# PROTOTYPING PROCESSING-DEMANDING PHYSICAL LAYER SYSTEMS FEATURING SINGLE OR MULTI-ANTENNA SCHEMES

*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
[2]Department of Signal Theory and Communications, Universitat Politècnica de Catalunya (UPC)
Campus Nord, Jordi Girona 1-3, 08034 Barcelona, Spain
email: {ofont, nbartzoudis, dlopez}@cttc.cat and antonio.pascual@upc.edu

## ABSTRACT

In the past years numerous algorithms, schemes and techniques have been proposed, in order to improve the performance of the wireless multi-antenna communication systems. Real-time multi-antenna testbeds are offering the means to analyse the real-world performance, implementation cost and feasibility of such novel techniques, accounting for hardware limitations and software constraints. This paper presents the physical-layer (PHY) design, implementation and validation of a high-performance real-time mobile WiMAX transceiver, accounting for low-level deployment issues and signal impairments. A first evaluation of the acquired results for both Single Input Single Output (SISO) and Multiple Input Multiple Output (MIMO) system-configurations, demonstrates the performance of the system using real-time channel emulation.

## 1. INTRODUCTION

The deployment of real-time baseband prototypes featuring wide bandwidth and adaptive performance has always been a challenging task for designers of communication systems. Additionally, the inclusion of multi-antenna schemes is scaling the computational complexity in the PHY-layer, compared to the traditional SISO ones. As a consequence, the design and implementation stages should not be underestimated, even if the underlying algorithmic part derives from well-established scientific knowledge. This is because the deployment of processing-demanding baseband prototypes always requires a mixture of state-of-the-art signal processing techniques and leading edge applied research. Moreover, the laboratory-based debugging, verification and testing of real-time MIMO systems is a very costly and painstaking process, especially when including Radio Frequency (RF) front-ends and channel emulators (i.e., the system-wide considerations are increasing). This paper presents the baseband design and implementation of a transceiver based on the IEEE 802.16e-2005 standard (denoted as mobile WiMAX) featuring results of both SISO and MIMO system-configurations using static channels. The performance of the real-time FPGA-based transceiver is also compared with the equivalent Matlab model. The GEDOMIS® (GEneric hardware DemOnstrator for MIMO Systems) testbed has facilitated the functional validation of the transceiver under realistic operating conditions.

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.

### 1.1 Related work

One of the main motivations of this paper is the absence in the related literature of real-time baseband implementations of the complete mobile WiMAX PHY-layer (featuring MIMO-OFDM). This is due to the fact that the design and deployment of such real-world systems implies a considerable overhead in terms of design complexity and implementation cost. In this context, it is common that researchers choose to implement SISO mobile WiMAX systems, like the ones that will be detailed next. For instance, in [1] it is presented the performance of a FPGA-based PHY-layer implementation. The authors of the encountered papers are also limiting the scope of their research to non real-time deployments [5] of the PHY-layer of mobile WiMAX. Another common practice is to assembly testbeds using Commercial-Off-The-Shelf (COTS) components and hardware boards that enable the deployment of measurement campaigns [4]; nevertheless, this approach does not offer an insight to the design and implementation complexity of the PHY-layer. Various papers present results related to the characterization of the PHY-layer of mobile WiMAX by using system-simulations [2]. Another example of related work is given by the authors of [3], where the performance of a baseband transceiver deployed in a multi-core software defined radio (SDR) platform is demonstrated by software simulations. In any case, it is important to underline the fact that simulations are not taking into account various important implementation parameters, which are inherent to the bit-intensive nature of the baseband processing algorithms. Finally, we have encountered very limited sources of related work regarding MIMO-OFDM mobile WiMAX systems. These include partial PHY-layer FPGA implementations of specific baseband algorithms [9] or implementations featuring emulation results [7]. Arguably, the most related reference-work is presented in [6], where the authors implemented a 2x2 MIMO base station transceiver based on cell broadband engine.

The goal of this paper is to present the critical aspects that need to be considered, when designing and implementing a wide-bandwidth baseband prototype that features multiple antennas and operates in real-time. Our aim is also to underline the value of the whole hardware set-up that makes use of heterogeneous equipment with critical system-integration features, which account for low-level implementation issues and signal impairments.

## 2. FLEXIBLE DESIGN METHODOLOGY

The PHY-layer of the MIMO-OFDM mobile WiMAX transceiver was modelled in Matlab, designed with custom VHDL code and implemented using a real-time FPGA platform. The WiMAX standard compliance was verified by using a well-established testing methodology and certified software tools for mobile WiMAX. It is important to underline

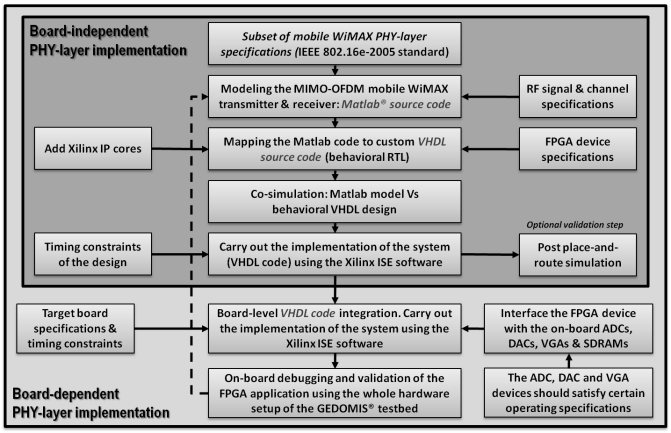



Figure 1: The design and implementation flow-chart.

that various additional instruments and equipment of the GEDOMIS® testbed [10] were used on top of the FPGA-based signal processing platform (please, refer to our previous work for a detailed description of the hardware setup). This enabled the validation and performance evaluation of CTTC's MIMO-OFDM mobile WiMAX prototype in close to real-world conditions. The focus of this paper is laid on the baseband design and the real-time implementation of a MIMO-OFDM mobile WiMAX transceiver, which included the hard task of board-level code-integration (e.g. AGC algorithm, interpolation filters, compensation of RF signal impairments, DAC ICs programming, digital down conversion, etc), as well as the real-time debugging and system-testing under realistic laboratory conditions. The design and implementation flow of the MIMO-OFDM mobile WiMAX transceiver is shown in figure 1. The adopted methodology is quite flexible allowing the re-utilization of the VHDL-based implementation in various hardware prototyping platforms. Moreover, the PHY-layer of our prototype can be easily expanded to include future mobile WiMAX features or to apply experimental PHY-layer concepts.

## 3. DEPLOYING THE MOBILE WIMAX TRANSCEIVER

The PHY-layer of the real-time mobile WiMAX transceiver features two transmit and two receive antennas and uses matrix A encoding based on Alamouti's Space Time Block Coding (STBC) [8] on a per carrier basis. The baseband signal processing architecture makes use of a wide channel bandwidth (20 MHz) and targets a high-capacity, pipelined FPGA-based architecture. The following sections will provide the design and implementation details of this system. Figure 2 is depicting the general architecture of the transceiver, containing as well various system-features and specifications (channel coding was not implemented).

### 3.1 The baseband part of the transmitter

The majority of the signal processing blocks found in the baseband part of the transmitter are responsible for formatting the OFDM symbols according to the mobile WiMAX standard. The 2x2 MIMO configuration uses only the Partial Usage of Sucbarriers (PUSC) permutation scheme, based on the following operations:

- *Subchannelization*: the subchannels are created by distributing adjacent groups of QPSK symbols between two consecutive OFDM symbols, in an interleaved fashion, until both symbols are filled up. This process is repeated to construct the whole OFDM frame.
- *Permutation*: logical structures of adjacent subcarriers, named clusters, are created by scrambling the outputs of the STBC processing component. Additionally, the subchannels are grouped in larger structures, namely Major Groups (MG). The mobile WiMAX standard defines the following permutation formula for the PUSC-structured OFDM symbols:

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

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, $permbase$ is a predefined subchannel permutation mask, $s$ is the subchannel index within the MG, and $k$ is the subcarrier index within the subchannel.
- *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 fixed-length sets of null-carriers are inserted in the beginning and in the end of each OFDM symbol.
- *Weighting*: an additional subcarrier-randomization is applied (i.e., some subcarriers will be inverted), using a PRBS generator, that homogenises the transmitted power distribution.

The STBC is an important processing component of the transmitter since it allocates the carriers for each transmit antenna according to the matrix A configuration specified in the mobile WiMAX standard, as follows:

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

where $S_1, ..., S_N$ is the single complex-symbol stream at the output of the subchannelization block; the columns represent consecutive OFDM symbols (i.e., time) and the rows represent the different transmit antennas (i.e., spatial streams).

#### 3.1.1 Digital to Analogue Conversion (DAC)

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 the FPGA board, it was selected to apply a x2 interpolation in the FPGA and a x8 interpolation in each dual DAC device. The interpolation filter strategy 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 (i.e., interpolation is an excessively processing-consuming operation).

### 3.2 Received signal model

The WiMAX signal is frame-based (i.e., user data is encapsulated within frames). Silence periods are inserted between the frames during which, the receiver is continuously monitoring the incoming signal in order to detect the beginning of the following frame. In a real-world MIMO-OFDM testbed like GEDOMIS®, the system designer should provide mechanisms to compensate the received RF signal-impairments that stem from the performance characteristics of the testbed's equipment (e.g., channel emulator, RF front-end, baseband signal processing boards). The performance of the system can be dramatically degraded if the received subcarriers lose their orthogonality. Therefore, it is vital to determine and remove the signal-impairments before making the symbol decisions.

In the present signal model, certain negligible signal-impairments have been ignored due to the advanced performance-features of the equipment composing the GEDOMIS® testbed (e.g., in-phase and quadrature (I/Q) gain and phase imbalances, inaccuracy between the hardware clocks in respect to the ideal sampling frequency, local oscillator (LO) drifts, random phase noise due to LO instability). Thus, the resulting received signal model at the output of the RF down-converters at the $i$th receive antenna can be expressed as:

$$c_i(t) = \Re\{x_i(t) \cdot e^{j2\Pi(f_{IF}+\Delta f)t}\} + w_i(t) + A_i + \\ +B_i \cdot cos(2\Pi(f_{IF} + \Delta f)t + \varphi_i), \quad (3)$$

where $x_i(t)$ represents the useful part of the received baseband signal, $f_{IF}$ is the Intermediate Frequency (IF), $\Delta f$ is the Carrier Frequency Offset (CFO), $w_i(t)$ is the Gaussian noise, $A_i$ is the Direct Current (DC) level introduced by the baseband board chassis, and finally, $B_i \cdot cos(2\Pi(f_{IF} + \Delta f)t + \varphi_i)$ represents the unwanted residual carrier, located at the center of the useful signal-spectrum (i.e., introduced by the LO coupling at the transmitter). The useful part of the received baseband signal at the $i$th receive antenna can be expressed as follows:

$$x_i(t) = \sum_{j=1}^{n_T} \tilde{x}_j(t) \star H_{i,j}(t), \quad (4)$$

where $\tilde{x}_j(t)$ is the equivalent baseband signal transmitted from the $j$th transmit antenna, with $n_T$ being the number of assumed transmit antennas, and $H_{i,j}(t)$ is the equivalent baseband of the time impulse response of the channel between the $j$th transmit antenna and the $i$th receive antenna.

### 3.3 The baseband processing blocks of the receiver

Before starting describing the MIMO-OFDM mobile WiMAX receiver, we find it useful to shortly reference its SISO counterpart. The implementation and computational complexity is significantly reduced in the SISO receiver. However, the MIMO receiver features only the PUSC permutation scheme, whereas the SISO architecture is implementing both the PUSC and the Adaptive Modulation and Coding (AMC) permutation schemes. The AMC is required to develop closed MIMO schemes.

#### *3.3.1 RF front-end, analogue to digital conversion (ADC) and automatic gain control (AGC)*

The RF front-end at the receiver is applying low-noise amplification, down-conversion from RF to IF (i.e., in our case centred at $f_{IF} = 156.8$ MHz) and suppression of out-of-band unwanted signals such as noise and spurs. The A/D converters apply under-sampling featuring an ADC sampling rate of $f_s = 89.6$ MHz. Therefore, after the ADC, one of the aliases of the discrete signal will be located at 22.4 MHz, which is the sampling frequency at the baseband receiver as described by the WiMAX standard.

The AGC component is interfacing and adjusting the signal power level between the digital and the analogue domain. In more detail, a digitally-controlled analogue circuit, namely the Programmable Gain Amplifier (PGA), adjusts the power of the received IF signal, whereas a dedicated algorithm implemented in the digital domain decides the gain value to be applied in the PGA. The correct operation of the AGC is a decisive factor for the overall performance of a mobile receiver. Frame-based OFDM systems like WiMAX ones are prone to high Peak-to-Average Power Ratio (PAPR) a fact that imposes the use of a back-off safety margin.

The AGC algorithm is based on a signal peak-detector that operates in a per-frame basis and by this provides a

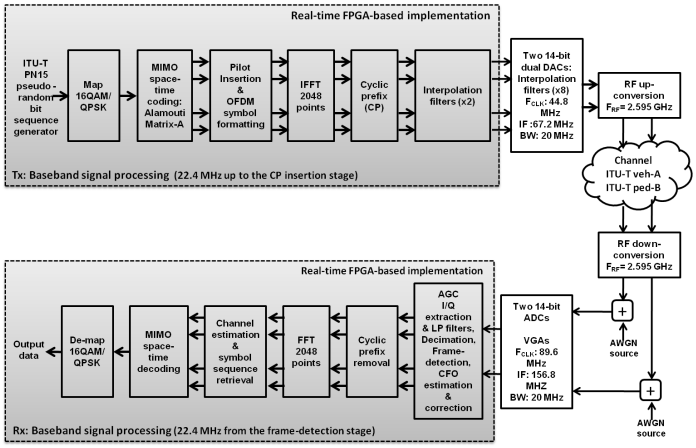



Figure 2: Overview of the MIMO transceiver architecture.

baseline trade-off between implementation complexity and efficiency. The inter-frame silence period is used to calculate and apply the new PGA gain (i.e., taking into account the peak value of the previous data frame). The PGA used in our receiver has 16 gain steps (i.e., 1.5 dB gain corrections). The gain-adjustment $\Delta G$, is calculated as follows:

$$\Delta G = 10 \cdot log_{10} \frac{\frac{v_{FS}^2}{v_{BM}^2}}{v_{PK}^2} \, dB = 10 \cdot log_{10} g \, dB, \quad (5)$$

where $v_{FS}$ is the digital full scale of the quantizer in the ADC, $v_{BM}$ accounts for the back-off safety margin, and $v_{PK}^2 = max \, |c_i[n]|^2$, with $c_i[n]$ representing the samples at the output of the ADC during the previous frame.

#### *3.3.2 Digital down converter (DDC)*

The DDC comprises the Direct Digital Synthesizer (DDS) component which translates any frequency band within the analogue bandwidth of the ADCs down to baseband, a complex Finite Impulse Response (FIR) low-pass filter responsible for eliminating the out-of-band components and an output decimator and formatter, which is keeping one out of every 4 samples, delivering the complex representation of the digitalized signal (i.e., I/Q, components). The DC level introduced by the baseband processing boards is transformed to a synchronization-altering sinusoid, which is eliminated in the digital filtering stage.

The output frequency of the DDS, $f_{DDS}$, is controlled by the phase increment, $\Delta\theta$, as follows:

$$f_{DDS} = \frac{f_s \cdot \Delta\theta}{2^{B_{\theta(n)}}} \, Hz, \quad (6)$$

where $f_s$ is the ADC sampling rate (89.6 MHz) and $B_{\theta(n)}$ is the resolution in bits of the internal accumulator used in the DDS (32 bits in our case). On power-up the $f_{DDS}$ is tuned at 22.4 MHz and then is constantly updated in real-time to compensate the effects of the CFO:

$$f_{DDS} = 22.4 + \Delta f \; MHz, \quad (7)$$

$\Delta f$ represents the CFO that is defined in terms of the separation between adjacent subcarriers:

$$\Delta f = \alpha \frac{22.4 \cdot 10^6}{2048}, \quad (8)$$

where $\alpha$ is the CFO normalized with respect to the intercarrier separation (i.e., in practice the CFO will not be higher

than one half of the intercarrier separation, $\alpha \in [-0.5, 0.5]$), 22.4 MHz is the sampling frequency and 2048 is the FFT size. Combining (6), (7) and (8) the phase increment is given by:

$$\Delta\theta = (22.4 + \alpha \cdot \frac{22.4}{2048}) \cdot \frac{2^{32}}{89.6} = 2^{30} + \alpha \cdot 2^{19} \quad (9)$$

*3.3.3 Synchronization, CFO estimation and correction*

Each frame in our system is composed by 46 OFDM data-symbols, each one of which contains 2560 samples. The symbol detection is required to properly locate the FFT window of the samples corresponding at each OFDM symbol. This is feasible with the inclusion of a CP at the beginning of each OFDM symbol. In this paper, we have considered channels with a maximum delay spread of 2510 ns, where only 455 out of the 512 samples in the CP can be used for the timing synchronization (i.e., the remaining 57 samples are discarded to avoid unreliable operation of the FFT window-locator). The synchronization implementation is based on a sliding window of 2048+455 samples, allowing the calculation of the cross-correlation of two groups of 455 samples (having a separation of 2048 samples). In order to reduce the implementation complexity of this operation, the following simplification was applied:

$$|r_s[n]|^2 = \frac{|dn[n]|^2}{ds0[n] \cdot ds1[n]}, \quad (10)$$

where:

$$dn[n+1] = \begin{cases} dn[n] & + \sum_{i=1}^{n_R} s_i^*[n+455] \cdot s_i[n+2048+455] \\ & \text{if n}\leq 455 \\ dn[n] & - \sum_{i=1}^{n_R} s_i^*[n] \cdot s_i[n+2048] \\ & + \sum_{i=1}^{n_R} s_i^*[n+455] \cdot s_i[n+2048+455] \\ & \text{if n}>455, \end{cases} \quad (11)$$

where $s_i[n]$ is the equivalent complex baseband signal at the output of the DDC, sampled at 22.4 MHz, at the $i$th receive antenna processing branch, $n_R$ denotes the number of receive antennas and with $dn[0] = 0$. It should be noted that $ds0[n]$, $ds1[n]$ are calculated in a similar manner. With this optimization only four samples need to be introduced to the already calculated correlation. A peak in $|r_s[n]|^2$, indicates the detection of the symbol and thus the sample where the CP starts, i.e., $pos_{cp} = arg\ max_n |r_s[n]|^2$. Additionally the phase of the correlation (i.e., the numerator of $|r_s[n]|^2$), at $pos_{cp}$ can be used to estimate the phase shift of the received signal in the presence of CFO. Using the notation given in (8), the phase shift between two signal samples delayed by 2048 positions is equal to $e^{j2\Pi\Delta ft}|_{t=2048\frac{1}{22.4\cdot 10^6}} = e^{j2\Pi\alpha}$. Therefore, the estimated CFO (constant $\alpha$) is defined as:

$$\alpha = \frac{1}{2\Pi} \measuredangle (r_s[pos_{cp}]) = \frac{1}{2\Pi} \measuredangle (dn[pos_{cp}]), \quad (12)$$

where $\alpha$ can be calculated using a Coordinate Rotation Digital Computer (CORDIC) algorithm. As already mentioned, the estimated CFO is used to fine tune the DDS.

Our implementation accounted for another parasitic sinusoid, which, from the one hand, is due to the presence of CFO drifts and on the other hand, due to the DC-level resulting from the digital mixing of the unwanted residual carrier with a digitally generated sinusoid at the DDS component. The presence of this sinusoid misguides the detection of the correlation peak (i.e., erroneous detection during the silence period, resulting in misplacement of the window of samples forwarded to the FFT). Failure to prevent an erroneous symbol detection may render the system unusable. Therefore, when a correlation value above a predefined threshold is detected, the shape of the correlation curve was used to determine whether the located peak is credible or not.

*3.3.4 Pilot extraction and channel estimation*

The channel estimation in our receiver is pilot-based. Let $S_i[k]$ be the $k$th subcarrier in the OFDM symbol received by the $i$th receive antenna after the FFT (i.e., $S_i[k] = FFT(s_i[n])$), with $k \in [0..n_U - 1]$, where $n_U$ represents the number of subcarriers used to transmit user data and pilot tones. The IEEE 802.16e standard defines the value, $p_v = \frac{4}{3}$, and location (i.e., frequency), $p_{k,j}$, of the pilot subcarriers for each transmit antenna $j$, i.e., $p_{k,j} \in [0..n_U - 1]$. In the PUSC permutation scheme that has been implemented, out of the 2048 subcarriers available at each OFDM symbol, 1440 are used for data transmission and 240 for pilot tones transmission, i.e., $n_U = 1679$ (the remaining subcarriers are utilized for the guard-bands and DC carrier). When $n_T = 2$ the PUSC permutation scheme distributes the pilot tones for each antenna in two consecutive OFDM symbols; this implies that the channel estimation will be applied in pairs of consecutive OFDM symbols (i.e., the estimated channel frequency response will be the same for both).

Each processing branch of the MIMO-enabled receiver has to estimate the corresponding channels from all transmit antennas. First, the channel frequency response at the pilot tones, $\tilde{H}_{i,j}[p_{k,j}]$, is estimated as follows:

$$\tilde{H}_{i,j}[p_{k,j}] = \frac{Si[p_{k,j}]}{\frac{4}{3}}, \quad (13)$$

where $Si[p_{k,j}]$ represents the $k$th pilot tone from the $j$th transmit antenna after the FFT in the $i$th receive antenna processing chain, with $j \in [1, 2]$ in our case. Thus, $\tilde{H}_{i,j}[p_{k,j}]$ is a discrete function calculating the channel frequency response at the pilot tones between the $i$th receive antenna and the $j$th transmit antenna. An interpolation of the pilot positions is then required to estimate the channel at the frequencies where data subcarriers were transmitted for each transmit-receive antenna pair. A second order polynomial interpolation has been selected, as it provides the best trade-off between accuracy and implementation complexity (accounting for the number of pilot tones in each OFDM symbol and the channel specifications). Therefore, the channel frequency response for the data subcarriers is calculated by:

$$\begin{aligned} \tilde{H}_{i,j}[k] =& \tilde{H}_{i,j}[p_{c_1,j}] + \frac{\tilde{H}_{i,j}[p_{c_2,j}] - \tilde{H}_{i,j}[p_{c_1,j}]}{p_{c_2,j} - p_{c_1,j}} \cdot (k - p_{c_1,j}) \\ & + \frac{\frac{\tilde{H}_{i,j}[p_{c_3,j}] - \tilde{H}_{i,j}[p_{c_2,j}]}{p_{c_3,j} - p_{c_2,j}] } - \frac{\tilde{H}_{i,j}[p_{c_2,j}] - \tilde{H}_{i,j}[p_{c_1,j}]}{p_{c_2,j} - p_{c_1,j}}}{p_{c_3,j} - p_{c_1,j}} \cdot \\ & \cdot (k - p_{c_1,j}) \cdot (k - p_{c_2,j}), \end{aligned} \quad (14)$$

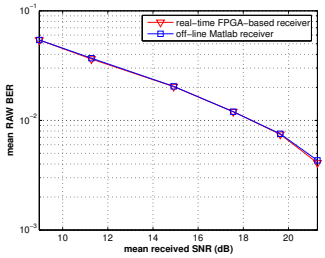
where $p_{c_r,j}$ represents the location of one of the three closest pilot tones, originated from transmit antenna $j$, to $S_i[k]$, respectively for each $r \in [1..3]$.

*3.3.5 Matrix A space-time decoding*

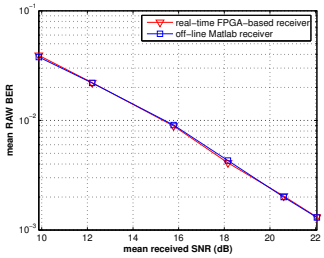
The Alamouti's STC (matrix A) transmission scheme encodes data symbols, $d_k$, as described previously. In order to decode and estimate the transmitted data symbols the following operations are applied:

$$\hat{d}_k[2O_n] = \frac{\sum_{i=1}^{n_R} \tilde{H}_{i,1}^*[k] \cdot S_i[k, 2O_n] + \tilde{H}_{i,2}[k] \cdot S_i^*[k, 2O_n + 1]}{\sum_{i=1}^{n_R} |\tilde{H}_{i,1}[k]|^2 + |\tilde{H}_{i,2}[k]|^2}, \quad (15)$$

where $O_n$ and $O_n+1$ represent the indexes of the two consecutive OFDM symbols. In the same clock cycle, $\hat{d}_{k+1}[2O_{n+1}]$ is calculated in a similar manner to (15). Note that it is

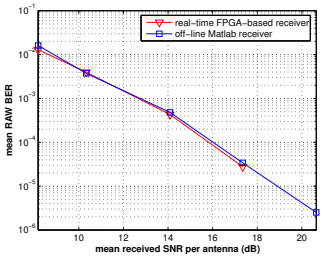



(a) ITU-T vehicular A-based

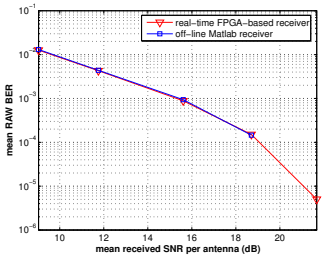



(b) ITU-T pedestrian B-based

Figure 3: SISO configuration: system performance under static channel conditions - BER-SNR curves.




(a) ITU-T vehicular A-based




(b) ITU-T pedestrian B-based

Figure 4: 2x2 MIMO configuration: system performance under static channel conditions - BER-SNR curves.

assumed that the gain applied by the AGC to the incoming sample-streams is equal for both receive antennas.

The following processing stages are related to the PUSC permutation, clustering, channelization and mapping of the data symbols, which enable the recovery of the originally transmitted bit sequence.

## 4. EXPERIMENTAL RESULTS

Dedicated logic has been inserted at specific points of the mobile WiMAX receiver, for both SISO and MIMO system-configurations, to allow the real-time data capturing and visualization, through vendor-specific tools and the Chipscope Pro software from Xilinx. The IF signal was captured at the outputs of the ADCs and used to calculate the received Signal-to-Noise Ratio (SNR). The same signal-capture was fed to the respective Matlab models of the receiver (both SISO and MIMO), allowing us to quantify the implementation losses. The output of the QPSK demodulator was also captured, enabling the calculation of the raw Bit Error Rate (BER) (i.e., no channel coding was used). This performance metric has been calculated in each case for a set of suitable SNR values covering 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 are covering several entire frames, allowing an averaged calculation of the main performance metrics.

For the tested scenarios, the real-time channel emulator was configured with static random channel models (i.e., no mobility is emulated), having the tap and delay spread specifications of the ITU Vehicular A and the ITU Pedestrian B. The performance of the SISO transceiver is shown in figure 3, while figure 4 presents the equivalent results for the 2x2 MIMO configuration. As it can be observed, the real-time implementation achieves nearly identical results with the Matlab model, demonstrating the robustness and high-precision of the fixed-point FPGA-based implementation. The absence of mobility in the experimental testing (i.e., variation of the channel frequency response) is preventing the assessment of the diversity gains introduced by the MIMO scheme; however, this is part of ongoing work.

## 5. CONCLUSION

This paper presented the design, implementation and validation of a custom FPGA-based PHY-layer of a high-performance real-time MIMO mobile WiMAX transceiver, for both the SISO and 2x2 MIMO configurations. The latter is featuring an Alamouti's STBC-based scheme. The validation of the implementation has been successfully conducted for static channel conditions using the GEDOMIS® testbed. The PHY-layer specifications featuring a 20 MHz bandwidth, have scaled up the implementation complexity, but can also be considered an added-value to the whole undertaking. The main contribution of the authors lies on the deployment and validation of complex real-time wireless communication systems, over a flexible multi-antenna testbed in close-to real-world conditions.

## REFERENCES

[1] A. Carro-Lagoa, P. Surez-Casal, A. Morales-Mndez, J. M. Camas-Albar and L. Castedo, "Prototipado SDR de la capa fsica OFDMA del estndar IEEE 802.16e," in *Proc. URSI 2009*, Cantabria, Spain, September 16-18.

[2] S. Shin, S. Lee, K. Wang, J. Choi and D. Chung, "Low-latency implementation of DFT-based transform domain channel estimation system in Mobile WiMAX system," in *Proc. ISCIT 2009*, Incheon, South Korea, September 28-30.

[3] M. A. Kadhim and W. Ismail, "Implementation of WIMAX IEEE 802.16e Baseband Transceiver on Multi-Core Software Defined Radio Platform," *International Journal of Computer Theory and Engineering*, vol. 2, pp. 820–825, Oct. 2010.

[4] G. Zaggoulos, M. Tran and A. Nix, "Mobile WiMAX system performance - simulated versus experimental results," in *Proc. PIMRC 2008*, Cannes, France, September 15-18.

[5] D. Iancu, H. Ye, E. Surducan, M. Senthilvelan, J. Glossner, V. Surducan, V. Kotlyar, A. Iancu, G. Nacer and J. Takala, "Software Implementation of WiMAX on the Sandbridge SandBlaster Platform," in *Proc. SAMOS 2006*, Samos, Greece, July 17-20.

[6] Q. Wang, D. Fan, Y. H. Lin, J. Chen and Z. Zhu Z., "Design of BS transceiver for IEEE 802.16E OFDMA mode," in *Proc. ICASSP 2008*, Las Vegas, USA, March 30-April 04.

[7] Y. J. Wu, J. M. Lin, H. Y. Yu and H. P. Ma, "A baseband testbed for uplink mobile MIMO WiMAX communications," in *Proc. ISCAS 2009*, Taipei, Taiwan, March 24-27.

[8] S. M. Alamouti, "A simple transmit diversity technique for wireless communications," *IEEE Journal on Selected Areas in Communications*, vol. 16, pp. 1451–1458, Oct. 1998.

[9] M. S. Khairy, M. M. Abdallah and S. E. D. Habib, "Efficient FPGA Implementation of MIMO Decoder for Mobile WiMAX System," in *Proc. ICC 2009*, Dresden, Germany, June 14-18.

[10] O. Font-Bach, N. Bartzoudis, A. Pascual-Iserte and D. López Bueno, "A Real-Time MIMO-OFDM Mobile WiMAX Receiver: Architecture, Design and FPGA Implementation," *Elsevier Journal of Computer Networks, Elsevier Journal of Computer Networks, doi:10.1016/j.comnet.2011.02.018 (in press)*, 2011.