Hardware simulator design for LTE applications with time-varying MIMO channels

Bachir Habib, Gheorghe Zaharia, Ghaïs El Zein

Hardware Simulator: Digital Block Design for LTE Applications with Time-Varying MIMO Channels

Bachir Habib, Gheorghe Zaharia, Ghais El Zein
Institute of Electronics and Telecommunications of Rennes (IETR) UMR CNRS 6164
INSA, Rennes, France
bachir.habib@insa-rennes.fr

***Abstract*—A hardware simulator facilitates the test and validation cycles by replicating channel artifacts in a controllable and repeatable laboratory environment. Thus, it makes possible to ensure the same test conditions in order to compare the performance of various equipments. This paper presents new frequency domain and time domain architectures of the digital block of a hardware simulator of MIMO propagation channels. The two architectures are tested with LTE standard, in outdoor environment, using time-varying channel models. After the description of the general characteristics of the hardware simulator, the new architectures of the digital block are presented and designed on a Xilinx Virtex-IV FPGA. Their accuracy and latency are analyzed. The result shows that the architectures produce low occupation on the FPGA and decrease the error at the output. Therefore, they present the best solution to simulate systems with high MIMO arrays order.**

***Index Terms*—Hardware simulator; FPGA; Time-varying MIMO radio channels; LTE**

I. INTRODUCTION

HARDWARE simulators of mobile radio channel are very useful for the test of wireless communication systems [1], [2]. The current communication standards indicate a clear trend in industry toward supporting Multiple-Input Multiple-Output (MIMO) functionality. In fact, several studies published recently present systems that reach an order of 8×8 and higher [3]. This is made possible by advances at all levels of the communication platform [4], [7].

With the continuous increase of field programmable gate array (FPGA) capacity, entire baseband systems can be efficiently mapped onto faster FPGAs for more efficient testing and verification. As shown in [5] and [6], the FPGAs provide the greatest flexibility in algorithm design. The present simulator is reconfigurable with standard bandwidths not exceeding 100 MHz (the maximum value for FPGA Virtex-IV). However, in order to exceed 100 MHz bandwidth, more performing FPGA as Virtex-VII can be used [7].

The simulator is configured with the Long Term Evolution System (LTE) and Wireless Local Area Networks (WLAN) 802.11ac standards. The channel models used by the simulator can be obtained from standard channel models, as the TGn 802.11n [8] and LTE [9] models, or from real measurements by using a time domain MIMO channel sounder designed at IETR [10], [11].

At IETR, several architectures of the digital block of a hardware simulator have been studied in [12] and [13]. In [14], a method based on determining the parameters of the simulator by fitting the space time-frequency cross-correlation matrix to the estimated matrix of a real-world channel was presented. This solution shows that the error can be important.

Typically, wireless channels are simulated using finite impulse response (FIR) filters, as in [13], [15] and [16]. The Fast Fourier Transform (FFT) modules are used to obtain an algebraic product, as in [12], [14] and [15].

In this paper, we present a study of two alternative approaches. The first approach performs in frequency domain, while the second approach is based on FIR filter. The main contributions of the paper are:

1) The considered frequency architecture in [12], [14] and [15] operates correctly for signals with a number of samples not exceeding the size of the FFT. Thus, in this paper, new frequency domain architecture avoiding this limitation and working in streaming mode is presented and tested with time varying LTE models.
2) The time domain architecture presented in [13] and [15] produce an occupation of 11 % to 13 % of slices on the FPGA for one SISO channel. However, in this paper, we present a time domain architecture with an occupation of 3 % for one SISO channel and up to 60 % for a MIMO 4×5 systems.
3) In general, the frequency responses can be presented in baseband with a complex envelope, or with the real signal with limited band between f_c - B and f_c + B, where f_c is the carry frequency and B in the bandwidth. In this paper, to eliminate the complex multiplication and the f_c, the hardware simulation are made between Δ and B + Δ, where Δ depends on the filter and it is introduced to prevent the overlap of the positive and the negative sides of the frequency responses.

4) For indoor environments, tests have been made for a SISO channel [17] and for a time varying MIMO channel [18] using 802.11ac signals. However, in this paper, tests are made for outdoor environment with LTE signals for a MIMO 2×2 time varying channel.
5) For the time domain architecture, studies are made for the first time relating the number of bits of the impulse responses to the error at the output. Thus, it is possible to have a trade-off between the occupation on the FPGA and the error.

The rest of this paper is organized as follows. Section II presents the architectures for a SISO channel. Section III describes their hardware implementation. Moreover, the accuracy of these two architectures is analyzed. Section IV presents a discussion. Lastly, Section V summarizes some conclusions and some prospects.

II. Principle, Architecture and Operation

The design of the radio frequency (RF) blocks of the simulator was realized during a previous work [12], [13]. PALMYRE II project mainly concerns the MIMO channel models and their hardware implementation into the simulator.

The channel models used by the simulator can be obtained from standard channel models or from real measurements by using a time domain MIMO channel sounder designed at our laboratory and shown in Fig. 1.

Fig. 1. MIMO channel sounder: receiver and transmitter.

However, in this paper, the hardware simulation will be made by using the LTE channel models which are popular and well known.

A. Standard LTE Channel Models

An overview on the channel models is given in [19]. The LTE models is used for mobile wireless applications and covers the most used scenarios for LTE applications. A set of 3 channel models are proposed in [9]: The Pedestrian A model (EPA), the Vehicular A model (EVA) and the Typical Urban model (ETU). The sampling frequency is f_s = 50 MHz and the sampling period is $T_s = 1/f_s = 20$ ns.

B. Time-Varying EVA Channel Model

In the MIMO context, little experimental results have been obtained regarding channel time-variation, partly because of limitations in channel sounding equipment [19]. For outdoor environments, the user terminal is typically moving.

At a center frequency of 1.8 GHz and an environmental speed at 80 km/h, the Doppler spread is f_d = 133 Hz. Thus, in this paper, we have chosen a refresh frequency f_{ref} = 0.3 kHz.

To vary the channel, we consider a 2×2 MIMO Rayleigh fading channel.

The result signal magnitude can be characterized by two parameters:

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

The ratio P_c / P_s is called Ricean K-factor and it is often represented in decibels.

Assuming all channel coefficient in the channel matrix H are Rice distributed with mean power P_c and mean power P_s, the MIMO channel matrix H for each tap can be expressed by:

$$H = \sqrt{P_c}\,.\,H_F + \sqrt{P_s}\,.\,H_v \tag{1}$$

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

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

$$P_c = P\,.\frac{K}{K+1} \tag{2}$$

$$P_s = P\,.\frac{1}{K+1} \tag{3}$$

where K is the Ricean factor.

Moreover, if we combine (2) and (3) in 1 we obtain:

$$H = \sqrt{P}.\left(\sqrt{\frac{K}{K+1}}H_F + \sqrt{\frac{1}{K+1}}H_V\right) \tag{4}$$

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

$$H = \sqrt{P}.H_V \tag{5}$$

where P is the power of each tap. For EVA channel model, P is given in [9] for each of the nine taps.

For 2 transmit and 2 receive antennas:

$$H = \sqrt{P}.\begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix} \tag{6}$$

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.
The vector H_V can be divided into a covariance matrix and a vector spatially white Rayleigh Independent and identically distributed MIMO channel:

$$vector(H_V) = R^{1/2}.vector(H_w) \quad (7)$$

H_w and R must be calculated.

H_w is a Rayleigh fading matrix of independent zero mean, unit variance, complex Gaussian random variables. The method for generating the Rayleigh random with the desired temporal correlation is:

1) Generation of two sequences (x_{1p} and x_{2p}) of complex Gaussian random variables from 0 to f_d.
2) We take the complex conjugate (x_{1c} and x_{2c}) of these sequences to generate the complex Gaussian random variables for the negative part from $-f_d$ *to* 0.
3) Therefore we obtain $x_1 = x_{1p} + x_{1c}$ and $x_2 = x_{2p} + x_{2c}$.
4) We multiply the above complex Gaussian sequences (x_1 and x_2) with the root of the Doppler Spectrum S given for LTE models and generated from $-f_d$ to f_d :

$$S(f) = \frac{3/2}{\pi.f_d.\sqrt{1-\left(\frac{f}{f_d}\right)^2}} \quad (8)$$

5) To obtain the signals in time domain, we take the IFFT of the two signals above resulting in time domain signals x and y.
6) We define r_i equal to $\sqrt{x^2 \times y^2}$.

r_i is an element of the H_w matrix and it is the desired Rayleigh distributed envelope with the required temporal correlation.

LTE has defined the correlation for all four channels which considered identically distributed and normalized providing unitary average energy:

$$R = \begin{bmatrix} 1 & \alpha_1 & \beta_1 & s_1 \\ \alpha_1^* & 1 & s_2 & \beta_2 \\ \beta_1^* & s_2^* & 1 & \alpha_2 \\ s_1^* & \beta_2^* & \alpha_2^* & 1 \end{bmatrix} \quad (9)$$

α_1, α_2 represent the correlations between channels at two receive antennas at one side, but originating from the same transmit antenna (MISO). β_1, β_2 represent the correlations between channels at two transmit antennas at one side, but originating from the same receive antenna (SIMO). s_1, s_2 are the cross-correlation between antennas of the same side of the link.

For simplification, we consider:

1) $\alpha_1 = \alpha_2$ and $\beta_1 = \beta_2$, therefore they can be denoted as α and β (T_x (resp. R_x) correlation coefficients are (in magnitude) independent from the considered R_x (resp. T_x) antenna).
2) $s_1 = \alpha \times \beta$ and $s_2 = \alpha^* \times \beta$.

Then it is possible to define a 2×2 transmit and receive correlation matrices, R_t and R_r, as to decompose any MIMO system into two interconnected MISO/SIMO sub systems. This decomposition is resulted in the development of simpler and less general model of the covariance Matrix:

$$R = R_t \otimes R_r \quad (10)$$

Where $\otimes$ is the Kronecker product, and R_t and R_r are the correlation matrices at the transmitter and the receiver respectively, and they are defined by:

$$R_t = \begin{bmatrix} 1 & \alpha \\ \alpha^* & 1 \end{bmatrix} \quad , \quad R_r = \begin{bmatrix} 1 & \beta \\ \beta^* & 1 \end{bmatrix} \quad (11)$$

α and β are defined by the LTE models as:

1) Low Correlation: α =0, β =0.
2) Medium Correlation: α =0.3, β =0.9.
3) High Correlation: α =0.9, β =0.9.

In this paper, we will consider the high correlation, therefore R is considered:

$$R = \begin{bmatrix} 1 & 0.9 & 0.9 & 0.81 \\ 0.9 & 1 & 0.81 & 0.9 \\ 0.9 & 0.81 & 1 & 0.9 \\ 0.81 & 0.9 & 0.9 & 1 \end{bmatrix} \quad (12)$$

Thus, for each tap P, H can be calculated by:

$$H = \begin{pmatrix} h_{11} \\ h_{12} \\ h_{21} \\ h_{22} \end{pmatrix} = \sqrt{P}.\, R^{1/2}.\begin{pmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \end{pmatrix} \quad (13)$$

To calculate $R^{1/2}$ we must first calculate the eigenvalues (L1, L2, L3, L4) and the eigenvectors Q of R. Then we have:

$$R = Q^{-1}.\begin{bmatrix} L1 & 0 & 0 & 0 \\ 0 & L2 & 0 & 0 \\ 0 & 0 & L3 & 0 \\ 0 & 0 & 0 & L4 \end{bmatrix}.Q \quad (14)$$

And

$$\sqrt{R} = Q^{-1}.\begin{bmatrix} \sqrt{L1} & 0 & 0 & 0 \\ 0 & \sqrt{L2} & 0 & 0 \\ 0 & 0 & \sqrt{L3} & 0 \\ 0 & 0 & 0 & \sqrt{L4} \end{bmatrix}.Q \quad (15)$$

Table I presents the relative power for the four sub-channels impulse responses: h_{11}, h_{12}, h_{21}, h_{22}, at a given time, with f_{ref} = 0.3 kHz between the successive profiles.

TABLE I
RELATIVE POWER OF MIMO 2X2 IMPULSE RESPONSES

Tap index	Excess delay [nT$_s$]	h_{11} - *RP* [dB]	h_{12} - *RP* [dB]	h_{21} - *RP* [dB]	h_{22} - *RP* [dB]
1	**0T_s**	-3.85	-3.92	-3.68	-3.90
2	**2T_s**	-2.22	-2.87	-2.27	-3.04
3	**8T_s**	-3.26	-2.55	-2.73	-2.05
4	**16T_s**	-7.89	-8.46	-7.15	-7.57
5	**19T_s**	-4.04	-4.46	-3.57	-4.43
6	**36T_s**	-8.50	-8.84	-8.24	-8.77
7	**55T_s**	-7.28	-6.35	-6.84	-5.70
8	**87T_s**	-8.56	-9.02	-9.01	-9.50
9	**126T_s**	-12.30	-12.50	-12.68	-12.21

C. Digital Block

In this section, an improved frequency architecture and a time domain architecture based on a FIR filter are presented.

1) *New Frequency Domain Architecture*

The new frequency architecture presented in Fig. 2 has been verified with Gaussian impulse signals [20]. It operates correctly for signals with a number of samples exceeding *N*, where $N = 2^n$ is the size of the FFT module.

For EVA channel, the largest excess delay is 126 samples. Thus, N= 128 samples. However, it is mandatory to extend each partial input of *N* samples with a "tail" of *N* null samples, as in [20], to avoid a wrong result. Therefore, the FFT/IFFT modules operate with 256 samples.

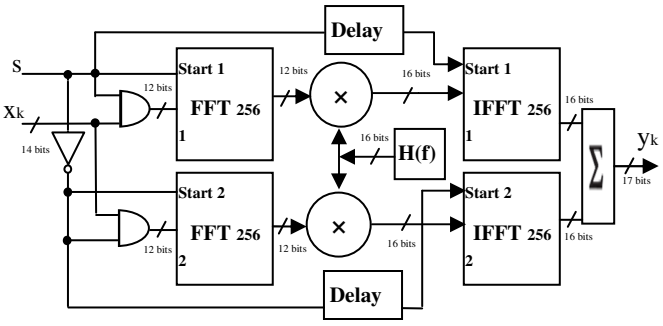

Fig. 2. Frequency architecture for a SISO channel.

Due to the use of a 14-bit digital-to-analog convertor (DAC), the final output must be truncated. The immediate solution is to use the "brutal" truncation which keeps the 14 first bits. However, a better solution is the sliding truncation presented in Fig. 3 which uses the 14 most significant bits.

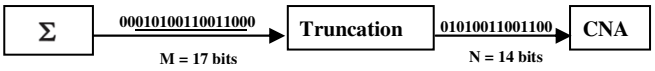

Fig 3. Sliding window truncation from 17 to 14 bits.

2) *Time Domain Architecture*

For ETU, N = 126 samples and it imposes the use of 9 multipliers. Fig. 4 presents a FIR filter 126 with 9 multipliers.

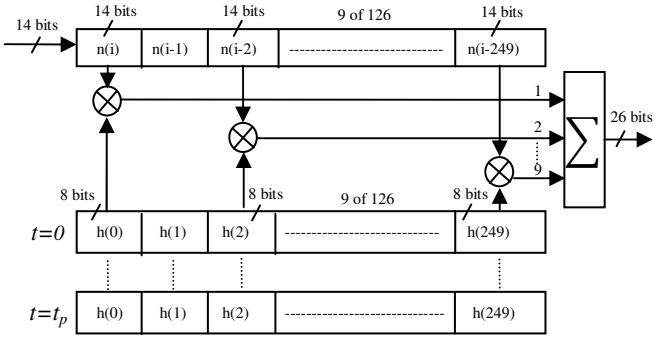

Fig. 4. FIR 126 with 9 multipliers for a SISO channel.

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 a FIR 126 with 9 multipliers is:

$$y_q(i) = \sum_{k=1}^{9} h_q(i_k).x_q(i.t_s - i_k t_s), i \in N \qquad (16)$$

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 $i_k T_s$.

III. IMPLEMENTATION AND TESTS

Fig. 5 shows the XtremeDSP Virtex-4 board from Xilinx [7] used to implement the simulator, and described in [20].

Fig. 5. XtremeDSP Development board Kit-IV for Virtex-IV.

The simulations and synthesis are made with Xilinx ISE [7] and ModelSim software [21].

A. Implementation and Results of the Frequency Architecture

As the 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, four SISO channels in frequency domain are needed to simulate a one-way 2×2 MIMO radio channel. Fig. 6 shows the connection between the computer and the FPGA board to reload the coefficients.

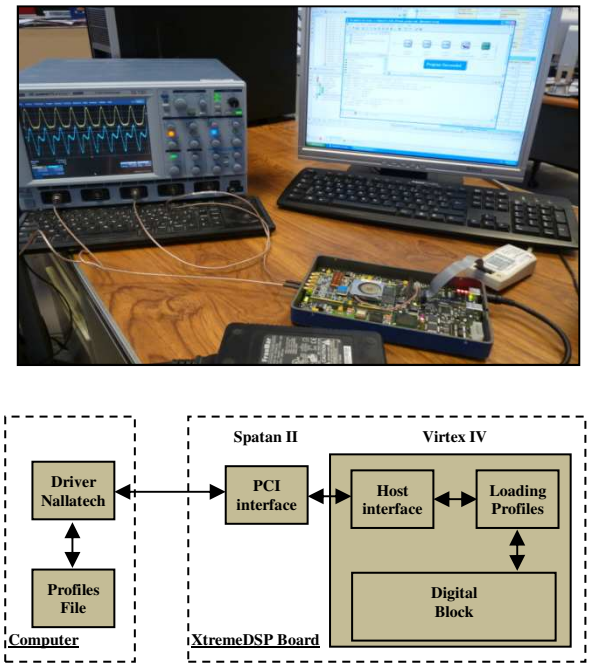

Fig. 6. Connection between the computer and the XtremeDSP board.

The refreshing period is (1/0.3) ms during which we must refresh all of the four profiles, i.e. (256+1)×4 = 1028 words of 32 bits = 4112 bytes to transmit for one profile, which is: 4112×0.3KHz = 123.360 KB/s. The PCI bus is chosen to load the profiles. It has a speed up to 30 MB/s.

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

TABLE II
VIRTEX-IV SX35 UTILIZATION FOR MIMO 2×2 PING-PONG FREQUENCY ARCHITECTURE

Number of slices	10,965 out of 15,360	72 %
Number of blocs RAM	93 out of 192	49 %
Number of multipliers	127 out of 192	66 %

B. Implementation and Results of Time Domain Architecture

For the time domain architecture, the amount of data transmit for a profile is: (9+1)×4 = 40 words of 8 bits = 40 bytes, which is: 40×0.3KHz = 12 kB/s.

Table III shows the device utilization for four FIR filters 250 with 9 multipliers, in one V4-SX35.

TABLE III
VIRTEX-IV SX35 UTILIZATION FOR MIMO 2×2 FIR ARCHITECTURE

Number of slices	1,821 out of 15,360	12 %
Number of blocs RAM	36 out of 192	19 %
Number of multipliers	36 out of 192	19 %

C. Accuracy of the Architectures

In order to determine the accuracy of the digital block, a comparison is made between the theoretical and the Xilinx output signal. An input Gaussian signal *x(t)* is considered and long enough to be used in streaming mode:

$$x(t) = x_m e^{-\frac{(t-m_x)^2}{2\sigma^2}}, 0 \le t \le 3W_t \quad (17)$$

where $n_{final} = 126T_s$, $W_t = NT_s$, $m_x = 3$. n_{final} /16 and $\sigma = m_x/12$.

For EVA model, the impulse response has 9 paths. The A/D and D/A convertors of the development board have a full scale $[-V_m, V_m]$, with V_m = 1 V. For the simulations, we consider $x_m = V_m/2$. The theoretic output signals are:

$$y_1(t) = \sum_{k=1}^{9} h_{11}(i_k).x(t - i_k t_s) + \sum_{k=1}^{9} h_{21}(i_k).x(t - i_k t_s) \quad (18)$$

$$y_2(t) = \sum_{k=1}^{9} h_{12}(i_k).x(t - i_k t_s) + \sum_{k=1}^{9} h_{22}(i_k).x(t - i_k t_s) \quad (19)$$

The relative error is computed for each output sample by:

$$\varepsilon(i) = \frac{Y_{xilinx}(i) - Y_{theory}(i)}{Y_{theory}(i)}.100\ [\%] \quad (20)$$

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

$$SNR(i) = 20.log_{10}\left|\frac{Y_{theory}(i)}{Y_{xilinx}(i) - Y_{theory}(i)}\right|\ [dB], i = \overline{1, 3N + \iota_9} \quad (21)$$

Fig. 7 presents the Xilinx output signals, the SNR and the relative error, for the two successive packs for the MIMO 2×2 frequency architecture.

The relative error is high only for small values of the output signal because the Gaussian signal test is close to 0. Thus, in this case, the Xilinx output signal is smaller than $\Delta x = \Delta y = 2 \times 0.5/2^{14}$ used for the A/D conversion of the signals.

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

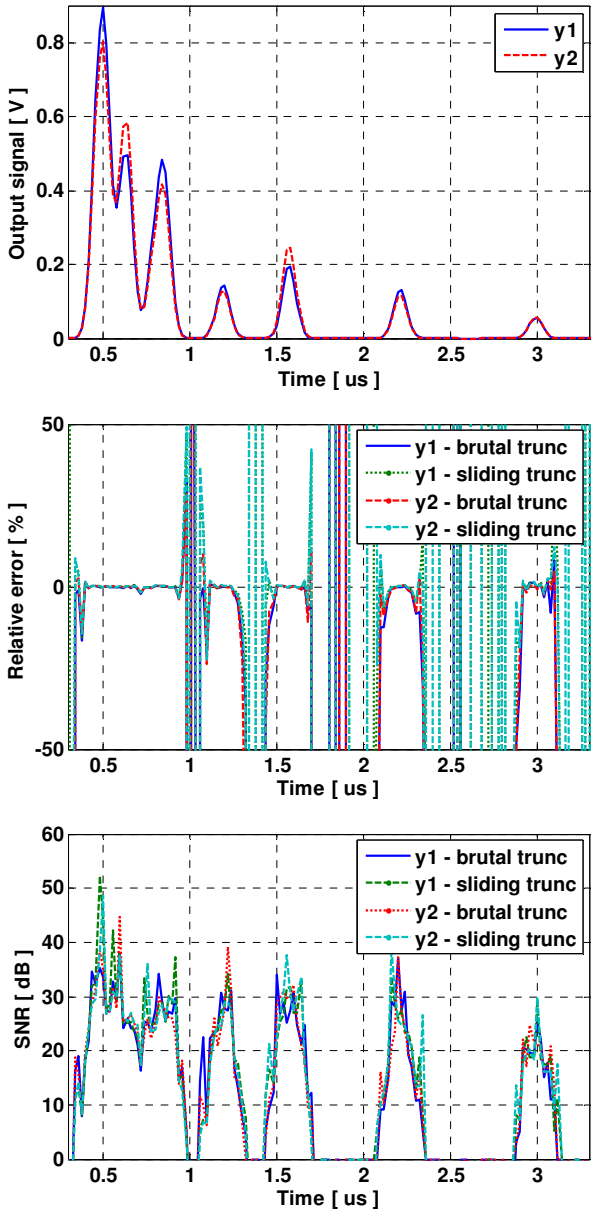

Fig. 7. Xilinx output signals, SNR and relative error, using frequency domain architecture.

The global values of the relative error and of the global SNR of the output signal before and after the final truncations are necessary to evaluate the accuracy of the architectures.

The global relative error and SNR are computed by:

$$\varepsilon = \frac{\|E\|}{\|Y_{theory}\|} \times 100\ [\%] \tag{22}$$

$$SNR_g = 20 \times log_{10} \frac{\|Y_{theory}\|}{\|E\|}\ [dB] \tag{23}$$

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:

$$\|x\| = \sqrt{\frac{1}{L}\sum_{k=1}^{L} x_k^2} \tag{24}$$

Fig. 8 presents the Xilinx output signal, the SNR and the relative error, for the two successive packs for the MIMO 2×2 FIR architecture.

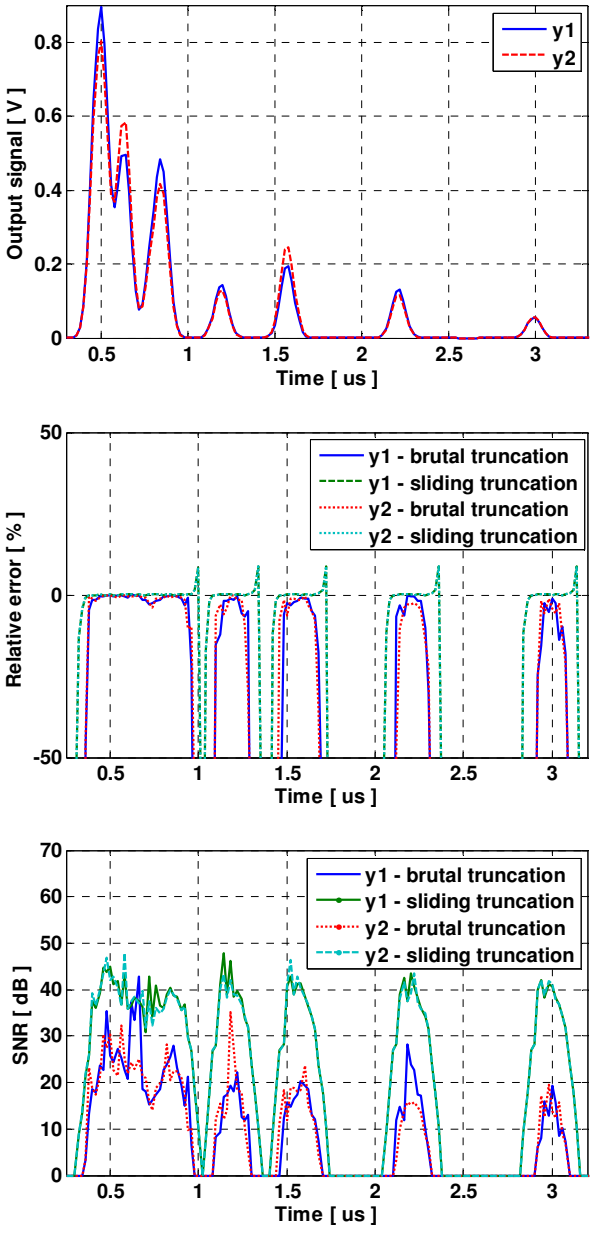

Fig. 8. Xilinx output signals, SNR and relative error, using time domain architecture.

Table IV shows the global values of the relative error and SNR for the two successive packs using the MIMO 2×2 architecture in frequency and time domains.

TABLE IV
THE GLOBAL RELATIVE ERROR AND THE SNR

	Y 1		Y 2	
	Error (%)	SNR (dB)	Error (%)	SNR (dB)
MIMO 2×2 FREQ. DOMAIN ARCHITECTURE	**WITHOUT TRUNCATION**			
	0.4537	46.8665	0.4641	46.6728
	WITH SLIDING WINDOW TRUNCATION			
	0.4539	46.8640	0.4642	46.6706
	WITH BRUTAL TRUNCATION			
	0.4839	46.3035	0.4849	46.2872

	Y 1		Y 2	
	Error (%)	SNR (dB)	Error (%)	SNR (dB)
MIMO 2×2 TIME DOMAIN ARCHITECTURE	**WITHOUT TRUNCATION**			
	0.0150	76.4985	0.0157	76.0898
	WITH SLIDING WINDOW TRUNCATION			
	0.0151	76.3938	0.0159	75.9875
	WITH BRUTAL TRUNCATION			
	0.8478	41.3976	0.9425	40.4688

IV. DISCUSSION

With the frequency architecture presented in Fig. 2, it is not possible to modify the number of bits of *H* to reduce the slice occupation on the FPGA. In fact:

1) The global error presented in Table IV increases brutally for a small reduction of the number of bits.
2) The occupation on the FPGA presented in Table III will decrease about 10 % which will not have an effect on implementing a higher order of MIMO systems.

For the frequency architecture, the results are given in Table IV. The sliding truncation reduces just 16 % of the error with brutal truncation. Thus, the brutal truncation is more suitable to use. Also, it offers a reduction of the slice occupation on the FPGA and it avoids the need of a reconfigurable analog amplifier after the DAC.

With the time domain architecture, while reducing the number of bits of *h*, the global relative error increases as presented in Fig. 9.

Moreover, the number of bits at the output before the truncation is equal to the number of bits of *h* plus the number of bits of the input signal (14 bits) plus the power of 2 of the number of taps (9 taps give 4 bits).

From Fig. 9, we can conclude that for a number of bits for *h* higher than 8 bits, the average error is acceptable when using the sliding window truncation and the global SNR is more than 50 dB. By reducing the number of bits of *h* from 16 to 8, we reduce the occupation on the FPGA from 12 % to 11.6 % which is not a big deference.

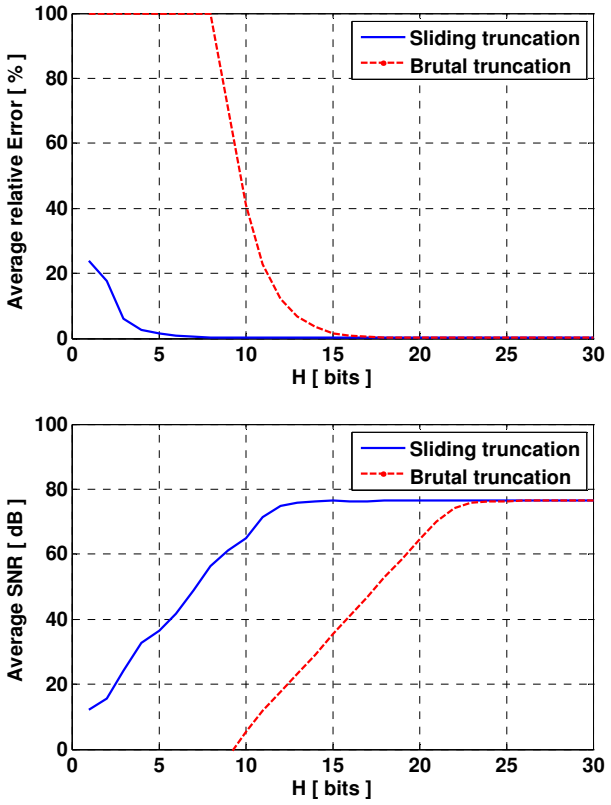

Fig. 9. Average global relative error and average global SNR versus the number of bits of *h*.

The goal is to compare the frequency architecture with the time domain architecture, by considering three points: the precision, the FPGA occupation and the latency.

If we compare the results in Table IV, we observe that the SNR is higher with the time domain architecture with sliding window truncation. However, with the frequency domain architecture the SNR is higher using the brutal truncation.

According to Tables II and III, the time domain architecture presents a slice occupation of 12 % on the FPGA Virtex-IV, which is better than the occupation of the frequency architecture (72 %). Thus, in time domain, 4×5 MIMO channels can be implemented on a single Virtex-IV.

The time domain architecture presents another advantage by generating a latency of 103 ns for each simulated profile. However, the frequency architecture has a latency of 8.8 µs.

V. CONCLUSION

After a comparative study, in order to reduce the occupation on the FPGA, the error of the output signals and the latency of the digital block, the time domain architecture presents the best solution, especially for MIMO systems.

Simulations will be made using a Virtex-VII [7] XC7V2000T platform will allow us to simulate systems with high MIMO arrays order. Measurement campaigns will also be carried with the MIMO channel sounder realized by IETR, for various types of environments. A Graphical User Interface will also be designed to allow the user to select the

propagation environment, to select the channel model and to reconfigure the channel parameters. The final objective of this work is to simulate realistic propagation channel for different MIMO standards and environments.

VI. ACHNOWLEDGMENTS

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] Wireless Channel Emulator, Spirent Communications, 2006.

[2] Baseband Fading Simulator ABFS, Reduced costs through baseband simulation, Rohde & Schwarz, 1999.

[3] A. S. Behbahani, R. Merched, and A. Eltawil, “Optimizations of a MIMO relay network,” IEEE, Trans. on Signal Proc, vol. 56, no. 10, pp. 5062–5073, Oct. 2008.

[4] B. A. Cetiner, *et al.*, “A MIMO system with multifunctional reconfigurable antennas,” IEEE Antennas Wireless Propag. Lett., 2006.

[5] 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 Nov. 2003.

[6] P. Murphy, F. Lou, J. P. Frantz, "A hardware testbed for the implementation and evaluation of MIMO algorithms", Conf. on Mobile and Wireless Communications Networks, Singapore, 27-29 Oct. 2003.

[7] "Xilinx: FPGA, CPLD and EPP solutions", www.xilinx.com.

[8] V. Erceg, L. Shumacher, P. Kyritsi, et al., “TGn Channel Models”, IEEE 802.11- 03/940r4, May 10, 2004.

[9] Agilent Technologies, “Advanced design system – LTE channel model - R4-070872 3GPP TR 36.803 v0.3.0”, 2008.

[10] G. El Zein, *et al.*, “Characterization and modeling of the MIMO propagation channel: an overview”, ECWT, Paris, 3-4 Oct. 2005.

[11] H. Farhat, R. Cosquer, G. El Zein, "On MIMO channel characterization for future wireless communication systems", 4G Mobile & Wireless Comm.Tech., River Publishers, Aalborg, Denmark, 2008, pp. 225-233.

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

[13] S. Picol, G. Zaharia, D. Houzet and G. El Zein, "Hardware simulator for MIMO radio channels: Design and features of the digital block", Proc. of IEEE VTC-Fall 2008, 21-24 Sept. 2008, Calgary, Canada.

[14] D. Umansky, M. Patzold, "Design of Measurement-Based Stochastic Wideband MIMO Channel Simulators", IEEE Globecom, Hawai, 2009.

[15] H. Eslami, *et al.*, "Design and implementation of a scalable channel Emulator for wideband MIMO systems", IEEE Trans. on Vehicular Technology, 2009.

[16] S. Fouladi Fard, A. Alimohammad, B. Cockburn, C. Schlegel, "A single FPGA filter-based multipath fading emulator", Globcom, Canada, 2009.

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

[18] 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, In press(2012).

[19] P. Almers, E. Bonek *et al.*, "Survery of Channel and Radio Propagation Models for Wireless MIMO Systems", EURASIP Journal on Wireless Communications and Networking, Volume 2007, Article ID 19070.

[20] B. Habib, G. Zaharia and G. El Zein, "Improved Frequency Domain Architecture for the Digital Block of a Hardware Simulator for MIMO Radio Channels", IEEE ISSCS, Iasi, Romania, June 2011.

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