Hardware-in-the-Loop based fast system verification and parameter fine tuning platform

Aritz Alonso[1, a], Andoni Irizar[1, b], Ainhoa Cortés[1, c], Jorge Alberto Paredes [1, d], Marta Turrillas[1, e]

[1]Department of Electronics & Comunications

CEIT and Tecnun (University of Navarra)

Manuel de Lardizábal 15, 20018 San Sebastián, Spain

[a]aadomingo@ceit.es, [b]airizar@ceit.es, [c]acortes@ceit.es, [d]jparedes@ceit.es, [e]mturrillas@ceit.es

Keywords: OFDM, Hardware-in-the-Loop, co-simulation, verification.

Abstract. The time required for system analysis and verification has invariably become the bottleneck of the development process as designs become more complex. Methodologies involving configurable hardware have proven to be the only ones to break the inverse relationship between accuracy in simulation and performance in verification. In this work we present a Hardware-in-the-Loop based platform that substantially reduces the time required for system verification and parameter fine tuning. The verification platform has been used to evaluate the performance of the physical layer of a WLAN 802.11a compliant transceiver. Compared to a software RTL simulation this platform reduces simulation time by a factor of at least 10^3.

Introduction

In 1965 Gordon E. Moore published [1] his well-known observation that states that the number of transistors that can be fitted inexpensively on a silicon device doubles every two years. This observation, which was introduced in the early days of the semiconductor technology, continues to hold true nowadays. Consequently, circuit designs are expected to continuously grow in complexity and include more advanced and accurate functions.

The increase in the complexity of the designs has some severe drawbacks as well: bigger projects require longer and more meticulous design processes; implementation time increases and demands a thorough verification stage. Moreover, the execution time of optimization stages where a number of parameters (memory and cache sizes …) has to be determined can become unacceptable [2].

With the improvement of the synthesis tools, the verification process has become the design bottleneck and can suppose about the 60% or the 70% of the development time [3]. Any reduction in the verification stage can lead to a considerable shorter time to market.

Many techniques have arisen to speed up simulation and reduce verification time. Among them, methodologies involving configurable hardware have proven to be the only ones to break the inverse relationship between accuracy in simulation and performance in verification stages [4]. These methodologies, also known as as *hardware-in-the-loop* simulations, lie in the principle of executing the most complex and time consuming tasks of the design on dedicated hardware while the lightest ones are executed in software.

Hardware-in-the-loop simulations could theoretically be executed at real-time speeds. However, communication between the hardware and software layers has shown to be the limiting factor of the simulation acceleration the platform can achieve [5].

Hardware-software co-simulation has been used for a variety of purposes such as hardware model verification [6] and algorithm calculation acceleration [7]. Obtaining the *packet error rate* (PER) curves of a transceiver is a time consuming task wich could be improved significantly with the use of *hardware-in-the-loop* simulations.

In this work we present a *System Generator* based hardware-software platform for fast transceiver verification. The platform uses an Ethernet connection that reaches communication speeds of up to 1 Gbps, much faster than what series connections RS232 [8] or USB 2.0 [9] can achieve. The platform is suitable for medium-size projects and requires less effort to configure and control the simulation than those of proposed works [10] [11].

The proposed hardware-software verification platform has been used to analyse the performance of a WLAN 802.11a compliant transceiver [12] measuring its PER curves under real communication channels. The platform has also been used to quickly study the impact that critical parameters have on the transceiver size and performance.

Transceiver architecture

WLAN 802.11a was one of the first standards to be built upon the now common *orthogonal frequency division multiplexing* (OFDM) scheme. OFDM distributes the information to be transmitted between a series of orthogonal carriers so that a single high-speed data stream is divided into multiple slower data-streams that are transmitted in parallel through a channel. The inverse and direct FFT algorithms have proven to be an efficient way to achieve the carrier orthogonality at the transmitter and receiver sides.

Figure 1 shows the architecture of the WLAN 802.11a compliant transceiver to be verified. As it can be seen several submodules are multiplexed in the transmission and reception chains to reduce hardware resource consumption. The FFT core is the most versatile submodule for it implements the inverse FFT algorithm in transmission mode and the direct FFT algorithm in reception mode, as well as supporting the synchronization stage.

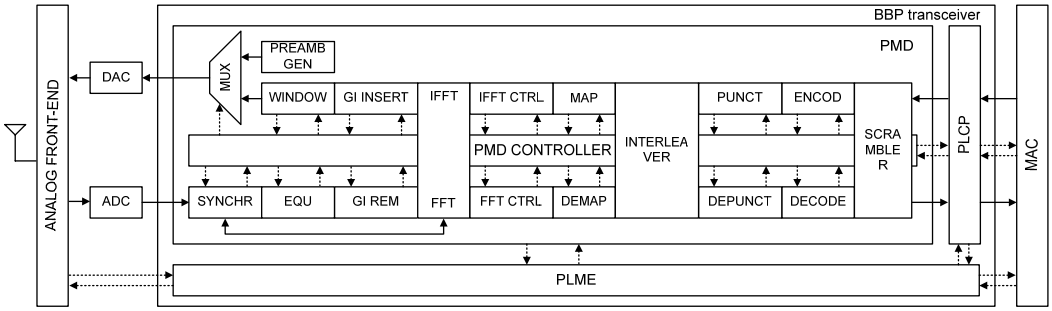

Fig. 1 - Block diagram of the transceiver architecture

Fast Verification Platform

The fast verification platform is based on System Generator [13], a component of the Xilinx ISE Design Suite that enhances Matlab's Simulink platform. Simulink offers a powerful design environment for multidomain simulation. System Generator adds a variety of new block libraries specially designed for digital signal processing.

Among its features, System Generator allows to import custom IPs in HDL and to compile designs to run them on real time under Xilinx FPGAs. Several evaluation boards are supported by default, but the environment allows any board if proper configuration files are supplied by the user and the board meets certain hardware requirements. System Generator also provides an Ethernet or JTAG based interface for communication between the hardware and software layers of the simulation which is transparent to the user.

All this reasons make System Generator an interesting tool when designing hardware-in-the-loop simulations. Figure 2 depicts the block diagram of the platform that was designed to verify the performance of hardware transceiver implementations.

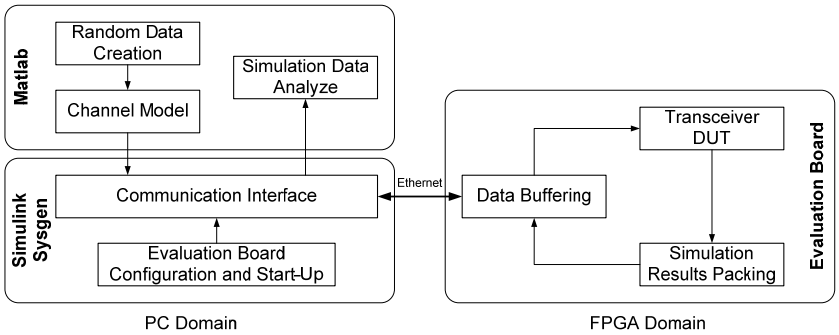

Fig. 2 - Architecture of the fast prototyping sytem

The software component in the simulation is responsible for creating *PLCP Protocol Data Units* (PPDU) with random data and implementing the WLAN multipath channel with different values of noise and delay spread. The transceiver architecture is implemented in the hardware component of the simulation. It is responsible for decoding the received PPDU and counters the number of erroneous bits to release overhead to the software layer. It also returns information about the status of the transceiver (time and frequency offset estimations …) to the software layer for statistical analysis. System Generator is responsible for waking up and configuring the evaluation board and synchronizing the software and hardware components of the simulation.

Given that the multipath channel model is implemented in software its statistical properties can get closer to a real channel than those obtained with embedded implementations. Moreover, it gives the possibility to evaluate the design in a more accurate scenario and not only on an Additive White Gaussian Noise (AWGN) channel like in [14]. The transceiver is also provided with an equalizer, so the design is self-sufficient and is capable of obtaining the channel state information as opposed to [6].

Figure 3 gives more information about the underlying hardware layer.

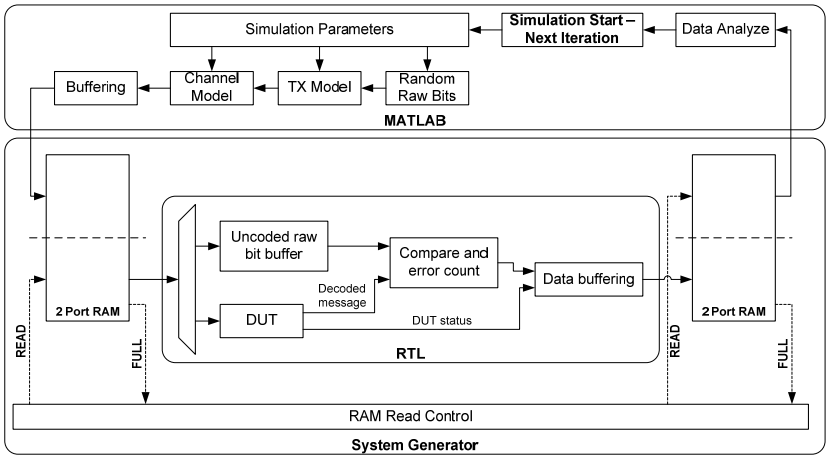

Fig. 3 - Simulation flow dissection

Data buffering between the software and hardware layers is executed by means of double port RAMs available in the FPGA. Each port of the RAM is controlled exclusively by the software or the hardware layer, and blocking mechanisms have been provided so that no layer can gain access to the RAM while the other layer is using it.

Matlab and Simulink provide a data flow that contains the source random bits and the transmitted PPDU signal with the *received signal strength indicator* (RSSI). 12 bits are used to represent the I and Q components of the PPDU and the RSSI level is transmitted using 8 bits.

The RTL description of the hardware layer stores the source random bits in a temporal buffer and sends the PPDU and RSSI signals to the transceiver. The decoded bit stream sequence is compared on real time with that stored in the temporal buffer and the number of erroneous bits is

obtained. The error count along with other receiver parameters are stored in a specific format and sent back to Matlab with the output double port RAM buffer.

Transceiver parameter study

Given a software model of the transceiver architecture, finding a balance between precision in the operations and area consumption is sometimes a time consuming task. Other times adjusting the size of internal accumulators depends on the working conditions of the device (channel characteristics and behaviour of different submodules of the entity). This section explores the use of the verification platform as a quantization effect analyser: due to the time reduction hardware-in-the-loop simulations can achieve it seems logical to use them in tune-up stages.

The parametric study was carried out to minimize the size of the Viterbi decoder available in the transceiver while at the same time keeping its decoding performance as high as possible. The parameters that were considered for the optimization were the number of bits used to represent the *channel state information* (CSI), the number of bits the *add-compare-select unit's* (ACSU) registers are extended in order to prevent overflow and the traceback depth of the decoder.

Taking into account the simulation flow introduced in figure 3 several System Generator models have been synthesized, one for each combination of parameters in the transceiver that are going to be analysed. The methodology employed has been as follows. First, an initial parameter configuration for the transceiver was chosen that at the same time could fit in the target FPGA and maximized the parameter values. For the Spartan-3A DSP 1800 used in this work, the maximum values for the CSI quantization, ACSU accumulator extension and traceback depth were 8, 7 and 60 respectively. Then the parameters were optimized one by one. When the influence of one parameter was being studied, the rest of the parameters were kept fixed to their initial value (or their optimum value if the corresponding analysis was already performed).

The architectures were tested under the highest rate available on the 802.11a standard using a multipath channel with a delay spread of 150ns. The signal to noise level was ranged from 18 to 30dBs and the frequency offset was randomly selected between ±120*ppm* with a 20MHz clock. Each SNR point was simulated 10^3 times. The same set of data was provided to all architectures under test.

Conclusions

The transceiver architecture performance has been thoroughly analysed after the optimum decoder parameters were obtained in the previous section. 6Mbps, 36Mbps and 54Mbps rates where simulated. The multipath channel had a random delay spread with a uniform distribution between 50 and 150ns. The frequency offset was also randomly selected from a uniform distribution between 0 and ±120*ppm* with a sampling clock speed of 20MHz. Each point in the PER curves of figure 4 was obtained after 10^5 simulations with PPDUs transmitting 2000 data bits.

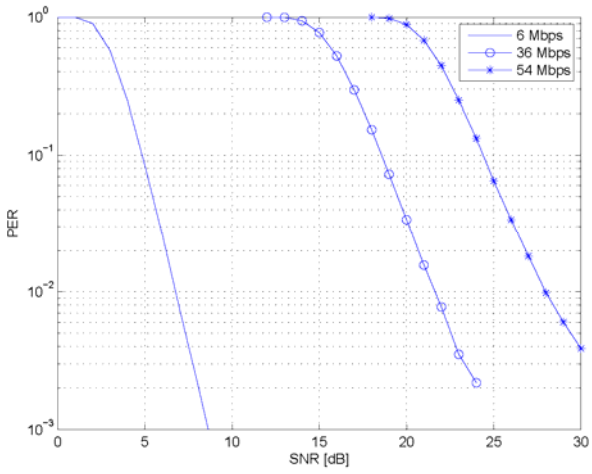

Fig. 4 - PER of the transceiver

The PER of the transceiver architecture presented in this paper is around 4dBs better than those of the works in [15] and [16].

The mean time to obtain a single PER point varied depending on the number of OFDM symbols that contained each PPDU. 6Mbps data rates required 80 minutes to obtain a PER point, 36Mbps data rates required 65 minutes and 54Mbps data rates required 60 minutes. In contrast, a single RTL simulation of the transceiver required around 50 seconds using a Core2-Quad processor at 2.5Ghz with 2GB of RAM. Therefore, the fast verification platform can reduce the simulation time in a factor of at least 10^3.

References

[1] G.E.Moore: Cramming More Components Onto Integrated Circuits (Proceeding of the IEEE, 1965)

[2] S.Niar, N.Inglart: Rapid Performance and Power Consumption Estimation Methods for Embedded System Design (Proceedings of the Seventeenth IEEE International Workshop on Rapid System Prototyping, 2006)

[3] S.Sjoholn, L.Lindh: The need for Co-Simulation in ASIC verification (Proceedings of the 23rd EUROMICRO Conference, 1997)

[4] D.Amos, A.Lesea, R.Richter: FPGA-Based Prototyping Methodology Manual (Synopsys 2011)

[5] J.Ou, V.K.Prasanna: MATLAB/Simulink Based Hardware/Software Co-Simulation for Designing Using FPGA Configured Soft Processors (Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium, 2005)

[6] A.Alimohannad, S.F.Fard, B.F.Cockburn: FPGA-based Accelerator for the Verification of Leading-Edge Wireless Systems (46th ACM/IEEE Design Automation Conference, 2009)

[7] J.Xing, S.Liu, W.Zhao: FPGA-Accelerated Real-time Colume Rendering for 3D Medical Image (3rd International Conference on Biomedical Engineering and Informatics, 2010)

[8] F.Jun, L.Shaobin: Application of FPGA to accelerate plasma FDTD Algorithm (3rd IEEE International Symposium on Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications, 2009)

[9] Y.A.Chapuis, L.Zhou, D.Casner, H.Ai, Y.Hervé: FPGA-in-the-Loop for Control Emulation of Distributed MEMS Simulation usign VHDL-AMS (First Workshop on Hardware and Software Implementation and Control of Distributed MEMS, 2010)

[10] Y.Guo, D.McCain: Compact Hardware Accelerator for Functional Verification and Rapid Prototyping of 4G Wireless Communication Systems (Conference Record of the Thirty-Eight Asilomar Conference on Signals, Systems and Computers, 2004)

[11] X.Ling, Z.Li, J.Hu, S.Wu: HW/SW Co-Simulation Platforms for VLSI Design (IEEE Asia Pacific Conference on Circuits and Systems, 2008)

[12] IEEE Standard 802.11a-1999, "Supplement to Information Technology—Telecomm. and Information Exchange between Systems - Local and Metropolitan Area Networks-Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: High Speed Physical Layer(PHY) in the 5 GHz Band"

[13] System Generator for DSP. User Guide

[14] V.Singh, A.Root, E.Hemphill, N.Shirazi, J.Hwang: Accelerating Bit Error Rate Testing Using a System Level Design Tool (Proceedings of the 11th Annual IEEE Symposium on Field-Programmable Custom Computing Machines, 2003)

[15] Y.Jung, S.Noh, H.Yoon, H.Jaeseok: Implementation of Wireless LAN Baseband Processor Based on Space-Frequency OFDM Transmit Diversity Scheme (IEEE Transactions on Consumer Electronics, 2005)

[16] A.Doufexi, S.Armour, M.Butler, A.Nix, D.Bull: A Study of the Performance of HIPERLAN/2 and IEEE 802.11a Physical Layers (2001 Vehicular Technology Conference)