MIMO Hardware Simulator Using Standard Channel Models and Measurement Data at 2.2 and 3.5 GHz

Bachir Habib, Gheorghe Zaharia, Hanna Farhat, Ghaïs El Zein

Jan. 2013
Journal of Communication and Computer, USA

MIMO Hardware Simulator Using Standard Channel Models and Measurement Data at 2.2 and 3.5 GHz

Bachir Habib, Gheorghe Zaharia, Hanna Farhat, Ghaïs El Zein
Institute of Electronics and Telecommunications of Rennes, IETR, UMR CNRS 6164, Rennes, 35708, France

Received: October 31, 2012 /

Abstract: A wireless communication system can be tested either in actual conditions or by using a hardware simulator reproducing actual conditions. With a hardware simulator it is possible to freely simulate a desired type of a radio channel. This paper presents architectures for the digital block of a hardware simulator of Multiple-Input Multiple-Output (MIMO) propagation channels. This simulator can be used for Long Term Evolution System (LTE) and Wireless Local Area Networks (WLAN) 802.11ac applications, in indoor and outdoor environments. The first architecture is appropriate for shipboard environments, while the second corresponds to outdoor-to-indoor environments and considers the wave propagation penetration within buildings. Measurements campaigns carried out at 2.2 and 3.5 GHz have been conducted to obtain the impulse responses of the channel using a MIMO channel sounder designed at IETR. The measurements are processed with an algorithm extracting the dominant paths. The architectures of the digital block are implemented on a Xilinx Virtex-IV Field Programmable Gate Array (FPGA). After the implementation of the impulse responses, the accuracy, the occupation on the FPGA and the latency of the architectures are analyzed.

Key words: Hardware simulator, FPGA, time-varying MIMO channels, channel sounder, 802.11ac

1. Introduction

Wireless communication systems offer high data bit rates using Multiple-Input Multiple-Output (MIMO) techniques. MIMO systems make use of antenna arrays simultaneously at both transmitter and receiver sites to improve the capacity and/or the system performance. The current communication standards indicate a clear trend in industry toward supporting MIMO functionality. In fact, several studies published recently present systems that reach a MIMO order of 8×8 and higher [1]. This is made possible by advances at all levels of the communication platform, as the monolithic integration of antennas [2] and the simulator platforms design [3]. However, the transmitted electromagnetic waves interact with the propagation environment. Thus, it is necessary to take into account the main propagation parameters during the design of the future communication systems.

The objectives of our work mainly concern the channel models and the digital block of the simulator, as shown in Fig. 1. The design of the RF blocks was completed in a previous project [4].

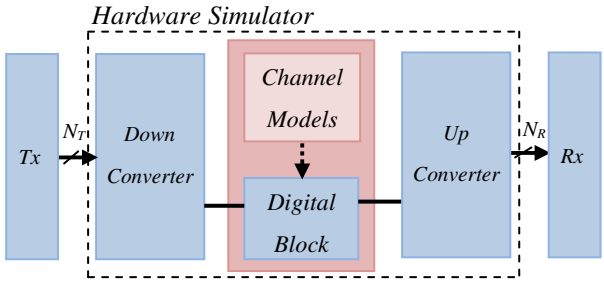


Fig. 1 Block diagram of MIMO hardware simulator.

The channel models can be obtained from standard channel models, as the TGn IEEE 802.11n [5] and the LTE models [6], or from real measurements

Corresponding author: Bachir Habib (1987), Ph.D student, research fields: wireless communications and MIMO systems with a focus on the propagation side, hardware simulator design for radio propagation environments, signal processing. Email: Bachir.habib@insa-rennes.fr

conducted with the MIMO channel sounder designed and realized at IETR [7] and shown in Fig. 2.

Fig. 2 MIMO channel sounder: receiver (left) and transmitter (right).

The channel sounder has a 100 MHz bandwidth and 200 MHz sampling frequency at a carrier frequency of 2.2 GHz or 3.5 GHz. Recently, the channel sounder was used during a measurement campaign on the ferryboat "Armorique" of Brittany Ferries and for outdoor-to-indoor environments.

The simulator is also configured with Long Term Evolution System (LTE) signals for LTE channel models and with Wireless Local Area Networks (WLAN) 802.11ac signals for TGn channel models.

In the MIMO context, little experimental results have been obtained regarding time-variations, partly due to limitations in channel sounding equipment [8]. However, theoretical models of time-varying channels can be obtained using Rayleigh fading [9].

Tests of radio communication systems, conducted under actual conditions are difficult, because tests taking place outdoors, for instance, are affected by random movements. In addition, a test conducted in one environment (city A) does not fully apply to a second corresponding environment (city B). Usually, under actual conditions, it is difficult to achieve the most difficult radio propagation conditions in order to determine the performance of a wireless communication system.

However, with hardware simulators, it is possible to very freely simulate desired types of radio channels. Moreover, a hardware simulator provides the necessary processing speed and real time performance, as well as the possibility to repeat the tests for any MIMO system. Thus, it can be used to compare the performance of various communication systems in the same desired conditions.

These simulators are standalone units that provide the fading output signal/signals for SISO/MIMO systems in the form of analog or digital samples. Some MIMO hardware simulators are proposed by industrial companies like Spirent (VR5) [10], Azimuth (ACE), Elektrobit (Propsim F8) [11], but they are quite expensive.

With continuing increase of the Field Programmable Gate Array (FPGA) capacity, entire baseband systems can be mapped onto faster FPGAs for more efficient prototyping, testing and verification. Larger and faster FPGAs permit the integration of many Single-Input Single-Output (SISO) channels and make it possible to design and test MIMO systems. As shown in [12], the FPGAs provide the greatest design flexibility and the visibility of resource utilization.

The MIMO hardware simulator is reconfigurable with sample frequencies not exceeding 200 MHz, which is the maximum value for FPGA Virtex-IV.

The channel sounder realized at IETR provides a sample frequency f_s of 200 MHz. For LTE signals, f_s = 50 MHz. For WLAN 802.11ac signals, f_s = 180 MHz. Thus, they are all compatible with the FPGA Virtex-IV. However, in order to exceed 200 MHz, more performing FPGA as Virtex-VII can be used [3].

Moreover, [13] presented a study relating f_s to the occupation on the FPGA. This study shows that doubling f_s allows us to simulate two channel systems (instead of one) in the same time period.

The simulator is able to accept input signals with wide power range, between -50 and 33 dBm, which implies a power control for the input signals.

At IETR, several architectures of the digital block of a hardware simulator have been studied, in both time and frequency domains [4, 14]. Moreover, [15] presents a new method based on determining the parameters of a channel simulator by fitting the space time-frequency cross-correlation matrix of the simulation model to the estimated matrix of a real-world channel. This solution shows that the obtained error can be important.

Typically, wireless channels are commonly simulated using finite impulse response (FIR) filters, as in [4, 14, 16]. The FIR filter output is a convolution between a channel impulse response and a fed signal in such a manner that this input signal, delayed by different delays, is weighted by the channel coefficients, i.e. tap coefficients, and these weighted components are summed up. The channel coefficients are periodically modified to reflect the behavior of an actual time-variant channel. Nowadays, different approaches have been widely used in filtering, such as distributed arithmetic (DA) and canonical signed digits (CSDs) [17].

Nevertheless, using FIR filter has a limitation. With a FPGA Virtex-IV, it is impossible to simulate a FIR filter with more than 192 multipliers (impulse response with more than 192 taps). In fact, to simulate 4 SISO channels (2×2 MIMO systems), 4 FIR filters must be used. Thus, a single FIR filter cannot use more than 192/4 = 48 multipliers. The number of taps can be increased on condition that each multiplier is used many times in the same sampling period. This solution is used if $f_s < 100$ MHz.

To simulate an impulse response with more than 192 taps, the Fast Fourier Transform (FFT) module can be used. With a FPGA Vitrex-IV, the size N of the FFT module can reach 65536 samples. Thus, several frequency domain architectures have been considered and tested [4, 18]. Their disadvantages are high latency and high occupation on FPGA.

The main contributions of the paper are:

- The considered frequency architecture in [4, 18] operates correctly for signals with a number of samples not exceeding the size of the FFT. Thus, in this paper, a new frequency architecture avoiding this limitation and operating in streaming mode is tested with TGn standard model B and with measured outdoor-to-indoor channel responses.
- The time domain architecture presented in [4, 14, 16] produces an occupation of 12 % of slices on the Vitrex-IV for a SISO channel. However, in this paper, we present a time domain architecture with an occupation of 3 % for one SISO channel and up to 49 % for a 4×4 MIMO systems.
- In general, the channel impulse responses can be presented in baseband with its complex values, or as real signals with limited bandwidth B between $f_c - B/2$ and $f_c + B/2$, where f_c is the carrier frequency. In this paper, to eliminate the complex multiplication and the f_c, the hardware simulation operates between Δ and $B + \Delta$, where Δ depends on the band-pass filters (RF and IF). The value Δ is introduced to prevent spectrum aliasing In addition, the use of a real impulse response allows the reduction by 50% of the size of the FIR filters and by 4 the number of multipliers. Thus, within the same FPGA, larger MIMO channels can be simulated.
- Studies of the digital block of hardware simulators were made using standard channel models for indoor [19], outdoor [20], vehicular [21] and for indoor for two successive MIMO profiles [22]. In this paper, which is an extension of [22], a study with measured impulse responses is presented. These impulse responses are obtained by measurement campaign using a channel sounder realized at our laboratory. Also, the implementation of 500 successive profiles for TGn

model B is made. Moreover, the average relative error and SNR for all the profiles are calculated.

- In this study, we related the number of bits used in the time domain architecture to the relative error of the output signals in order to identify the best trade-off between the occupation on the FPGA and the accuracy. Therefore, two improvement solutions are presented.

In this paper, we present a study of two alternative approaches as presented in Fig.3. The first approach performs in frequency domain, while the second approach is based on FIR filter.

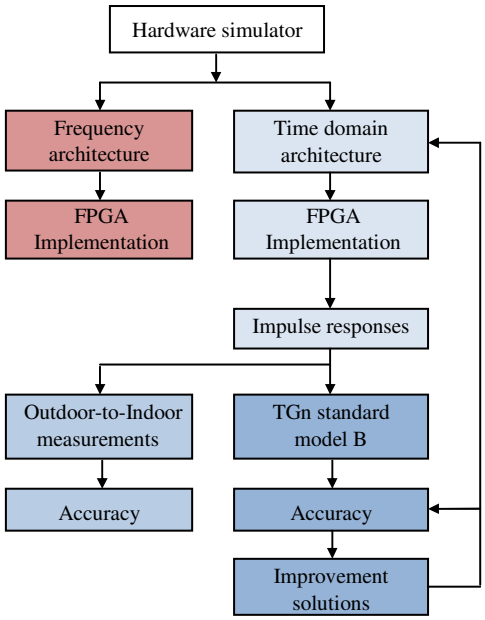


Fig. 3 Organization of the paper.

The paper is organized as follows. Section 2 presents the channel models. Section 3 describes the new architectures of the simulator in the frequency and time domain respectively. The prototyping platform used to implement these architectures and their occupation on the FPGA are also described. This Section shows that the time domain architecture is better in term of occupation on the FPGA. Therefore, it will be considered in the next Sections. Section 4 presents the accuracy of the output signals when standard TGn model B and outdoor-to-indoor measured impulse responses are used. Section 5 describes some improvement solutions to reduce the error, the latency and the occupation on the FPGA. Lastly, Section 6 gives concluding remarks and prospects.

2. Channel Models

A MIMO propagation channel is composed of several time variant correlated SISO channels. For MIMO 2×2 channel, the received signals $y_j(t,\tau)$ can be calculated using a time domain convolution :

$$y_j(t,\tau) = x_1(\tau) * h_{1j}(t,\tau) + x_2(\tau) * h_{2j}(t,\tau) \ , j = 1,2 \quad (1)$$

The associated spectrum is calculated by the Fourier transform (using FFT modules):

$$Y_j(t,f) = X_1(f).H_{1j}(t,f) + X_2(f).H_{2j}(t,f) \ , j = 1,2 \quad (2)$$

According to the considered environment, Table 1 summarizes some useful parameters.

Table 1 Simulator parameters.

	Type	Cell size	$W_{t\,eff}$ (µs)	N_F	W_{tF} (µs)	N_T	W_{tT} (µs)
LTE B=20 MHz	Rural	2-20 km	20	2048	40.96	1000	20
	Urban	0.4-2 km	3.7	512	10.24	185	3.7
	Indoor	20-400 m	0.7	256	2.56	35	0.7
802.11ac B=80 MHz	Office	40 m	0.35	128	0.64	70	0.35
	Indoor	50-150 m	0.71	512	2.56	142	0.71
	Outdoor	50-150 m	1.16	1024	5.12	232	1.16
Channel sounder B=100 MHz	Shipboard	9 m	20.48	512	2.56	200	1
	Outdoor to Indoor	100 m	20.48	512	2.56	200	1

W_{teff} represents the time window of the MIMO impulse responses. The number of samples computed for the frequency domain is given by:

$$N_F = W_{tF}.f_s \quad (3)$$

and for the time domain by:

$$N_T = W_{tT}.f_s \quad (4)$$

where W_{tF} is the closest value for W_{teff} which is imposed by the size $N_F = 2^n$ of the FFT modules.

Two channel models are considered to cover many types of environments: the TGn channel models (indoor) and the LTE channel models (outdoor). Moreover, using the channel sounder realized at IETR, measured impulse responses are obtained for specific environments: shipboard, outdoor-to-indoor...

In the next paragraph we introduce the method used to obtain a time variant channel.

2.1. TGn Channel Model

TGn channel models [5] have a set of 6 profiles, labeled A to F, which cover all the scenarios for WLAN applications. Each model has a number of clusters. Each cluster corresponds to specific tap delays, which overlaps each other in certain cases. 802.11ac signals are considered with a bandwidth of B = 80 MHz. The sampling frequency and period are f_s = 180 MHz and $T_s = 1/f_s$ respectively. The relative power of each tap of the impulse response for all TGn channel models are presented in [5] by taking the Line-Of-Sight (LOS) path as reference.

2.2. LTE Channel Model

LTE channel models are used for mobile wireless applications. A set of 3 channel models is used to simulate the multipath fading propagation conditions. A detailed description is presented in [6].

2.3. Measurement Data

The channel sounder uses a periodic pseudo binary sequence. It has 11.9 ns temporal resolution for 100 MHz sounding bandwidth. The carrier frequencies are 2.2 GHz and 3.5 GHz.

Two measurement campaigns were carried out: The first campaign concerns a shipboard environment, while the second one, which is described in this paper, considers outdoor-to-indoor environment. The measured MIMO impulse responses are used thereafter by the hardware simulator.

2.3.1. Shipboard Environment

For the shipboard measurement campaign at 2.2 GHz, a Uniform Linear antenna Array (ULA) and a Uniform Rectangular antenna Array (URA) were used for the transmitter (*Tx*) and the receiver (*Rx*) respectively, to characterize the double directional channel on a 120° beam width in the horizontal plan.

The position of *Tx* and *Rx* are located at floor 2 of the shipboard presented in Fig. 4.

Fig. 4 Ferryboat Armorique of Brittany Ferries.

2.3.2. Outdoor-to-Indoor Environment

Few MIMO outdoor-to-indoor measurements are reported in the literature [23], but not at 3.5 GHz.

The penetration of electromagnetic waves mainly occurs through openings like doors and windows. Thus, a receiver located inside a building receives signals coming from few main directions.

Two UCA (Uniform Circular Array) were developed at 3.5 GHz (Fig. 5) to characterize 360° azimuthal double directional channel at both link sides. *Tx* and *Rx* contains 4 and 16 active elements respectively. In this paper, only two successive active elements at the transmitter and two at the receiver are used to model a 2×2 MIMO channel.

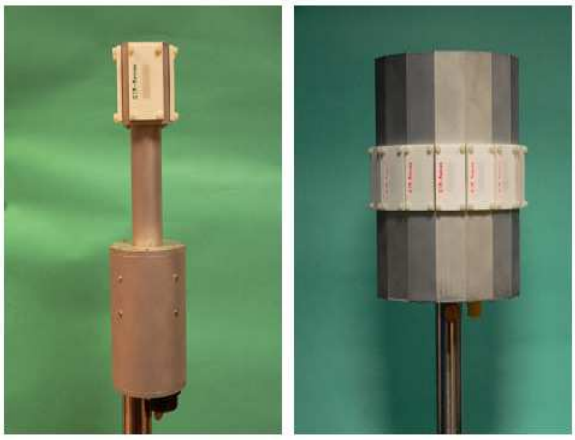


Fig. 5 UCA 4-element transmitter (left) and 16-element receiver (right).

(a)

(b)

Fig. 6 Measurement locations, a: Tx on a rooftop, b: Rx in classroom.

At the transmitter we integrated power amplifiers to increase the transmitted power. At the receiver we added Low Noise Amplifiers (LNA) behind the antennas to obtain more measurement dynamics. The collected channel data are stored on a laptop for post-processing. The transmitter was placed on the rooftop of a building and the receiver was located in multiple positions in different rooms (Fig. 6). The Tx-Rx distance is about 100 m. The transmitter height was set to 17.5 m above the ground (2 m above rooftop) for an orientation parallel to the building. The Rx antennas were fixed at 1.5 m and were oriented parallel to the building walls.

The channel sounder provides the complex envelope $h_{ce}(t)$ of the channel impulse response. The used real impulse responses in the band of $[\Delta, \Delta + B]$ are:

$$h(t) = h_p(t).\cos(2\pi f_0 t) - h_q(t).\sin(2\pi f_0 t) \quad (5)$$

where $h_p(t)$ and $h_q(t)$ are the real and imaginary parts of $h_{ce}(t)$ and:

$$f_0 = \frac{B}{2} + \Delta \quad (6)$$

The first results (without normalization) are obtained for a 2×2 MIMO channel. Fig. 7 presents the impulse responses given by the channel sounder on $2048T_s$.

However, with a FPGA Virtex-IV and with f_s=200 MHz, the number of multipliers used by a FIR filter is limited to 192. Thus, a high resolution method is proposed [24, 25] in order to estimate the propagation parameters of this channel and to obtain significant impulse responses with a limited number of taps and hence a limited number of multipliers. However, these methods are heavy computation load. Therefore, a new method is proposed which consist of detecting the taps considered as points of change for the sign of the slope of the curve.

Fig. 8 presents the impulse responses used by the simulator after discrimination, normalization and limitation between 0 and -20 dB of the measured impulse responses.

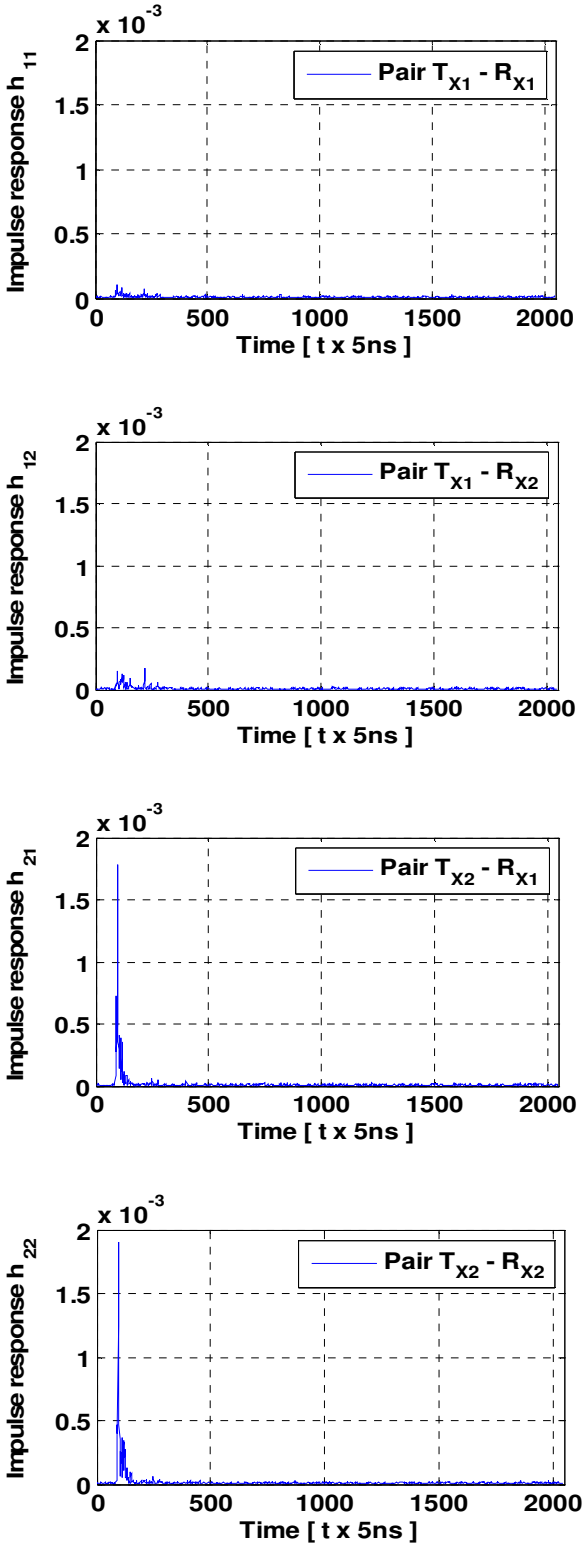


Fig. 7 Measured impulse responses.

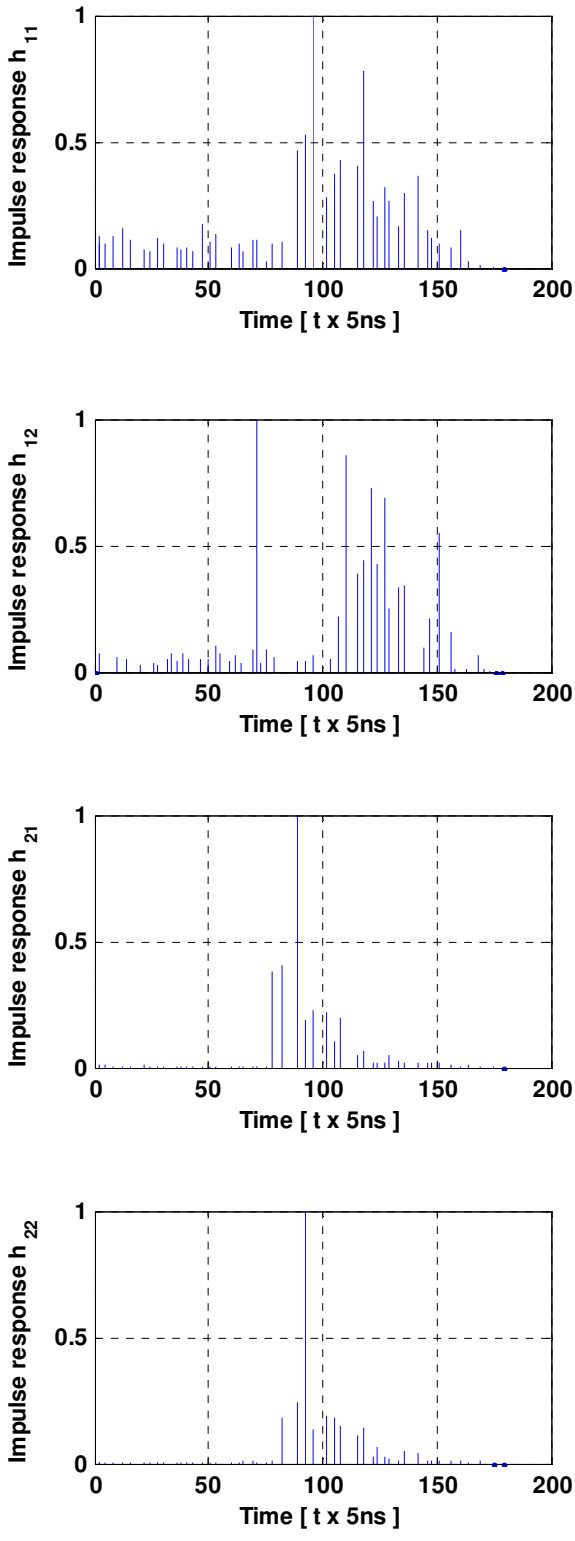


Fig. 8 Impulse responses used for the test.

Table 2 shows the number of taps and the time window W_t of the impulse response. W_t is equal to the number of the last non-null sample multiplied by T_s = 5 ns (W_t is the delay of the last non null sample).

Table 2 MIMO impulse responses.

	Number of taps	Last tap	W_t (ns)
h_{11}	49	180	900
h_{12}	46	175	875
h_{21}	47	177	885
h_{22}	48	173	865

2.4. Time-Varying Channels

For the outdoor-to-indoor measurements, we obtained invariant impulse responses due to the absence of any movement in the environment. In this section, we present the method used to obtain a model of a time variant channel, using the Rayleigh fading.

For outdoor-to-indoor measurements, at f_c = 3.5 GHz and an environmental speed v = 4 km/h, the Doppler frequency f_d is equal to:

$$f_d = \frac{f_c.v}{c} = 12.96\ Hz \tag{7}$$

where c is the celerity. We have chosen a refresh frequency f_{ref} = 28 Hz > $2.f_d$ to respect the Nyquist-Shannon sampling theorem.

For an indoor (TGn channel model B) environment, at f_c = 5 GHz and v = 4 km/h, f_d = 18.51 Hz. Thus, we have chosen a refresh frequency f_{ref} = 40 Hz.

We consider a 2×2 MIMO Rayleigh fading channel. The MIMO channel matrix H can be characterized by two parameters:

1) The relative power P_c of constant channel components which corresponds to the Line-Of-Sight (LOS).
2) The relative power P_s of the channel scattering components which corresponds to the Non-Line-Of-Sight (NLOS).

The ratio P_c/P_s is called Ricean K-factor.

Assuming that all the elements of the MIMO channel matrix H are Rice distributed, it can be expressed for each tap by:

$$H = \sqrt{P_c}\ .\ H_F + \sqrt{P_s}\ .\ H_v \tag{8}$$

where H_F and H_V are the constant and the scattered channel matrices respectively.

The total relative received power $P = P_c + P_s$. Therefore:

$$P_c = P\ .\frac{K}{K+1} \tag{9}$$

$$P_s = P.\frac{1}{K+1} \tag{10}$$

If we combine (9) and (10) in (8) we obtain:

$$H = \sqrt{P}.\left(\sqrt{\frac{K}{K+1}}H_F + \sqrt{\frac{1}{K+1}}H_V\right) \tag{11}$$

To obtain a Rayleigh fading channel, K is equal to zero, so H can be written as:

$$H = \sqrt{P}.H_V \tag{12}$$

P is derived from Fig. 8 for each tap for outdoor-to-indoor environment or from [5] for TGn channel model B environment. For 2 transmit and 2 receive antennas:

$$H = \sqrt{P}.\begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix} \tag{13}$$

where X_{ij} (i-th receiving and j-th transmitting antenna) are correlated zero-mean, unit variance, complex Gaussian random variables as coefficients of the variable NLOS (Rayleigh) matrix H_V.

To obtain correlated X_{ij} elements, a product-based model is used [26]. This model assumes that the correlation coefficients are independently derived at each end of the link:

$$X = (R_r)^{1/2}.H_w.\left((R_t)^{1/2}\right)^T \tag{14}$$

H_w is a matrix of independent zero means, unit variance, complex Gaussian random variables. R_r and R_t are the receive and transmit correlation matrices. They can be written by:

$$R_t = \begin{bmatrix} 1 & \alpha \\ \alpha^* & 1 \end{bmatrix}, \quad R_r = \begin{bmatrix} 1 & \beta \\ \beta^* & 1 \end{bmatrix} \tag{15}$$

where α is the correlation between channels at two receives antennas, but originating from the same transmit antenna (SIMO). In other words, it is the correlation between the received power of channels that have the same Angle of Departure (AoD) . β the correlation coefficient between channels at two transmit antennas that have the same receive antenna (MISO).

The use of this model has two conditions:

1) The correlations between channels at two receive (resp. transmit) antennas are independent from the *Rx* (resp. *Tx*) antenna.
2) If s_1, s_2 are the cross-correlation between antennas of the same side of the link, then :
   - $s_1 = \alpha + \beta$.
   - $s_2 = \alpha^* + \beta$.

For the uniform linear array, the complex correlation coefficients α and β are expressed by ρ:

$$\rho = R_{xx}(D) + j.R_{xy}(D) \tag{16}$$

where $D = 2\pi d/\lambda$, $d = 0.5\lambda$ is the distance between two successive antennas, λ is the wavelength and R_{xx} and R_{xy} are the real and imaginary parts of the cross-correlation function of the considered correlated angles:

$$R_{xx}(D) = \int_{-\pi}^{\pi} \cos\left(D.\sin(\varphi)\right).PAS(\varphi).d\varphi \tag{17}$$

$$R_{xy}(D) = \int_{-\pi}^{\pi} \sin\left(D.\sin(\varphi)\right).PAS(\varphi).d\varphi \tag{18}$$

The PAS (Power Angular Spectrum) closely matchs the Laplacian distribution [27, 28]:

$$PAS(\theta) = \frac{1}{\sqrt{2}\sigma} e^{-|\sqrt{2}\theta/\sigma|} \tag{19}$$

where σ is the standard deviation of the PAS (which corresponds to the numerical value of AS).

To calculate $R^{1/2}$ we must calculate the eigenvalues (L_1, L_2) and the eigenvectors Q of R:

$$R = Q^{-1}.\begin{bmatrix} L_1 & 0 \\ 0 & L_2 \end{bmatrix}.Q \tag{20}$$

Then,

$$R^{1/2} = Q^{-1}.\begin{bmatrix} \sqrt{L_1} & 0 \\ 0 & \sqrt{L_2} \end{bmatrix}.Q \tag{21}$$

3. Digital Block Design of the Hardware Simulator

In this section, improved frequency and time domains architectures are presented and implemented on a FPGA Virtex-IV.

3.1. New Frequency Domain Architecture

The new frequency architecture presented in Fig. 9 has been verified with a Gaussian impulse response [29]. It operates correctly for signals with a number of samples exceeding N_F, where $N_F = 2^n$ is the size of the FFT module.

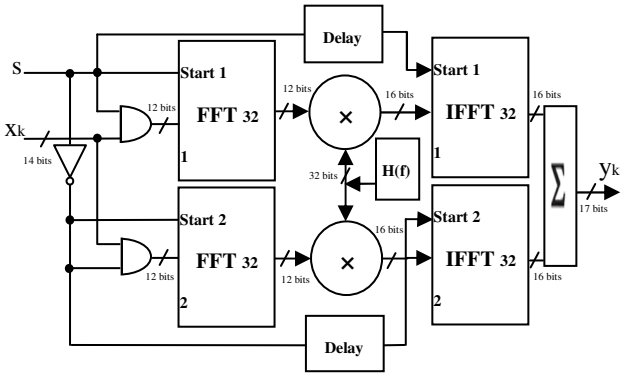


Fig. 9 Frequency architecture for a SISO channel for TGn channel model B.

For TGn model B, the maximum number of taps corresponds to 14 samples. Thus, N_F = 16 samples. However, it is mandatory to extend each partial input of N_F samples with a "tail" of N null samples to avoid a wrong result [29]. Therefore, the FFT/IFFT modules operate with 32 samples as presented in Fig. 9.

For outdoor-to-indoor channel, the largest excess delay is $180T_s$ (Fig. 8). Thus, N_F = 256. Therefore, the FFT/IFFT modules operate with 512 samples.

H is the representation of h in the frequency domain. It can be calculated by:

$$H = T_s . h_q . W_q \tag{22}$$

where h_q is h quantified on 16 bits and W_q is computed by:

$$W_q = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & (w)_q & (w^2)_q & \dots & (w^{N-1})_q \\ 1 & (w^2)_q & & & \vdots \\ \vdots & \vdots & & & \vdots \\ 1 & (w^{N-1})_q & \dots & \dots & \left(w^{(N-1)^2}\right)_q \end{bmatrix} \tag{23}$$

where

$$w^l = e^{-j.2.\pi.l.f_s.T_s} \tag{24}$$

and each w^l is quantified on 12 bits (which is the best trade-off between the occupation on FPGA of the FFT block and its accuracy).

The truncation block is located at the output of the digital adder. It is necessary to reduce the number of bits after the sum of the signals computed by the IFFT blocks to 14 bits. Thus, these samples can be accepted by the digital-to-analog converter (DAC), while maintaining the highest accuracy.

The immediate solution is to keep the most significant 14 bits. It is a "brutal" truncation. This truncation decreases the real value of the quantified output sample. 17 - 14 = 3 bits will be eliminated. Thus, instead of an output sample y, we obtain $\lfloor y/2^m \rfloor$, where $\lfloor u \rfloor$ is the biggest integer number smaller or equal to u.

However, for low voltages of the output of the digital adder, the brutal truncation generates zeros to the input of the DAC.

Therefore, a better solution is the sliding window truncation presented in Fig. 10 which uses the 14 most effective significant bits. This solution modifies the output sample values. Therefore, the use of a reconfigurable amplifier after the Digital-Analog convertor must be used to restore the correct output value.

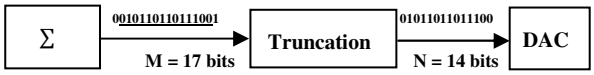


Fig. 10 Sliding window truncation from 17 to 14 bits.

3.2. New Time Domain Architecture

4 SISO channels are implemented. For each channel the FIR width and the number of used multipliers are determined by the number of taps of each channel.

Fig. 11 presents a FIR 14 (for TGn channel model B, the largest excess delay corresponds to 14 samples) with 9 multipliers (9 taps for each SISO impulse response). Thus, N_T = 14 samples.

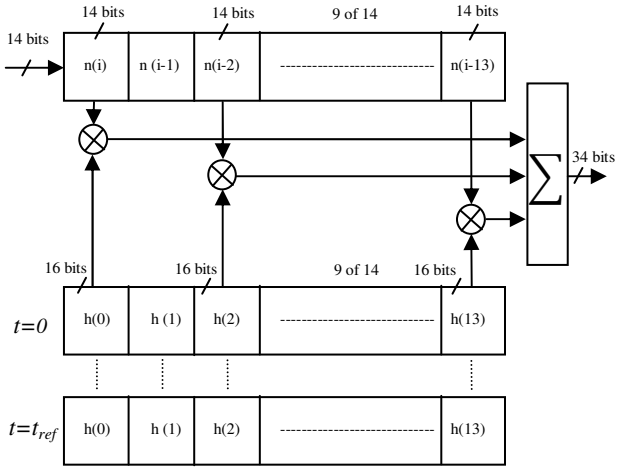


Fig. 11 FIR 14 filter with 9 multipliers for a SISO channel for TGn channel model B.

We have developed our own FIR filter instead of using Xilinx MAC FIR filter to make it possible to reload the FIR filter coefficients.

The general formula for FIR 14 filter with 9 multipliers is:

$$y_q(i) = \sum_{k=1}^{9} h_q(i_k).x_q(i - i_k), \qquad i \epsilon N \qquad (25)$$

In this relation, the index q suggests the use of quantified samples and $h_q(i_k)$ is the attenuation of the k^{th} path with the delay $i_k T_s$.

3.3. Implementation of each Architecture

Fig. 12 shows the Xtreme DSP Virtex-IV board from Xilinx [3] used for the implementation of each architecture. This prototyping board is described in [29].

The simulations and synthesis are made with Xilinx ISE [3] and ModelSim software [30].

The XtremeDSP features dual-channel high performance ADCs (AD6645) and DACs (AD9772A) with 14-bit resolution, a user programmable Virtex-IV FPGA, programmable clocks, support for external clock, host interfacing PCI, two banks of ZBT-SRAM, and JTAG interfaces.

Fig. 12 XtremeDSP Development board Kit-IV for Virtex-IV.

This development kit is built with a module containing the Virtex-IV SX35 component, selected to correspond to the complexity constraints. It contains a number of arithmetic blocks (DSP blocks) which makes it possible to implement many functions occupying most of the component. This device enables us to implement different time domain or frequency domain architectures and thus to reprogram the component according to the selected (indoor or outdoor) environment.

Frequency domain or time domain 2×2 MIMO architectures are implemented in the FPGA Virtex-IV. To test a higher order MIMO array, the use of more performing FPGA as Virtex-VII [3] is mandatory.

3.3.1. Implementation of the Frequency Domain Architecture

As a development board has 2 ADC and 2 DAC, it can be connected to only 2 down-conversion RF units and 2 up conversion RF units. Therefore, 4 frequency architectures are needed to simulate a one-way 2 x 2 MIMO radio channel.

The V4-SX35 utilization summary is given in Table 3 for 2×2 MIMO frequency architecture for TGn channel model B with their additional circuits used to dynamically reload the channel coefficients.

Table 3 FPGA occupation for 2×2 MIMO frequency architecture for TGn channel model B.

Number of slices	9,825 out of 15,360	64 %
Number of blocs RAM	36 out of 192	19 %
Number of multipliers	72 out of 192	38 %

For outdoor-to-indoor measurements, the V4-SX35 development board utilization summary shows that the 2×2 MIMO frequency architecture using 512 FFT/IFFT modules occupies more than 15,360 slides on the FPGA. Thus, more than 100 % of the slices are needed. It is impossible to simulate a 2×2 MIMO frequency architecture using 512 FFT/IFFT modules.

3.3.2. Implementation of the Time Domain Architecture

Table 4 shows the FPGA utilization of 2×2 MIMO time domain architecture using four FIR filters for

TGn channel model B with their additional circuits used to dynamically reload the channel coefficients.

Table 4 FPGA occupation for 2×2 MIMO time domain architecture for TGn channel model B.

Number of slices	1,946 out of 15,360	13 %
Number of blocs RAM	36 out of 192	19 %
Number of multipliers	36 out of 192	19 %

For outdoor-to-indoor measurements, the four FIR filter used: FIR 180 with 49 multipliers (for h_{11}), FIR 175 with 46 multipliers (for h_{12}), FIR 177 with 47 multipliers (for h_{21}) and FIR 173 with 48 multipliers (for h_{22}). Table 5 shows the utilization for 2×2 MIMO time architecture for outdoor-to-indoor channel.

Table 5 FPGA occupation for 2×2 MIMO time domain architecture for outdoor-to-indoor channel.

Number of slices	3,041 out of 15,360	20 %
Number of blocs RAM	190 out of 192	99 %
Number of multipliers	190 out of 192	99 %

The time domain architecture is better in terms of occupation on the FPGA. Moreover, in [19, 20] we showed that the time domain architecture has two other advantages: a higher SNR and a lower latency.

Thus, in this work, the time domain architecture is retained for the tests.

4. Implementation of the Impulse Responses in the Simulator

4.1. Description

The channel impulse responses are stored on the hard disk of the computer and read via the PCI bus and then stored in the FPGA dual-port RAM. Fig. 13 shows the connection between the computer and the FPGA board to reload the coefficients. 500 successive profiles are considered for the test of a 2×2 MIMO time-varying channel. We simulate an outdoor-to-indoor environment and a residential environment (TGn model B) where v is between 0 and 4 km/h. Therefore, in our test we choose to analyze the two extreme cases. However, to obtain non-null Doppler frequency, the two considered speeds are: 0.5 km/h and 4 km/h.

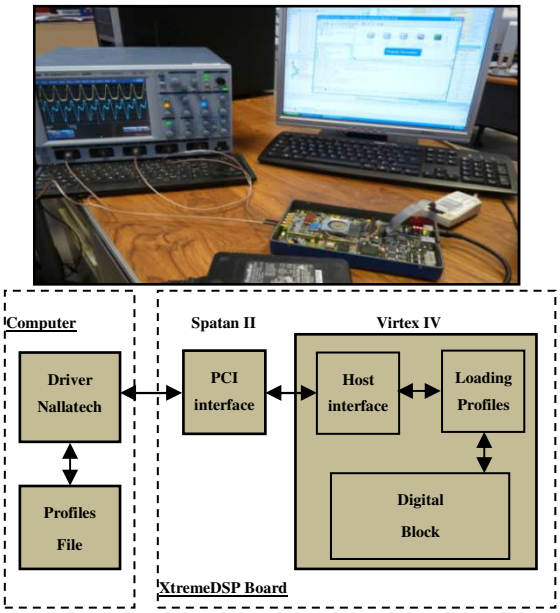


Fig. 13 Connection between the computer and the XtremeDSP board.

4.1.1. Using outdoor-to-indoor measurements

The channel sounder uses f_c = 3.5 GHz. For v = 0.5 km/h, f_{ref} = 3.5 Hz (calculated in detail in Section 2.4.) and the refreshing period T_{ref} = 0.285 s during which we must refresh all of the four SISO profiles, i.e. According to Table 2, 49 + 46 + 47 + 48 = 190 words of 16 bits = 380 bytes to transmit for a profile, which is: 380/0.285 = 1.333 kBps.

For v = 4 km/h, f_{ref} = 28 Hz and T_{ref} = 35.7 ms. Therefore, the amount of data transmitted for a MIMO profile is: 380/0.0357 = 10.644 kBps.

4.1.2. Using TGn channel model B

802.11ac signals uses a frequency f_c = 5 GHz. For v = 0.5 km/h, f_{ref} = 5 Hz and t_{ref} = 0.2 s during which we must refresh all of the four SISO profiles, i.e. 9 × 4 = 36 words of 16 bits = 72 bytes to transmit for a MIMO profile, which is: 72/0.2 = 360 Bps.

For v = 0.5 km/h, f_{ref} = 40 Hz and t_{ref} = 25 ms. Therefore, the amount of data transmitted for a MIMO profile is: 72/0.025 = 2.88 kBps.

The 500 MIMO profiles are stored in a text file on the hard disk of a computer. This file is then read to load the memory block which will supply RAM blocks on the simulator (one block for each tap of the impulse response). Each block RAM has a memory of 64 kB. Thus, up to 32000 MIMO profiles can be supplied in the RAM blocks.

Reading the file can be done either from USB or PCI interfaces, both available on the used prototyping board. The PCI bus is chosen to load the profiles. It has a speed of 30 MB/s. In addition, this is a bus of 32 bits. Thus, on each clock pulse two samples of the impulse response are transmitted.

The Nallatech driver presented in Fig. 13 provides an IP sent directly to the "Host Interface" that reads it from the PCI bus and stores these data in a FIFO memory. The module called "Loading profiles" reads and distributes the impulse responses in "RAM" blocks. This module called "BOX RAM" is the block "Memory" of the architecture.

While a MIMO profile is used, the following MIMO profile is loaded and will be used after the refresh period.

4.2. Accuracy of the Architecture

In order to determine the accuracy of the digital block, a comparison is made between the theoretical and the Xilinx output signals.

An input Gaussian signal $x(t)$ is considered for the two inputs of the 2×2 MIMO simulator. The use of a Gaussian signal is preferred because it has a limited duration in both time and frequency domains:

$$x(t)\begin{cases} x_{m1}e^{-\frac{(t-m_{x1})^2}{2\sigma_1^2}} & 0 \le t \le 3W_t/4 \\ -x_{m2}e^{-\frac{(t-m_{x2})^2}{2\sigma_2^2}} & 3W_t/4 \le t \le 3W_t/2 \end{cases} \quad (26)$$

where $n = 400$ (chosen greater than the length of the impulse response to test the architecture in streaming mode), $W_t = n.T_s$, $m_{x1} = 3.W_t/8$, $m_{x2} = 6.W_t/5$ and $\sigma_1 = \sigma_2 = m_{x1}/6$ (small enough to show the effect of each path of the impulse responses on the output signal). The input signal x is presented in Fig. 14.

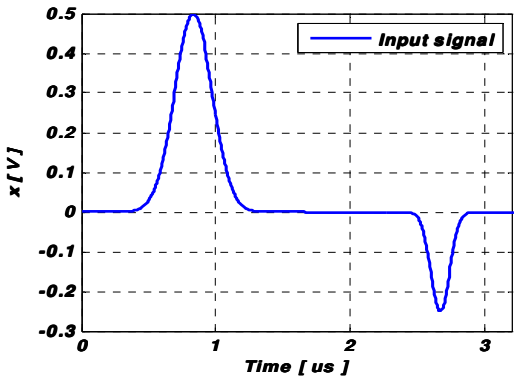


Fig. 14 Input signal.

The A/D and D/A converters of the development board have a full scale $[-V_m, V_m]$, with $V_m = 1$ V. For the simulations we consider $x_{m1} = V_m/2$ and $x_{m2} = V_m/4$. The theoretic output signals for TGn channel model B are calculated by:

$$y_1(t) = \sum_{k=1}^{9} h_{11}(i_k).x(t - i_kT_s) + \sum_{k=1}^{9} h_{21}(j_k).x(t - j_kT_s) \quad (27)$$

$$y_2(t) = \sum_{k=1}^{9} h_{12}(f_k).x(t - f_kT_s) + \sum_{k=1}^{9} h_{22}(l_k).x(t - l_kT_s) \quad (28)$$

The relative error is given for each output sample by:

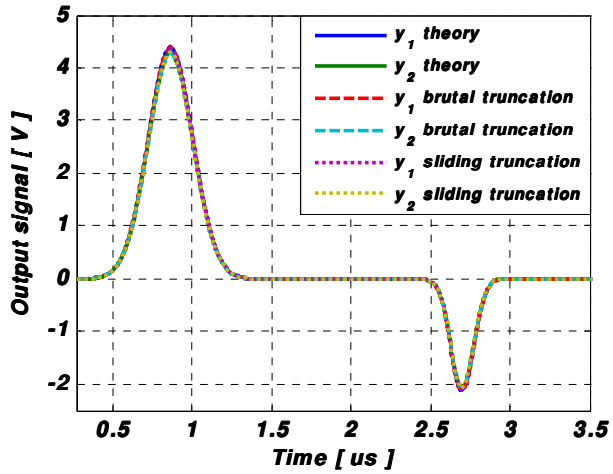
$$\varepsilon(i) = \frac{Y_{xilinx}(i) - Y_{theory}(i)}{Y_{theory}(i)}.100\ [\%] \quad (29)$$

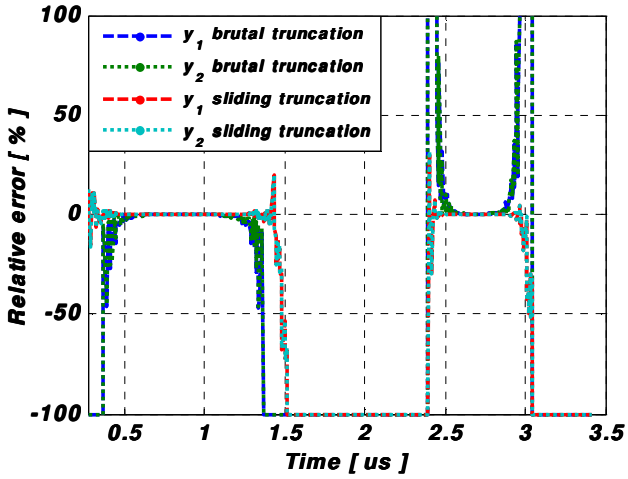
where Y_{Xilinx} and Y_{theory} are vectors containing the samples of corresponding signals. The Signal-to-Noise Ratio (SNR) is:

$$SNR(i) = 20.log_{10}\left|\frac{Y_{theory}(i)}{Y_{xilinx}(i) - Y_{theory}(i)}\right|\ [dB] \quad (30)$$

where $i = \overline{1,\ 3N + \iota_{Final}}$ and ι_{Final} is the maximum number of the last tap between h_{11} and h_{21}, then $max\{49, 46\} = 49$ for y_1, or between h_{12} and h_{22}, then $max\{47, 48\} = 48$ for y_2.

The output signals, the relative errors and the SNRs are presented in Fig. 15 for TGn model B, and in Fig. 16 for outdoor-to-indoor channel.





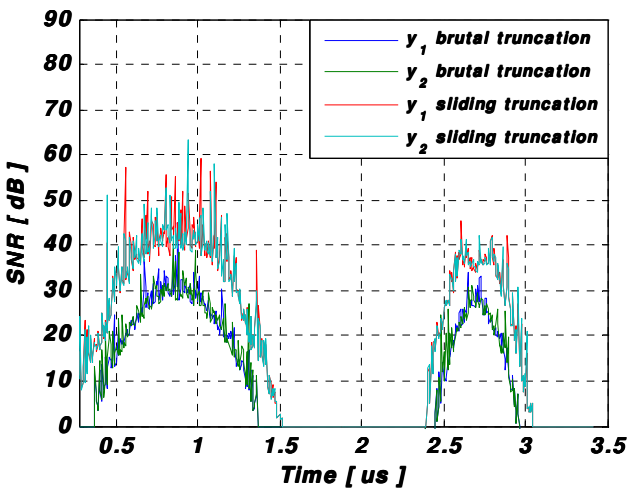
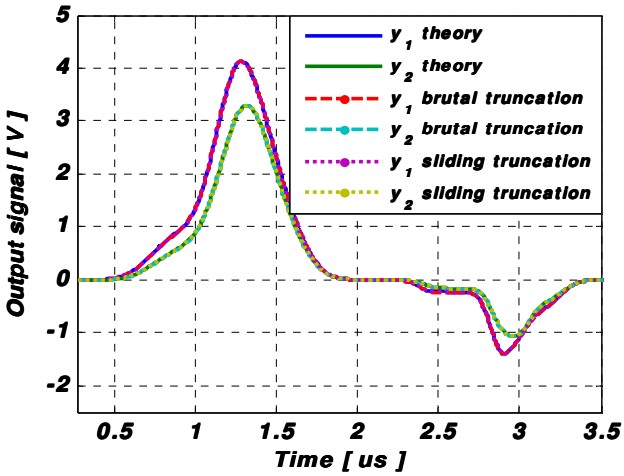
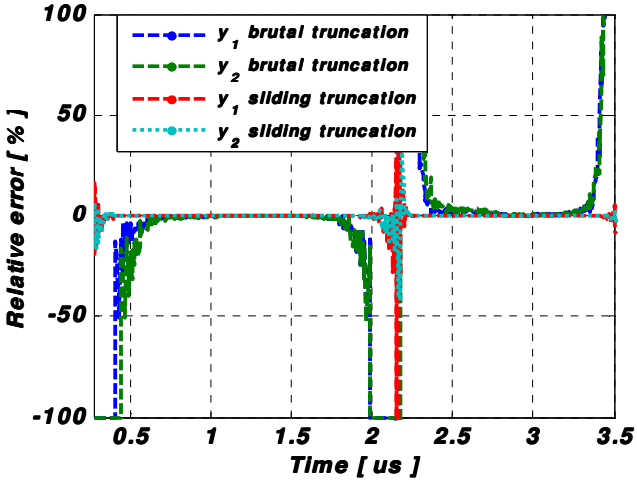


Fig. 15 The Xilinx output signals, the relative error and the SNR using TGn channel model B (residential).





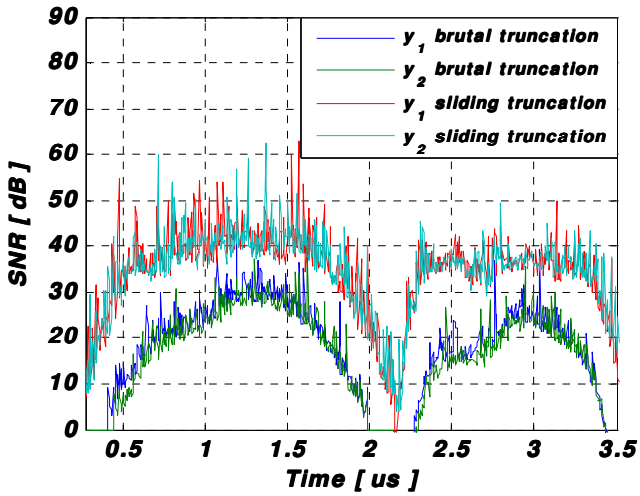


Fig. 16 The Xilinx output signals, the relative error and the SNR using outdoor-to-indoor measured data.

For TGn channel model B, the effect of the channel on the input signals is negligible. In fact, the length of the impulse responses is 14 (very low if we compare it to the length of $6\sigma_1$ or $6\sigma_2$). However, for outdoor-to-indoor measurement, the maximum length of the impulse responses is 180 which will affect the input signals.

After the D/A convertor, the signal is limited to $[-V_m, V_m]$ with $V_m = 1$. If $y_{max} > 1$ V as shown in Fig. 15 and Fig. 16, a reconfigurable analog amplifier placed after the DAC must multiply the signal with 2^{k_0}, where k_0 is the smallest integer verifying $y_{max} < 2^{k_0}$.

The relative error is high only for small values of the output signal.

The global values of the relative error and of the SNR computed for the output signal before and after the final truncations are necessary to evaluate the accuracy of the architecture. The global relative error is computed by:

$$\varepsilon = \frac{\|E\|}{\|Y_{theory}\|} \times 100\ [\%] \qquad (31)$$

The global SNR is computed by:

$$SNR_g = 20 \times log_{10} \frac{\|Y_{theory}\|}{\|E\|} [dB] \qquad (32)$$

where $E = Y_{Xilinx} - Y_{theory}$ is the error vector.

For a given vector $X = [x_1, x_2, \ldots, x_L]$, its Euclidean norm $\| x \|$ is:

$$\|x\| = \sqrt{\frac{1}{L}\sum_{k=1}^{L} x_k^2} \qquad (33)$$

Table 6 shows the global values of the relative error and the SNR between the Xilinx output signal and the theoretical output signal using 2×2 MIMO time domain architecture for TGn channel model B. The results are given without truncation, with sliding window and with brutal truncations.

Table 6 The global relative error and the global SNR using TGn channel model B.

Output	Error (%)	SNR (dB)
without truncation		
y_1	0.0061	84.29
y_2	0.0058	84.75
with sliding window truncation		
y_1	0.0112	78.36
y_2	0.0117	78.62
with brutal truncation		
y_1	0.3079	50.22
y_2	0.3881	48.21

Table 7 shows the global values of the relative error and the SNR using 2×2 MIMO time domain architecture for the outdoor-to-indoor environment.

Table 7 The global relative error and the global SNR using outdoor-to-indoor environment.

Output	Error (%)	SNR (dB)
without truncation		
y_1	0.0104	79.67
y_2	0.0105	79.59
with sliding window truncation		
y_1	0.0142	76.93
y_2	0.0142	76.93
with brutal truncation		
y_1	0.2547	51.87
y_2	0.2576	51.77

4.3. Global Error Variation with Time-Varying Profiles

To test the simulator with time-varying 2×2 MIMO channels, 500 successive profiles are considered.

4.3.1. Using outdoor-to-indoor measurements

For v = 0.5 km/h, f_{ref} = 3.5 Hz. Therefore, the time to simulate the 500 profiles is 143 s. Fig. 17 gives the time variation of the Average of the Global SNR of y_1 and the Global SNR of y_2 (AG SNR) for the 500 successive profiles.

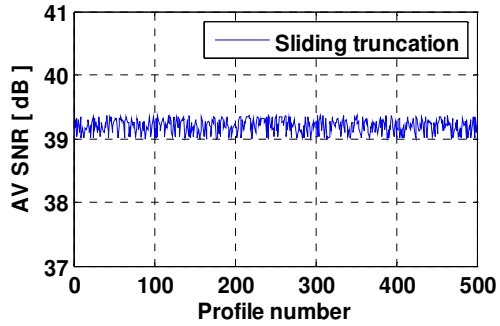


Fig. 17 AG SNR for v= 0.5 km/h for outdoor-to-indoor architecture.

For v = 4 km/h, f_{ref} = 28 Hz. Therefore, the time to simulate the 500 profiles is 18 s. Fig. 18 gives the time variation of the AG SNR for the 500 successive profiles.

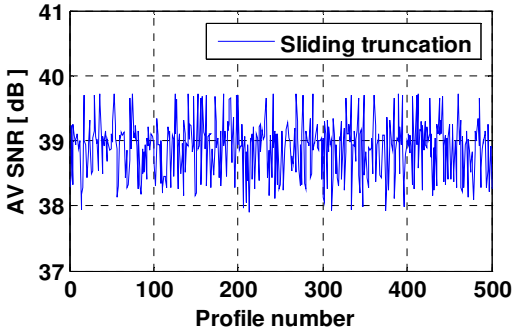


Fig. 18 AG SNR for v= 4 km/h for outdoor-to-indoor architecture.

4.3.2. Using TGn channel model B

For v = 0.5 km/h, f_{ref} = 5 Hz. Therefore, the time to simulate the 500 profiles is 100 s. Fig. 19 gives the time variation of the Average Global SNR (AG SNR) for the 500 successive profiles.

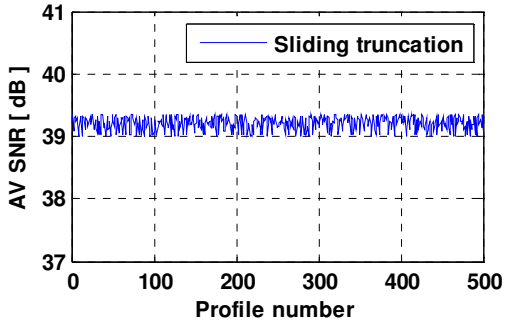


Fig. 19 AG SNR for v= 0.5 km/h for TGn model B architecture.

For v = 4 km/h, f_{ref} = 40 Hz. Therefore, the time to simulate the 500 profiles is 13 s. Fig. 20 gives the time variation of the Average Global SNR (AG SNR) for the 500 successive profiles.

For v = 0.5 km/h, the variation of SNR is 0.35 dB. For v = 4 km/h, the variation of SNR is 1.91 dB. Therefore, after several variation of v between 0 and 9 km/h (the maximum for an indoor environment), we conclude that the rate of variation of the SNR is related proportionally to the speed environment.

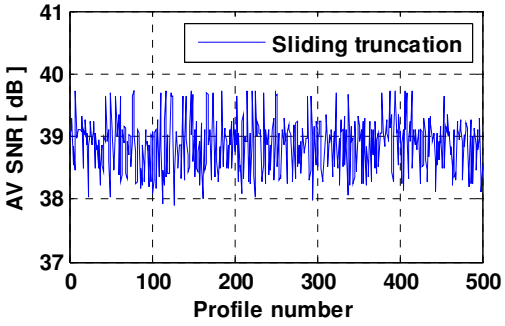


Fig. 20 AG SNR for v= 4 km/h for TGn model B architecture.

5. Improvement Solutions

The goal is to improve: the precision, the FPGA occupation and the latency.

Using an Auto Scale Factor (*ASF*) decreases significantly the error. Also, decreasing the number of bits of the implemented impulse responses presented

by h in Fig. 11 will decrease the occupation of slices in the FPGA and the latency.

In this section, the improvement solutions are tested using the digital block architecture defined for MIMO 2×2 TGn channel model B.

5.1. Auto Scale Factor (*ASF*)

After analyzing the SNR in Fig. 15, we conclude that it is high only for high values of the input signals. Therefore, to decrease the error, a solution is proposed. The input and output signals are limited to $[-V_m,V_m]$ with V_m = 1 V. The solution consists on multiplying each sample of the input signal with a corresponding 2^k where k is an integer verifying: $0.5 < 2^k.x < 1$. In fact, if $x > 0.5$ V, the SNR is high as presented in Fig. 14. However, we cannot predict x and multiply each sample by *ASF* at a high sample frequency.

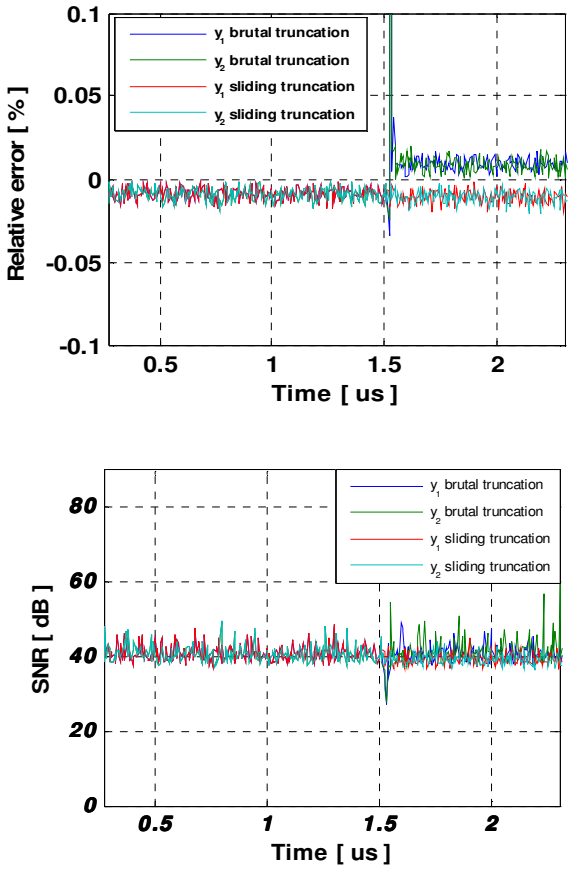


Fig. 21 Relative error and SNR using *ASF*

Therefore we will use the *ASF* on the MIMO impulse responses. If $h_{max} = max\ (|h|) < 0.5$ it will be multiplied by 2^{k_h} where k_h is the unique integer verifying $0.5 < 2^{k_h}.h_{max} < 1$.

In the case of a brutal truncation, $ASF=2^k$. However, for sliding truncation, if the output signals are presented on more than 14 bits, the sliding factor 2^{k_y} has to be considered to amplify the output signal in order to obtain the correct result. In this case, $ASF = k_h - k_y$. The *ASF* is sent to a reconfigurable analog amplifier to restore the true value of the output signals. *ASF* can be presented on 14 bits (limited by the DAC). The first bit is "1" if it is a multiplication by *ASF*, and "0" if it is a division by *ASF*.

Fig. 21 presents the relative error and the SNR between the Xilinx and the theoretical output signals. Table 8 presents the new values of the global relative error and the global SNR.

Table 8 Global relative error and global SNR using *ASF*.

Output	Error (%)	SNR (dB)
with sliding window truncation		
y_1	0.0095	80.40
y_2	0.0095	80.40
with brutal truncation		
y_1	0.0112	79.05
y_2	0.0113	78.96

We notice that after adding the *ASF*, the relative error decrease by 95 % for this specific x input signal, for 8 dB mean time-variation of h and using a brutal truncation. Thus, the use of the sliding window truncation is not recurred.

5.2. The Error versus the Number of Bits of *h*

To decrease the occupation of slices on the FPGA of the time domain architecture, we decrease the number of bits of h (n_h). A study of the average global relative error in function of the number of bits of h is given in Fig. 22.

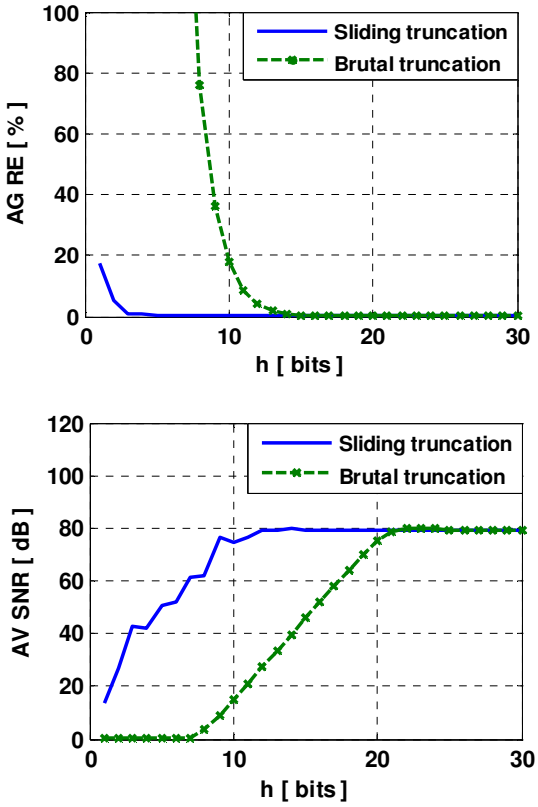


Fig. 22 AG RE and AV SNR versus n_h

We can conclude that, for a number of bits for h bigger than 6 bits, the AG RE is acceptable and the AV SNR is more than 40 dB using sliding truncation.

The number of bits at the output before the truncation is related to the number of bits of h:

$$n_y = n_h + n_x + n_t \tag{34}$$

where n_y is the number of bits at the output, $n_x = 14$ is the number of bits of the input signal and n_t can be expressed by:

$$n_t = \lceil log_2(n_{tap}) \rceil \tag{35}$$

where n_{tap} is the number of taps.

Fig. 23 presents the output signal, the relative error and the SNR using 6 bits for h with sliding window truncation.

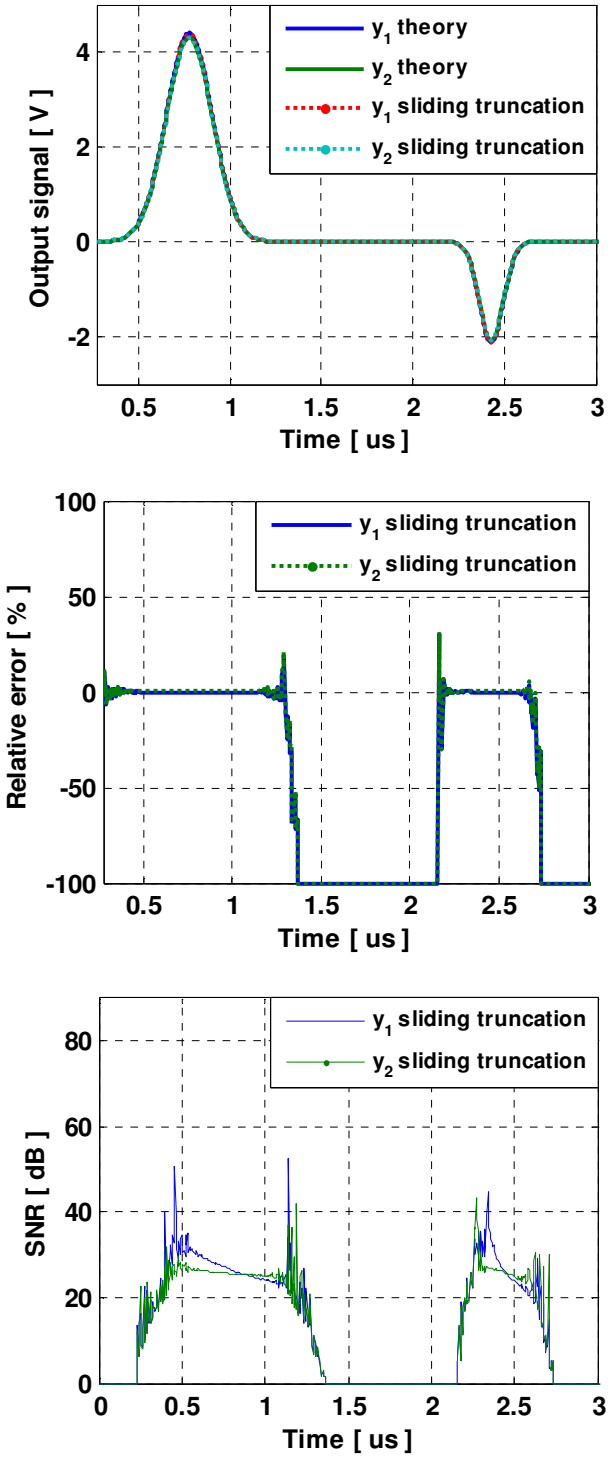


Fig. 23 Output signal, relative error and SNR for n_h = 6 bits

Table 9 summarizes the global relative error and the global SNR using 6 bits for h.

Table 9 The global relative error and the global SNR using 6 bits for *h*.

Output	Error (%)	SNR (dB)
without truncation		
y_1	0.2196	53.14
y_2	0.2633	51.61
with sliding window truncation		
y_1	0.2219	53.05
y_2	0.2657	51.53
with brutal truncation		
y_1	361.39	0.08
y_2	365.03	0.09

For a number of bits for *h* equal to 6 bits, the occupation on the FPGA is reduced from 13 % to 12 %. However, the average global error using a brutal truncation exceeds 100 %, while with a sliding truncation it is 0.75 % which is acceptable. Thus, the sliding truncation is mandatory to use in this case.

Also, the reduction of the number of bits of *h* increases the amount of data transmitted by 60 %. In fact, the PCI bus operates on 32 bits, on each clock pulse, five samples of the impulse response are transmitted (instead of two samples). Thus, 100×(5-2)/5 = 60 %.

6. Conclusions

In this paper, standard channel models have been presented. Also, real impulse responses for 2×2 MIMO propagation channel have been obtained by measurement campaigns carried out on a shipboard environment and for outdoor-to-indoor environment. These impulse responses have been implemented in the digital block of a hardware simulator.

After a comparative study, the time domain architecture used for the design of the digital block represents the best solution, especially for MIMO systems. In fact, it occupies just 13 % of slices on the FPGA Virtex-IV if we compare it to other frequency domain or time domain architectures. Also, it has a small latency of 115 ns.

Moreover, a study of the architecture accuracy for time-varying 2×2 MIMO channel has been presented for TGn model B and for outdoor-to-indoor measured data. It showed that the global relative error does not exceed 0.02 %, using sliding window truncation for both TGn model B and outdoor-to-indoor measurements. Therefore, time-varying impulse responses can be used by the architecture.

Lastly, in order to reduce the error of the output signals and the occupation on the FPGA, two improvement solutions have been presented. The first one uses an auto scale factor. It reduces the global output relative error by 96 % for 8 dB mean time-variation of *h* and using brutal truncation. The second one varies the number of bits used to represent the samples of impulse responses. It reduces the occupation of slices on the FPGA and the amount of data transmitted via the PCI interface for a MIMO profile using sliding truncation.

For our future work, simulations made using a Virtex-VII [3] XC7V2000T platform will allow us to simulate up to 300 SISO channels. In parallel, measurement campaigns will be carried out with the channel sounder realized by IETR, for various types of environments. The final objective of these measurements is to obtain realistic MIMO channel models in order to supply the hardware simulator. A graphical user interface will also be designed to allow the user to reconfigure the simulator parameters.

Acknowledgments

The authors would like to thank the “Région Bretagne” for its financial support of this work, which is a part of PALMYRE-II project.

References

[1] S. Behbahani, R. Merched, A. Eltawil, Optimizations of a MIMO relay network, *IEEE Trans. on Signal Processing*, vol. 56, no. 10, pp. 5062–5073 (2008).

[2] A. Cetiner, E. Sengul, E. Akay, E. Ayanoglu, A MIMO system with multifunctional reconfigurable antennas, *IEEE Antennas Wireless Propag. Lett.,* vol. 5, no. 1, pp. 463–466 (2006).

[3] Xilinx: FPGA, CPLD and EPP solutions, *www.xilinx.com.*

[4] S. Picol, G. Zaharia, D. Houzet, G. El Zein, Design of the digital block of a hardware simulator for MIMO radio channels, *IEEE PIMRC*, Helsinki, Finland (2006).

[5] V. Erceg, L. Shumacher, P. Kyritsi, et al., TGn Channel Models, *IEEE 802.11- 03/940r4* (2004).

[6] Advanced design system – LTE channel model - R4-070872, *Agilent Technologies* (2008).

[7] H. Farhat, R. Cosquer, G. Grunfelder, L. Le Coq, G. El Zein, A dual band MIMO channel sounder at 2.2 and 3.5 GHz, *Instrumentation and Measurement Technology Conference*, Victoria, BC, Canada (2008).

[8] P. Almers, E. Bonek et al., Survey of Channel and Radio Propagation Models for Wireless MIMO Systems, *EURASIP Journal on Wireless Communications and Networking*, Article ID 19070 (2007).

[9] J. Salz, J.H. Winters, Effect of fading correlation on adaptive arrays in digital mobile radio, *IEEE Trans. on Veh. Technol.,* vol. 43, pp. 1049-1057 (1994).

[10] Wireless Channel Emulator, *Spirent Communications* (2006).

[11] Baseband Fading Simulator ABFS, Reduced costs through baseband simulation, *Rohde & Schwarz* (1999).

[12] P. Murphy, F. Lou, A. Sabharwal, P. Frantz, An FPGA Based Rapid Prototyping Platform for MIMO Systems, *Asilomar Conf. on Signals, Systems and Computers*, *ACSSC*, Vol. 1, pp. 900-904, 9-12 (2003).

[13] S. Buscemi, R. Sass, Design of a scalable digital Wireless Channel Emulator for networking radios, *Military Communications Conference, MILCOM*, Charleston, SC, USA (2011).

[14] S. Picol, G. Zaharia, D. Houzet, G. El Zein, Hardware simulator for MIMO radio channels: Design and features of the digital block, *Proc. of IEEE VTC-Fall*, Calgary, Canada (2008).

[15] D. Umansky, M. Patzold, Design of Measurement-Based Stochastic Wideband MIMO Channel Simulators, *IEEE Globecom*, Honolulu (2009).

[16] S. FouladiFard, A. Alimohammad, B. Cockburn, C. Schlegel, A single FPGA filter-based multipath fading emulator, *IEEE Globecom*, Honolulu (2009).

[17] M. Al Mahdi Eshtawie, M. Bin Othma, An algorithm proposed for FIR Filter coefficients representation, *World Academy of Science, Engineering and Technology* (2007).

[18] H. Eslami, S.V. Tran, A.M. Eltawil, Design and implementation of a scalable channel Emulator for wideband MIMO systems, *IEEE Trans. on Vehicular Technology,* vol. 58, no. 9, pp. 4698-4708 (2009).

[19] B. Habib, G. Zaharia, G. El Zein, MIMO Hardware Simulator: Digital Block Design for 802.11ac Applications with TGn Channel Model Test, *IEEE VTC Spring,* Yokohama, Japan (2012).

[20] B. Habib, G. Zaharia, G. El Zein, Digital Block Design of MIMO Hardware Simulator for LTE Applications, *IEEE ICC*, Ottawa, Canada (2012).

[21] K. Borries, E. Anderson, P. Steenkiste, Network-Scale Emulation of General Wireless channels, *IEEE VTC Fall*, San Francisco, USA (2011).

[22] B. Habib, G. Zaharia and G. El Zein, Hardware Simulator: Digital Block Design for Time-Varying MIMO Channels with TGn Model B Test*, IEEE ICT*, Jounieh, Lebanon (2012).

[23] S. Wyne, P. Almers, G. Eriksson, J. Karedal, F. Tufvesson, A.F. Molisch, Outdoor-to-indoor office MIMO measurements at 5.2 GHz, *in Proc. IEEE VTC Fall*, Los Angeles, USA (2004).

[24] R. Roy, T. Kailath, ESPRIT – estimation of signal parameters via rotational invariance techniques, *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 37, no. 7, pp. 984-995 (1987).

[25] J. A. Fressler, A. O. Hero, Space-alternating generalized expectation-maximization algorithm, *IEEE Transactions on Signal Processing*, vol. 42, no. 10, pp. 2664-2677 (1994).

[26] L. Schumacher, K. I. Pedersen, P. E. Mogensen, From antenna spacings to theoretical capacities – guidelines for simulating MIMO systems*, in Proc. PIMRC Conf.*, vol. 2, pp. 587-592 (2002).

[27] Q. H. Spencer, et al., Modeling the statistical time and angle of arrival characteristics of an indoor environment, *IEEE J. Select. Areas Commun.,* vol. 18, no. 3, pp. 347-360 (2000).

[28] C-C. Chong, D. I. Laurenson, S. McLaughlin, Statistical Characterization of the 5.2 GHz wideband directional indoor propagation channels with clustering and correlation properties, *in proc. IEEE Veh. Technol. Conf.,* vol. 1, pp. 629-633 (2002).

[29] B. Habib, G. Zaharia, G. El Zein, MIMO Hardware Simulator: New Digital Block Design in Frequency Domain for Streaming Signals, *Journal of Wireless Networking and Communications,* 2(4): 55-65, doi: 10.5923/j.jwnc.20120204.05 (2012).

[30] ModelSim - Advanced Simulation and Debugging, *http://model.com*.