model satisfied the defined set of system requirements, every processing block of this model was mapped to RTL code (i.e., VHDL model). A very important stage of the design-validation flow was the Matlab/VHDL co-simulation. The latter offers a reliable comparison method that proves the

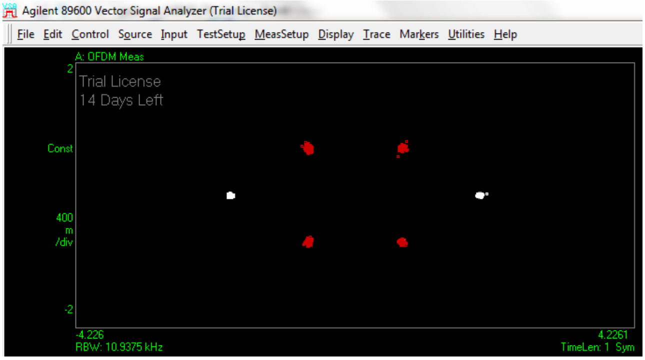

Agilent 89600 Vector Signal Analyzer (Trial License)
File Edit Control Source Input TestSetup MeasSetup Display Trace Markers Utilities Help
A: OFDM Meas
2
Trial License
14 Days Left
Const
400
m
/div
-2
-4.226
RBW: 10.9375 kHz
4.2261
TimeLen: 1 Sym

Fig. 2. Demodulation of the MIMO signal transmitted by CTTC's real-time FPGA-based MIMO-OFDM mobile WiMAX transmitter, using Agilent's VSA signal analysis software

functional and behavioural validity of the RTL-model and provides an assessment of the fixed-point implementation's precision.

The following stage included the implementation of the VHDL design targeting a specific FPGA-based processing platform. This implied the interfacing of the MIMO-OFDM mobile WiMAX transmitter with various on-board processing components using VHDL code (e.g., DACs, SDRAMs). Once the FPGA binary executable was produced, we were able to test the MIMO-OFDM mobile WiMAX transmitter using the entire equipment set-up of the GEDOMIS® testbed. A similar mobile WiMAX standard conformity test was conducted following the steps described before. However, this time the Matlab model was replaced by the real-time FPGA-based implementation of the equivalent baseband algorithms (a snap-shot of the demodulated signal is seen in figure 2). It has to be underlined that the same MIMO-OFDM mobile WiMAX signal generated by our FPGA-based transmitter, it was demodulated using the respective receiver, which we have developed previously [3] (figure 9 shows the demodulated data in a QPSK constellation). The real-world FPGA-based implementation of the MIMO-OFDM mobile WiMAX transmitter was debugged on the fly through vendor-specific tools and traces were captured to provide the final refinement of the system model.

C: Performance analysis

A final step of the presented design flow included the performance validation of the transmitter. Large buffers at the receiver side were used to capture data. This facilitated a detailed off-line analysis and characterization of the system performance. The extracted real-world performance metrics can be compared to the ones achieved by the Matlab/VHDL co-simulations, providing a quantification of the losses introduced by the hardware and by the RTL-implementation.

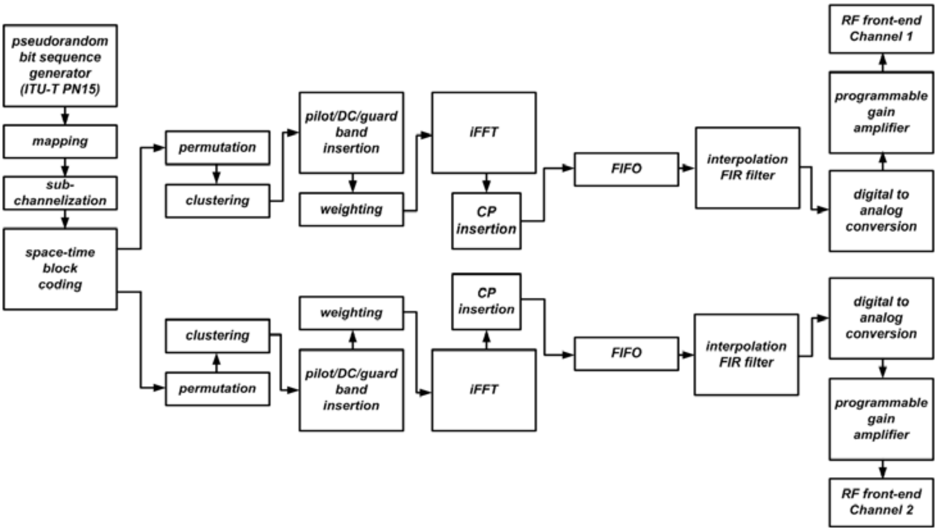


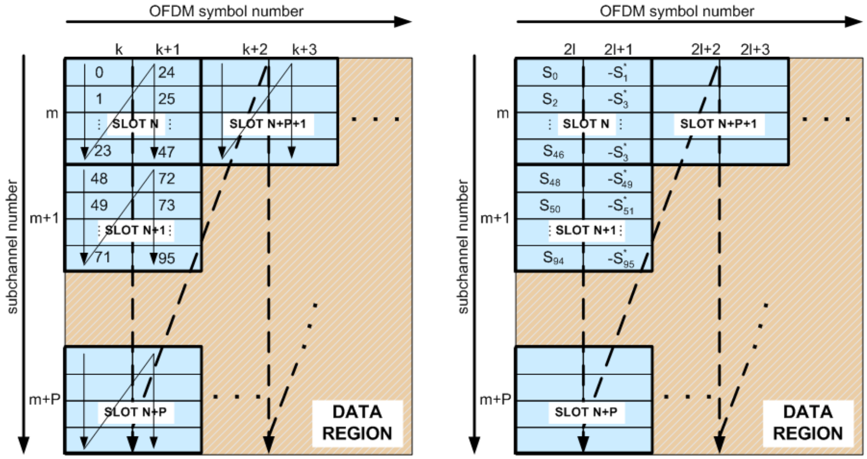
Fig. 3. General architecture of the real-time mobile WiMAX MIMO transmitter

3.3 General Architecture of the Transmitter

The general architecture of the system is depicted in figure 3. The first processing block of the real-time mobile WiMAX MIMO transmitter is a PseudoRandom Binary Sequence (PRBS) generator (i.e., based on the ITU-T PN15 specification [19]). Since our OFDM-frame is constructed by 46 data-symbols (using the PUSC scheme), each containing 2880 bits, the generated sequence will be composed of 132480 pseudo-random bits. The modulator then maps the generated sequence into QPSK constellation points.

At this stage, it is important to distinguish the operations related to the mobile WiMAX standard with those signal processing operations that are commonly encountered in MIMO-OFDM transmitters. The first type of signal processing operations define OFDM symbols that make use of the PUSC scheme according to the IEEE 802.16e standard:

- *Subchannelization*: the subchannels are created at the output of the modulator by distributing the sequence of QPSK symbols in two consecutive OFDM symbols in an interleaved fashion. This means that an adjacent group of 24 complex values is allocated in the first symbol and another one is allocated in the second symbol; this process is recursively repeated until the whole capacity of both symbols is filled up. The subchannelization process continues with the next two symbols, until the whole OFDM frame is constructed. The subchannelization process is shown in figure 4a.
- *Permutation*: logical structures of 12 adjacent subcarriers, named clusters, are created by scrambling the outputs of the STBC processing component. Additionally, the subchannels are grouped in 6 larger structures, namely Major Groups (MG) that have to be taken into account during this permutation



(a) Allocation of QPSK symbols to subcarriers and subchannels.

(b) Allocation of the data symbols for transmit antenna 1 according to matrix A (i.e., similar for transmit antenna 2).

Fig. 4. Mobile WiMAX standard-related operations applied on PUSC structured OFDM symbols

process. The mobile WiMAX standard defines the following permutation formula for the PUSC-structured OFDM symbols:

$$scr(k, s) = N_{sch} \cdot n_k + permbase[(s + n_k) mod N_{sch}], \qquad (1)$$

where *scr* is the index of the subcarrier within the OFDM symbol, $n_k = (k + 13s) mod 24$, N_{subch} is the number of subchannels within the MG (i.e., it can be either 8 or 12, depending on the parity of the MG's index), *permbase* is a predefined subchannel permutation mask (i.e., defined for both lengths of the MG), s is the subchannel index within the MG, and k is the subcarrier index within the subchannel (i.e., up to 24).

- *Clustering*: the previously created clusters are permuted according to a predefined renumbering sequence (i.e., interchanging of adjacent subcarrier groups).
- *Pilot/DC/guard-band insertion*: two pilot subcarriers are inserted to each cluster in given positions that depend on the parity of the OFDM symbol's index. The DC is inserted in the position 1024 of each OFDM symbol. Finally, two sets of null-carriers (with fixed-length) are inserted in the beginning and in the end of each symbol of the OFDM frame.
- *Weighting*: an additional subcarrier-randomization is applied (i.e., some subcarriers will be inverted), using the PRBS generator specified by the WiMAX standard, which applies a homogeneous distribution of the transmitted power.

The remaining of the signal processing operations at the transmitter are commonly found in MIMO-OFDM implementations. These include the inverse FFT that transforms the frequency-domain signal to a time-domain signal, and the Cyclic Prefix (CP) insertion, which complements each OFDM symbol in the frame (i.e., a portion of each symbol is repeated in the end of each symbol). A fundamental feature of the system is implemented in the STBC processing component, which allocates the carriers for each transmit antenna according to the matrix A configuration specified in the mobile WiMAX standard, as shown in figure 4b. This matrix for the case of two antennas is given by:

$$A = \begin{bmatrix} S_N & -S_{N+1}^* \\ S_{N+1} & S_N^* \end{bmatrix}, \tag{2}$$

where $S_1, S_2, ..., S_{66240}$ is the single stream of complex symbols at the output of the subchannelization block; the columns represent consecutive OFDM symbols (i.e., time) and the rows represent the transmit antennas (i.e., spatial streams).

After the insertion of the CP, several OFDM symbols must be stored in a large First In First Out (FIFO) memory in order to provide an uninterrupted flow of data to the DAC device. Although it is not shown in figure 3, dedicated-logic has been designed to control the synchronous communication between the different processing components. In other words, this logic is responsible for counting the length of each inter-frame silence period (during which null-carriers will be transmitted) and trigger the FIFO to provide the stored symbols in a strict timing manner. An additional OFDM symbol (i.e., preamble) must be inserted to facilitate the synchronization process at the receiver. The preamble is calculated off-line, stored in a RAM memory and read just before the first FIFO outputs are available.

3.4 Intelligent Memory System

The real-time implementation of the MIMO-OFDM mobile WiMAX transmitter requires a new bit to be generated at each clock cycle (i.e., a new QPSK symbol is generated each two clock cycles). At the same time, both the generic MIMO-OFDM processing stages and the operations related to the WiMAX standard are applied to the generated symbols. The implementation complexity of all these signal processing operations (e.g., symbols need to be permuted, inverted and grouped in logical structures) and the high bandwidth that was selected (i.e., 20 MHz, which corresponds to an FFT size of 2048 points), imposed a pipelined memory-structure with high throughput. The latter involved the storage of large data-sets in numerous intermediate stages that require low-latency interfacing. This was achieved by designing a generic memory structure for each intermediate stage in the processing chain, comprising an advanced memory controller and an adaptive memory block as shown in figure 5.

The operation of the transmitter involves the storing of large sequences of complex values that comprise two 16-bit words (i.e., real and imaginary parts - in-phase and quadrature, I&Q, components). The length of this sequence varies according to the operations applied in each processing stage (i.e., additional subcarriers are inserted in various stages of the baseband processing chain). Therefore, the first step towards the design of a generic memory structure was the

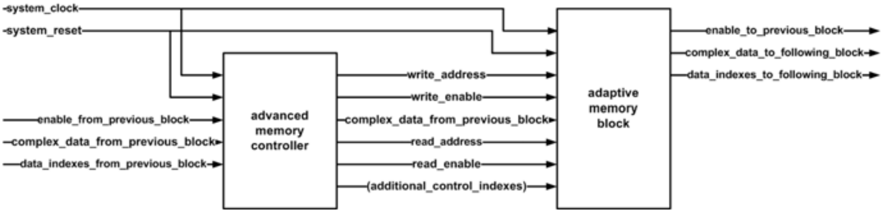

system_clock
system_reset
enable_from_previous_block
complex_data_from_previous_block
data_indexes_from_previous_block
advanced memory controller
write_address
write_enable
complex_data_from_previous_block
read_address
read_enable
(additional_control_indexes)
adaptive memory block
enable_to_previous_block
complex_data_to_following_block
data_indexes_to_following_block

Fig. 5. Generic memory structure for each signal processing stage

implementation of adaptive memory blocks. These include a mechanism that enables the adjustable instantiation and grouping of numerous FPGA RAM-block primitives as a single memory entity, offering by this way a simplified interface for seamless memory-access. Another important benefit of this mechanism is that it allows the simultaneous read and write operations of the OFDM-symbols, minimizing the memory-access latency and optimizing the usage of the embedded RAM blocks of the target FPGA device. The adaptive memory block includes additional functionality that enables the concurrent execution of complementary baseband operations with the storage and retrieval of the data sequences (featuring minimal latency). This top-up functionality is triggered by the advanced memory controllers using dedicated control signalling.

The operations related to the mobile WiMAX standard are basically devoted to reordering the data at each processing stage. The advanced memory controllers are therefore abstracting away the complexity of the required access to the embedded-memory elements. This is achieved by providing transparent memory read and write operations that allow the processing elements of the transmitter to continue performing data-manipulation operations. Moreover, the synchronous control of the FPGA RAM blocks composing each adaptive memory block is not introducing any impairment to the distinct processing stages. Thus, the main function of the advanced memory controllers is to optimize the read and write operations by implementing the required scrambling-operations with the minimum possible latency. During the write operations the symbol-reordering is applied with no latency by storing each incoming QPSK symbol in the indicated position. The latter is calculated on-the-fly according to the description given by the mobile WiMAX standard for the respective operation. For instance, the value of the index received with each symbol's complex value will be introduced together with additional control indexes (that are automatically generated), to a dedicated logic-component that implements the permutation formula. The result of this operation indicates the position where the given complex value needs to be stored (i.e., position within the OFDM symbol). Once a whole data stream (i.e., 1440 QPSK symbols for a PUSC-structured OFDM symbol) has been stored, the read operation will be triggered simultaneously with the next write operation.

It has to be noted that the read operations take into account the interrelations between the different OFDM symbols in a frame (i.e., in the PUSC permutation scheme, the pilot-tone distribution is applied in groups of two OFDM symbols). This in more detail means that the read operations are used to apply additional functions to the QPSK symbol sequence, before reaching the following stage. These functions include additional scrambling, zero latency subcarrier insertion (i.e., pilots, DC, guard-band), subcarrier weighting, insertion of the CP after the inverse FFT (i.e., allowing cyclic memory reading) and cyclic shift of the data-symbols inserted in the FIFO (i.e., the DC carrier is to be found in position 0). As previously mentioned, additional dedicated logic, memory-initialization values or predefined masks have been included within the adaptive memory blocks, accommodating by this the latter functionality.

4 Implementation and Integration

4.1 Testbed Setup

A graphic-overview of the GEDOMIS® testbed setup featuring a 2x2 MIMO configuration is shown in figure 6. The whole transceiver is implemented in a high-end signal processing development platform equipped with Lyrtech's DAC, ADC and FPGA/DSP boards. The VHS-DAC board features 8 phase synchronous channels with 14-bit DAC devices. The board also includes a Xilinx Virtex-4 FPGA device that hosts the complete PHY-layer of the real-time MIMO-OFDM mobile WiMAX transmitter. Similarly, the VHS-ADC board, features 8 channels with 14-bit ADC devices and a Xilinx Virtex-4 FPGA device that partially hosts the PHY-layer implementation of our MIMO-OFDM mobile WiMAX receiver. The signal acquisition board also includes a digitally controlled Programmable Gain Amplifier (PGA) for each channel, facilitating by this the development of automatic gain control algorithms. Finally, the SignalMaster Quad board has four TSM320C6416 DSPs and two Xilinx Virtex-4 FPGAs. One of these FPGAs is hosting the remaining signal processing blocks of the receiver.

The IF outputs generated by the DAC components of the MIMO-OFDM mobile WiMAX transmitter are fed in two ESG4438C VSG instruments, which up-convert the signal and provide the RF output centered at 2.595 GHz. The Elektrobit Propsim C8 radio channel emulator, allows accurate multi-channel emulation of custom or standardized models at the laboratory. This is feasible by adding complex and time-varying effects of multipath and Doppler-shifts in the digital domain (e.g., adjusting the tap amplitude, delay spread, operation frequency and mobile speed). For the 2x2 MIMO scenario considered in this paper, four uncorrelated multipath fading channels were created (with different distribution seeds), using either the ITU Vehicular A or the ITU Pedestrian B channel model [20]. The receiver's RF front-end from Mercury Computer Systems (Echotek Series RF 3000 Tuners) comprises 1 Direct Digital Synthesizer (DDS) and 4 receiver modules. This instrument features high spectral purity and dynamic range and applies phase-coherent down-conversion of the RF signal to the system's IF frequency. Finally, two uncorrelated broadband RF noise generators provide extremely flat white noise at IF and facilitate the measurement campaign and the assessment of system's performance under variable SNR conditions (e.g., the two noise sources were calibrated and balanced for every 2dB attenuation step).

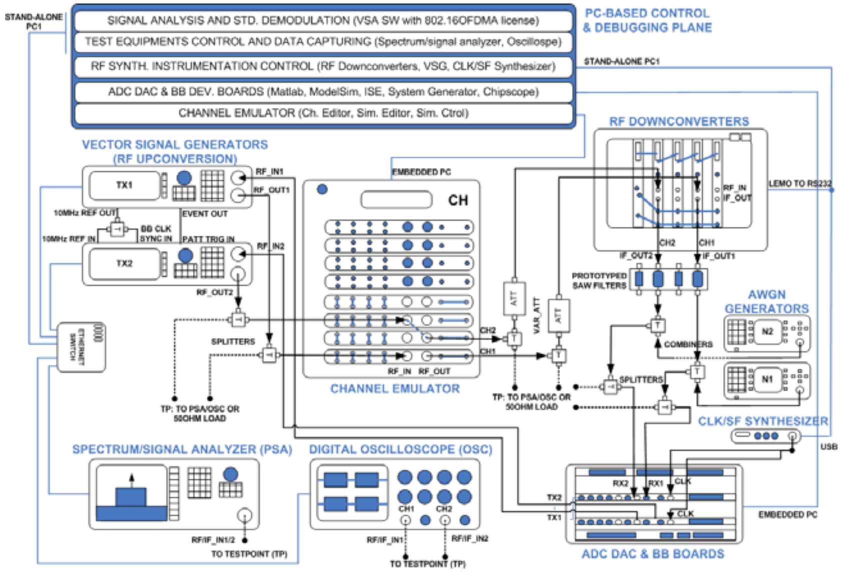

STAND-ALONE PC1
SIGNAL ANALYSIS AND STD. DEMODULATION (VSA SW with 802.16OFDMA license)
TEST EQUIPMENTS CONTROL AND DATA CAPTURING (Spectrum/signal analyzer, Oscillospe)
RF SYNTH. INSTRUMENTATION CONTROL (RF Downconverters, VSG, CLK/SF Synthesizer)
ADC DAC & BB DEV. BOARDS (Matlab, ModelSim, ISE, System Generator, Chipscope)
CHANNEL EMULATOR (Ch. Editor, Sim. Editor, Sim. Ctrol)
PC-BASED CONTROL & DEBUGGING PLANE
STAND-ALONE PC1
VECTOR SIGNAL GENERATORS (RF UPCONVERSION)
TX1
RF_IN1
RF_OUT1
10MHz REF OUT
EVENT OUT
BB CLK SYNC IN
10MHz REF IN
PATT TRIG IN
TX2
RF_IN2
RF_OUT2
ETHERNET SWITCH
SPLITTERS
TP: TO PSA/OSC OR 50OHM LOAD
EMBEDDED PC
CH
CH2
CH1
RF_IN
RF_OUT
CHANNEL EMULATOR
ATT
VAR_ATT
ATT
TP: TO PSA/OSC OR 50OHM LOAD
RF DOWNCONVERTERS
RF_IN
IF_OUT
LEMO TO RS232
CH2
CH1
IF_OUT2
IF_OUT1
PROTOTYPED SAW FILTERS
AWGN GENERATORS
N2
N1
COMBINERS
SPLITTERS
CLK/SF SYNTHESIZER
USB
SPECTRUM/SIGNAL ANALYZER (PSA)
RF/IF_IN1/2
TO TESTPOINT (TP)
DIGITAL OSCILLOSCOPE (OSC)
CH1
CH2
RF/IF_IN1
RF/IF_IN2
TO TESTPOINT (TP)
RX2
RX1
CLK
TX2
TX1
CLK
EMBEDDED PC
ADC DAC & BB BOARDS

Fig. 6. The GEDOMIS® testbed setup

4.2 Baseband and DAC Circuitry Integration

Once the baseband RTL design of the MIMO-OFDM mobile WiMAX transmitter was verified (VHDL versus Matlab co-simulations), a detailed study was conducted to provision its integration with the DAC circuitry of the target boards.

In order to produce the global synthesized IF (i.e., 67.2 MHz) and take the maximum advantage of the filters contained in the DAC devices of Lyrtech's VHS-DAC board, an appropriate filter-interpolation strategy was selected. Hence, considering that a total x16 interpolation is required for each of the two I&Q data-streams of the transmitter, it was selected to apply a x2 interpolation in the FPGA domain and a x8 interpolation in each dual DAC device.

In more detail, the I&Q 16-bit data at the output of the FIFOs shown in figure 3 are fed to four separate filters (hosted in the FPGA device), which apply an interpolation by two. The filters are clocked at 44.8 MHz (i.e., the baseband clock is at 22.4 MHz). The use of two separate filters for the I&Q data components of each antenna was made to avoid the multiplexing of data, which requires a double clock rate from the one currently used at baseband (i.e. increasing the input clock from 44.8 MHz to 89.6 MHz). This decision was made after realizing that what was gained in terms of FPGA slice area (by using two instead of four interpolation filters clocked at 89.6 MHz), was resulting in a complicated place-and-route process, which is prone to timing errors (i.e., the FPGA routing algorithms are stressed to the limits for achieving the desirable clock rate). Manual placing of the implemented logic to the FPGA device area is another design option that is equally complicated when considering large FPGA design with numerous digital processing components.

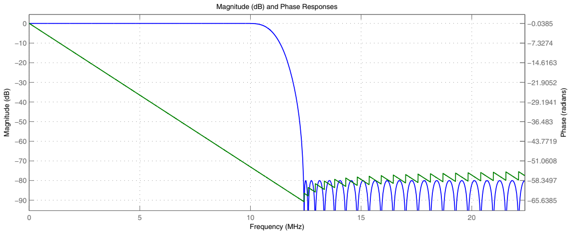


Fig. 7. The magnitude and phase response of the x2-interpolation pre-DAC interpolation filters used in the transmitter

Each dual DAC device, from the other side, is using 3 filters that are applying a x8 interpolation, which results in a 358.4 MHz clock (i.e., x8 FMIX QMC CMIX [21]). Taking into account the pass-bands of the different interpolation filters within the DAC, the frequency response of the inverse-sinc filter, and the frequency bands of the signals and their aliases, the coarse mixer was set to 89.6 MHz (equal to Fs/4, where Fs 358.4 MHz, i.e., the sampling frequency of the DAC) and the fine mixer was set to -22.4 MHz. The registers corresponding to the NCO frequency, the fine mixer gain and the coarse mixer mode-configuration (i.e., cm_mode(3:0) = 1000) were programmed through a C-based application. The latter allows the read/write access of several internal registers of the DAC 5687 chips on-the-fly.

The choice of the interpolation filters was a significant design, implementation and system-integration decision, which in fact has to be carefully considered every time a designer of MIMO-OFDM transmitters intends to interface the baseband design with a DAC device. The filter interpolation is an excessively processing-consuming implementation for FPGA devices, having a very hard to define trade-off between of FPGA resources utilization (embedded memory blocks and DSP48s versus generic FPGA slices) and maximum achievable clock rate. The heavily populated FPGA designs with multiple clock domains are making the place-and-route process of the FPGA implementation-flow, a particularly hard task. Therefore, a key system-design objective is to migrate the challenging implementation of the interpolating filters from the FPGA domain to high performance versatile DAC devices. In our case for instance, it would have been impossible to implement a higher order interpolation in the available FPGA area (perhaps, this is feasible using the largest Xilinx Virtex-6 devices).

The filters were designed using the FDAtool of Matlab having a high rejection response (i.e., 80 dB). This in turn implied a high number of filter coefficients, which ultimately is translated to high computational cost. The filter coefficients are featuring a symmetric response (i.e., 76 coefficients) and the produced file was used to configure the four respective Xilinx IP cores (i.e., Finite Impulse Response (FIR) filter). Distributed arithmetic logic has been selected for implementing the internal filter computational architecture, because the multiply and accumulate filter implementation option is requiring a high number of embedded FPGA RAM blocks (i.e., RAMB16s). This implementation decision was taken

Table 1. The main FPGA utilization metric corresponding to the PHY-layer implementation of the real-time MIMO-OFDM mobile WiMAX transmitter

Device utilization summary			
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slice Flip Flops	76,598	135,168	56%
Number of 4 input LUTs	69,189	135,168	51%
Logic Distribution			
Number of occupied Slices	52,650	67,584	77%
Total Number of 4 input LUTs	80,466	135,168	59%
Number of FIFO16/RAMB16s	285	288	98%
Number of DSP48s	81	96	84%
Number of DCM_ADVs	8	12	66%
Total equivalent gate count for design	20,565,470		

considering the total number of RAMB16s, DPS48s and slices available in the Virtex-4 LX160 device and the internal processing architecture of the MIMO-OFDM mobile WiMAX transmitter. The latter has several intermediate stages that require large data-set storage (i.e., high RAMB16s utilization). Finally, the 28-bit output of the I and Q filters had to be appropriately truncated to allow the utilization of the full dynamic range of the baseband part of the transmitter. This was achieved by running simulations of the whole MIMO mobile WiMAX transmitter. The truncated 14 bits are then interfaced with the DAC device of the Lyrtech board (the board firmware utilizes a 2 bit back-off margin).

4.3 FPGA Resource Utilization

The design of the transmitter has passed through numerous optimization stages to improve the performance and minimize the FPGA resource utilization, when targeting the Virtex-4 LX-160 device and using the Xilinx ISE 9.2 (i.e., mandatory software version for maintaining compatibility with the target board's firmware). The FPGA implementation was proved to be quite challenging since the chosen arithmetic-logic implementation of the four interpolation filters residing in the FPGA domain is making exclusive use of FPGA slices. Therefore, a study of the appropriate synthesis, mapping and place-and-route options was conducted trying to balance the implementation trade-offs, in terms of design speed (e.g., meet timing requirements) and FPGA area utilization. The place-and-route process resulted in significantly different FPGA implementation times, since the ISE 9.2 software tool is applying radically different execution routines.

As it is seen in table 1 the MIMO-OFDM mobile WiMAX transmitter is practically leaving very limited resources to the given Virtex-4 LX160 FPGA device for future extensions of the design. This could be partially solved with the migration of the whole design to the new Xilinx ISE 12 design suite. We have conducted some preliminary tests which show a reasonable reduction of the FPGA resource utilization. However, it has to be noted that the implementation

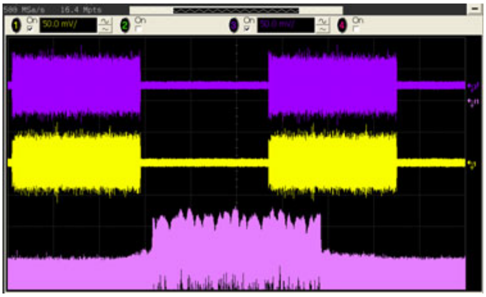


Fig. 8. Resulting MIMO time-signal, along with its spectrum (Agilent 80804B Infiniium Oscilloscope)

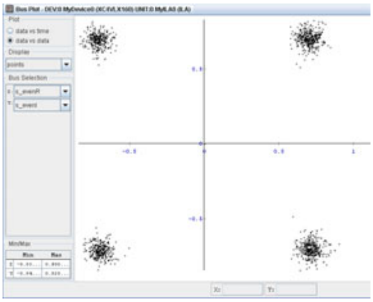


Fig. 9. Visualization of the demodulated data at the receiver side (Xilinx Chipscope Pro)

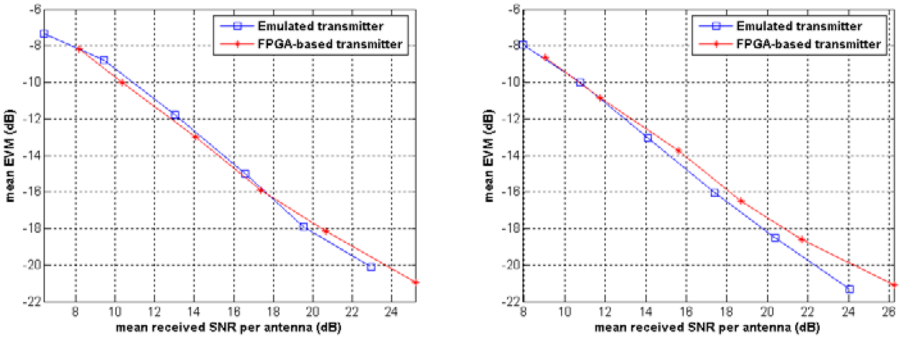
feasibility-analysis of this option is tightly coupled with an equivalent migration to a new FPGA board (i.e., out of scope for this paper).

5 Experimental Validation and Results

The resulting data at the output of the PHY-layer of the mobile WiMAX receiver is captured and visualized in real-time through vendor-specific tools and the Chipscope Pro software from Xilinx, using two different hardware configurations. In the first case, the emulated transmitter presented in [3] is used, while in the second case it is used the real-time FPGA-based MIMO-OFDM mobile WiMAX transmitter presented in this paper (see figures 8 and 9).

It is important to underline that the same VSGs are used to playback in real-time the Matlab-generated MIMO mobile WiMAX signal (i.e., PHY-layer emulation and baseband to RF signal up-conversion), and to up-convert to RF the IF signal generated by the real-time FPGA-based MIMO-OFDM mobile WiMAX transmitter. Therefore, the RF front-end performance is common for both testing scenarios. In other words, the deviations that may be observed when comparing the performance of both deployments have their origins in the baseband signal processing implementation (e.g., fixed-point arithmetic operations). Other indicative factors that could also result in differences during the result-comparison are the DAC features and the connector losses.

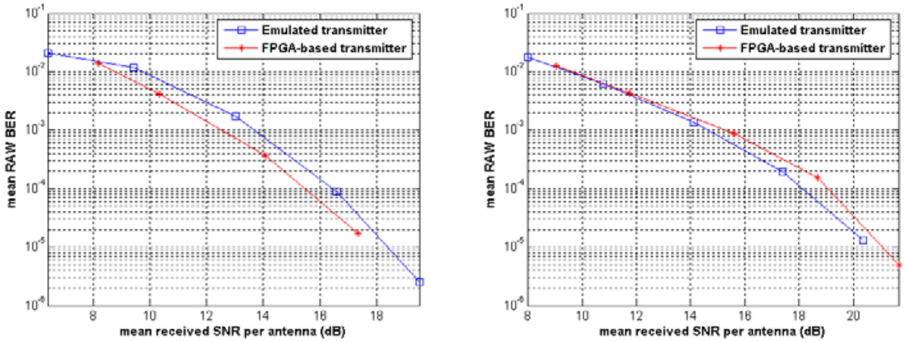
In order to analyse the performance of the system, the real-time channel emulator was configured with two static MIMO channel models (i.e., no mobility is emulated). These random channel models have the tap and delay spread characteristics, defined by the ITU Vehicular A and the ITU Pedestrian B specifications. The accuracy of the received constellation points is validated at the receiver by calculating the Error Vector Magnitude (EVM), as shown in figure 10. Additional assessment of the system's performance is provided by means of the raw Bit Error Rate (BER) calculation (i.e., no channel coding was used), as it can be seen in figure 11. Both performance metrics are calculated for a



(a) ITU-T vehicular A-based

(b) ITU-T pedestrian B-based

Fig. 10. System performance under static channel conditions: EVM-SNR curves



(a) ITU-T vehicular A-based

(b) ITU-T pedestrian B-based

Fig. 11. System performance under static channel conditions: BER-SNR curves

set of suitable Signal-to-Noise Ratio (SNR) values that cover a wide range of signal reception conditions (i.e., pre-defined white noise gain attenuation steps are used to generate uncorrelated white noise signals, which are added to each of the received signals). The real-time captures that have been performed are covering several entire frames, allowing by this way an averaged (and thus more robust) calculation of the presented metrics for each analysed scenario.

As it may be observed in the related figures, the performance for both deployments is quite similar. This not only proves the functional validity of the FPGA development, but also demonstrates the high implementation precision of the baseband signal processing algorithms. The minimal performance differences are due to the factors described next. First of all, the MIMO-OFDM mobile WiMAX transmitter is only validated under static channel conditions; this means that the performance comparison between the emulated transmitter and its FPGA-based counterpart is only analysed for a specific channel frequency response for each

Fig. 12. The MIMO-OFDM mobile WiMAX transmitter during a live demonstration (European Nanoelectronics Forum 2010)

channel model, which could represent a different performance condition among models at a given time (e.g., very low or very high attenuation). This frequency response would vary over time in mobility testing scenarios. In that case, the two system performance metrics are extracted by averaging numerous measurements corresponding to different channel seeds. This important testing scenario will be presented as a future work. Second, despite the fact that we tried to re-produce the exact testing conditions in both scenarios (e.g., transmitted power, noise levels), it is unavoidable that the hardware set-up features certain differences (e.g., number of RF/IF cables, DAC features, intermediate gain values) that influence the overall system's performance.

A limited setup of the full MIMO-OFDM mobile WiMAX transceiver is shown in figure 12. The real-time operation of the transmitter was demonstrated with success during the European Nanoelectronics Forum 2010.

6 Conclusions and Future Work

This paper presented the design, implementation and validation of the PHY-layer of a high-performance real-time MIMO mobile WiMAX transmitter featuring the Alamouti's STBC-based scheme. The system has been successfully validated under realistic channel conditions in static scenarios. The performance of the system has been analysed, by comparing the FPGA-based deployment to an emulated version of the transmitter (i.e., ideal baseband implementation). The presented work covers the most relevant issues to be taken into account, when confronting the complex task of designing, implementing and validating a real-time MIMO-OFDM system. The authors' contribution is mainly found in building and validating a realistic MIMO system based on the mobile WiMAX standard, having a high channel bandwidth of 20 MHz. These system specifications have scaled up the implementation complexity, but have offered at the same time an added-value to the whole undertaking.

The modular design of the system will be extended to include more advanced MIMO-OFDM schemes in the future. For instance, a real-time MIMO-OFDM closed-loop scheme could enhance the scalability and MIMO configuration options of the entire testbed. This will be implemented by using a feedback channel from the receiver to the transmitter. This real-time feedback link will contain quantized information for the actual channel conditions, which will enable an adaptive signal transmission. Finally, a broader analysis will be conducted in the future, to cover the performance of the system under the presence of mobility at high-speeds (e.g., ITU-T vehicular A up to 120 Km/h).

References

1. Mobile WiMAX air interface, P802.16Rev2/D9 (Revision of IEEE Std 802.16-2004 and consolidates material from IEEE Std 802.16e-2005, IEEE Std 802.16-2004/Cor1-2005)
2. Alamouti, S.M.: A simple transmit diversity technique for wireless communications. IEEE Journal on Selected Areas in Communications 16(8), 1451–1458 (1998)
3. Font-Bach, O., Bartzoudis, N., Pascual-Iserte, A., López Bueno, D.: A real-time MIMO-OFDM mobile WiMAX receiver: Architecture, design and FPGA implementation. Elsevier Journal of Computer Networks (2011) (in press), doi:10.1016/j.comnet.2011.02.018
4. De Bruyne, J., Joseph, W., Verloock, L., Olivier, C., De Ketelaere, W., Martens, L.: Field Measurements and Performance Analysis of an 802.16 System in a Suburban Environment. IEEE Transactions on Wireless Communications 8(3), 1424–1434 (2009)
5. Pelechrinis, K., Broustis, I., Salonidis, T., Krishnamurthy, S.V., Mohapatra, P.: Design and Deployment Considerations for High Performance MIMO Testbeds. In: 4th International Wireless Internet Conference (WICON), Hawaii (2008)
6. Imperatore, P., Salvadori, E., Chlamtac, I.: Path Loss Measurements at 3.5 GHz: A Trial Test WiMAX Based in Rural Environment. In: 3rd International ICST Conference on Testbeds and Research Infrastructure for the Development of Networks and Communities (TridentCom), Orlando (2007)
7. Wang, G., Yin, B., Amiri, K., Sun, Y., Wu, M., Cavallaro, J.R.: FPGA Prototyping of a High Data Rate LTE Uplink Baseband Receiver. In: 2009 Asilomar Conference on Signals, Systems and Computers (ASILOMAR), Monterey (2009)
8. Pérez-Iglesias, H.J., García-Naya, J.A., Dapena, A., Castedo, L., Zarzoso, V.: Blind Channel Identification in Alamouti Coded Systems: A Comparative Study of Eigendecomposition Methods in Indoor Transmissions at 2.4 GHz. European Transactions on Telecomunications. Special Issue: European Wireless 2007 19(3), 751–759 (2008)
9. Ramírez, D., Santamaría, I., Pérez, J., Vía, J., García-Naya, J.A., Fernández-Caramés, T.M., Pérez-Iglesias, H.J., González-López, M., Castedo, L., Torres-Royo, J.M.: A comparative study of STBC transmissions at 2.4 GHz over indoor channels using a 2 x 2 MIMO testbed. Wireless Communications and Mobile Computing 8(9), 1149–1164 (2008)
10. Gil Jiménez, V.P., Fernández-Getino García, M.J., García Armada, A., Torres, R.P., García Fernández, J.J., Sánchez-Fernández, M.P., Domingo, M., Fernández, O.: MIMO-OFDM Testbed, Channel Measurements, and System Considerations

for Outdoor-Indoor WiMAX. EURASIP Journal on Wireless Communications and Networking 2010 (2010)

11. Khairy, M.S., Abdallah, M.M., Habib, S.E.-D.: Efficient FPGA Implementation of MIMO Decoder for Mobile WiMAX System. In: IEEE International Conference on Communications 2009 (ICC), Dresden (2009)
12. Dick, C., Trajkovic, M., Denic, S., Vuletic, D., Rao, R., Harris, F., Amiri, K.: FPGA implementation of a near-ML sphere detector for 802.16e broadband wireless systems. In: SDR Technical Conference and Product Exposition 2009 (SDR), Arlington (2009)
13. Wang, Q., Fan, D., Lin, Y.H., Chen, J., Zhu, Z.: Design of BS transceiver for IEEE 802.16E OFDMA mode. In: IEEE International Conference on Acoustics, Speech and Signal Processing 2008 (ICASSP), Las Vegas (2008)
14. Wu, Y.-J., Lin, J.-M., Yu, H.-Y., Ma, H.-P.: A baseband testbed for uplink mobile MIMO WiMAX communications. In: IEEE International Symposium on Circuits and Systems 2009 (ISCAS), Taipei (2009)
15. Chen, J., Zhu, W., Daneshrad, B., Bhatia, J., Kim, H.S., Mohammed, K., Sasi, S., Shah, A.: A Real Time 4x4 MIMO-OFDM SDR for Wireless Networking Research. In: 15th European Signal Processing Conference (EUSIPCO), Poznań (2007)
16. Haene, S., Perels, D., Burg, A.: A Real-Time 4-Stream MIMO-OFDM Transceiver: System Design, FPGA Implementation, and Characterization. IEEE Journal on Selected Areas in Communications 26(6), 877–889 (2008)
17. Ibing, A., Kühling, D., Kuszak, M., Helmolt, C.v., Jungnickel, V.: Flexible demonstrator platform for cooperative joint transmission and detection in next generation wireless MIMO-OFDM networks. In: 4th International ICST Conference on Testbeds and Research Infrastructure for the Development of Networks and Communities (TridentCom), Innsbruck (2008)
18. Jungnickel, V., Schellmann, M., Thiele, L., Wirth, T., Haustein, T., Koch, O., Zirwas, W., Schulz, E.: Interference-Aware Scheduling in the Multiuser MIMO-OFDM Downlink. IEEE Communications Magazine 47(6), 56–66 (2009)
19. General requirements for instrumentation for performance measurements on digital transmission equipment, Rec. ITU-T O.150 (1996)
20. Guidelines for Evaluation of Radio Transmission Technologies for IMT-2000, Rec. ITU-R M.1225 (1997)
21. 16-bit, 500 MSPS 2x-8x Interpolating Dual-Channel Digital-to-Analog Converter (DAC), Texas Instruments, `http://focus.ti.com/docs/prod/folders/print/dac5687.html`