A System Level Hardware/Software Partition of a MIMO-OFDM System for Systemc Modeling

Particionamiento hardware/software de un sistema MIMO-OFDM para modelamiento en Systemc

Recibido:25 de febrero de 2013
Aprobado:28 de febrero de 2013

Sebastian Eslava G.*, Carlos Sanchez** y Catalina Muñoz Morales***

Abstract

This paper presents the design process and description at the system level of a communication system based on Multiple Input-Multiple Output and Orthogonal Frequency Division Multiplexing (MIMO-OFDM). These are used in fourth generation (4G) systems, due to the performance improvement when facing rapidly changing wireless environments. Hardware/software partitioning is taken under consideration for the design, with developed criteria for measuring system performance. The design and validation of the system is made using *SystemC* language. This is part of a research work, carried out in order to study and establish an appropriate methodology for Hardware/Software co-design.

Keywords

SystemC, MIMO, OFDM, hardware/software partitioning.

Resumen

Este artículo presenta el proceso de diseño y descripción de un sistema de comunicaciones inalámbricas basado en tecnologías MIMO-OFDM (*Multiple Input- Multiple Output* y *orthogonal Frequency Division Multiplexing)*; estas tecnologías son ampliamente utilizadas

* Ph. D Profesor Asociado, Departamento de Ingeniería Eléctrica y Electrónica, Universidad Nacional de Colombia Bogota, Colombia, E-mail: jseslavag@unal.edu.co

** M. Sc. Profesor Asociado Departamento de Ingeniería Eléctrica y Electrónica, Universidad Nacional de Colombia, Bogota, Colombia. E-mail: cesanchezd@unal.edu.co

*** M. Sc. Ingeniera Electrónica, Universidad Nacional de Colombia Bogota, Colombia. E-mail: catmunozmor@unal.edu.co

en sistemas de comunicaciones móviles de cuarta generación (4G) ya que ayudan a mejorar el desempeño del sistema cuando este enfrenta un canal inalámbrico altamente variable. Para diseñar el sistema se consideró una metodología de particionamiento *hardware/software* y se desarrolló un criterio de medición de desempeño, obteniendo finalmente una descripción a nivel de sistema en lenguaje de *SystemC*. Este trabajo es parte de una investigación llevada a cabo con el objetivo de estudiar y establecer una metodología apropiada para el codiseño hardware/software.

Palabras clave

SystemC, MIMO, OFDM, particionamiento hardware/software.

I. Introducción

The difficulty present in data transmissions using the wireless environment has led to the need of increasing the performance of communication systems. This means, the development of new techniques for data transmission and reception, such as modulation and coding schemes, along with channel multiplexing techniques that increase data transmission rate in addition to reduce the error in the information sent.

Multiple Input and Multiple Output uses more than one transmit and receive antennas increasing data throughput without the necessity of additional bandwidth [2]. Since 1970, Orthogonal Frequency-Division Multiplexing is used to ensure high data rate on wireless transmission, by converting a fading channel of specific bandwidth into a number of orthogonal subchannels. MIMO and OFDM technologies combined with nearly perfect channel estimation, allow incrementing on the channel and system capacity, this means reliable and bandwidth efficient data transmission [1] [2].

Research in digital design has been focused in finding solutions for robust Hardware or Software implementations at higher abstraction levels such as: System-Level. However, in recent times, hardware and software co-design has been taken as an answer for increasing performance on electronic systems; the main task in this case arises when is time to specify which elements of the system that is being designed will be implemented using hardware and which ones using software; this important task is called Hardware and software partitioning (Hw/Sw partitioning). Previous works have shown the importance of having a clear understanding of the constraints presented for Hw/Sw partitioning; in [3] steps are presented for partitioning in SoC design, and in [4] optimization algorithms are used to find an optimal solution for co-designed systems. Other research work is focused using a System on Chip solution for systems [5], such as Wimax [6].

The structure of this paper is as follows: Section two presents the basic structure of a communication system with the before mentioned technologies, making an introduction to each of the processing techniques present in both MIMO and OFDM. Section three presents a methodology for hardware/Software partitioning, and the description of the communication system at a system level, using SystemC, and section fourth is conclusions and future.

II. System design

As an example for 4G communication systems, Wimax is a worldwide used IEEE standard for long range wireless networks, proposed in IEEE 802.16. Wimax considers the use of MIMO-OFDM on the physical layer (PHY), and therefore it will be the starting point for this work. Figure 1 shows the structure described above, and following, a description of the techniques used. [7].

The designed MIMO-OFDM system is organized in three stages for both transmission and reception systems: For the transmitter part, in the first stage the bit stream to be sent is arranged in order to preserve data integrity; additionally, there is a block for Forward Error Correction (FEC) that aims to control the amount of errors introduced in the transmission. Forward Error Correction techniques are integrated with digital modulation and interleaving schemes. In the second stage, the MIMO stage, spatial multiplexing techniques are used. Finally, at the third stage, OFDM is used for frequency multiplexing and pilot insertion for channel estimation at the receiver. For the receiver part, inverse operations are performed in order to decode data; for the MIMO decode, Zero Forcing (ZF) method is used. In the receiver`s first stage, the convolutional decoding stage, Viterbi algorithm is developed.

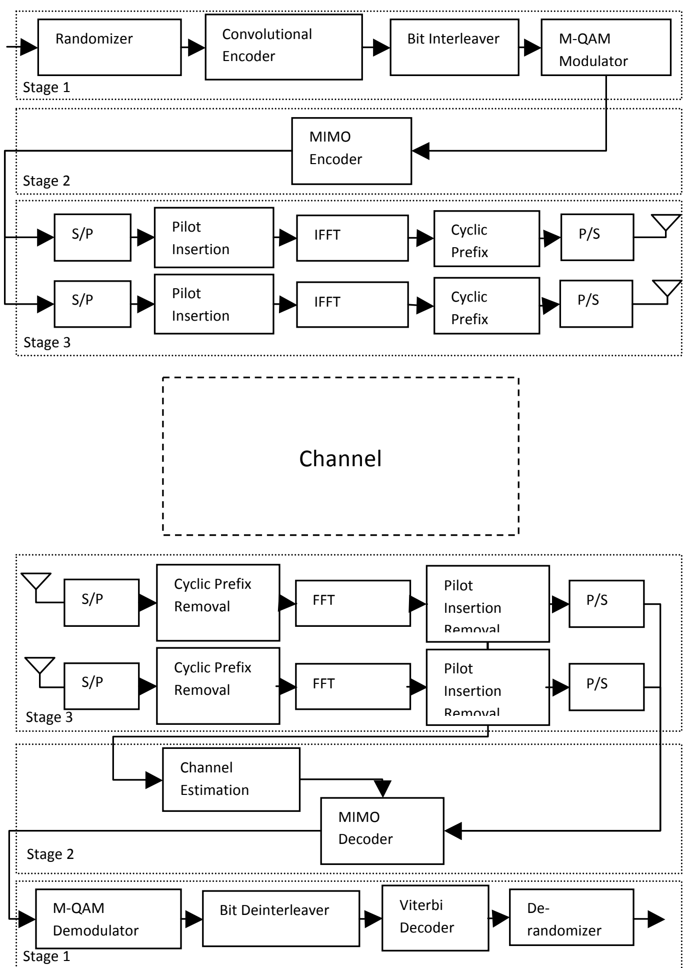


Figure 1. MIMO-OFDM System.

Forward error Correction (FEC): TrellisCoding

Forward error correction (FEC) is a channel coding technique used to control error in data transmission over noisy communication channels. Typically, a FEC encoder generates redundancy in the data so it is possible to obtain information about an error occurring in the stream to correct it. Redundancy added during the encoding is given by a determined type of code; depending on the complexity of that code, there are two main categories of codes: Block codes, such as Reed Solomon Codes [13] and Convolutional codes, such as Trellis Coded Modulation [14], which is the one used in for this paper. It can also be found a group of high performance codes, called turbo codes [15].

In convolutional coding, the current output signal depends not only on the current input, but also on a specific number of *n* inputs from a data stream stored in a shift register memory. A convolutional encoder is defined by three parameters:

1. Rate: is the ratio of the number of input bits to the number of output bits.
2. Constraint length: is the number of delays in de encoder.
3. Generator polynomial: in a polynomial, represents de connections among delay elements and the inputs

that form an output with module-2 adders. As an example, consider the following polynomial in (1) describing figure 2.

$$[171{,}133]_8 = [1111001{,}1011011]_2 \qquad (1)$$

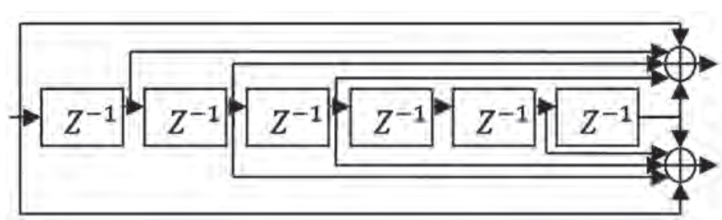


Figure 2. Polynomial diagram of a convolutional encoder.

The output from 7_8 adds (module-2 add) the current input, the previous input and the previous to the previous input to form $output_1$. In the same way, 5_8 adds the current input and the previous to the previous input to form $output_2$.

After input bits are encoded, a bit interleaver arranges the data bits in a way that if a loss of data or an error occurs during a time step, it does not affect the entire symbol sent. A bit interleaver is of great importance and it is usually integrated in the encoder, in order to add diversity [11] [12].

Modulation Schemes

Digital modulation has been made the technique for modern communication systems because it offers several advantages over analogue modulation techniques. Powerful error correction algorithms and higher data rates are some of the performance improvements, result of transmission and signal processing using information in the form of bit symbols. In MIMO-OFDM systems amplitude and phase modulation schemes such a QAM, QPSK and BPSK are generally used. In this case, we use M-QAM modulation with M equal 16,64 and 256.

MIMO: Multiple Input-Multiple Output

Is related to the use of multiple transmit and receive antennas as shown in Figure 3.a.Space-Time codes are used in MIMO scenarios in order to improve data reliability. Alamouti STBC (Space Time Block Code) [16] is an early, but commonly used technique, where two data symbols are Transmitted in two time laps according to figure 3.b.

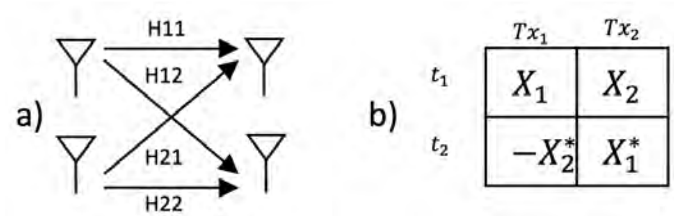


Figure 3. (a) 2 X 2 MIMO channel. (b) Alamouti Encoding scheme.

The Zero Forcing (ZF) MIMO is one of the decoding methods along with Minimum Mean Square Estimation (MMSE) and Maximum Likelihood (ML). ZF is used for its simplicity. Equation (2) demonstrates how sent data $\hat{x}$ is estimated based on the channel model H and received data Y for 2 transmit and 2 receive antennas.

$$\begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} = (H^H H)^{-1} H^H \begin{bmatrix} y_1^1 \\ y_2^1 \\ y^{*2}_{1} \\ y^{*2}_{2} \end{bmatrix} \qquad (2)$$

Where y_n^m represents the received data from antenna n at time slot m.

OFDM: Orthogonal Frequency Division Multiplexing is the ultimate data transmission scheme for multimedia communications. It utilizes multiple orthogonal carriers to accomplish spectral efficiency using a fixed bandwidth.

MIMO-OFDM communication systems can show a good performances if there is information about the channel state, usually obtained by adding a series of pilot symbols arranged in a specific periodical structure on the OFDM symbol, which is made of pilot symbols, a DC sub-carrier and a guard band.

In this stage, Cyclic Prefix (CP) extension is added at the front part of the transmission. Figure 4 shows the structure of the OFDM symbol. For this work we developed two lengths of cyclic prefix, 1/41/4 which is used for fixed Wimax and 1/81/8 for mobile Wimax. Table I specifies OFDM transmission for this work [8].

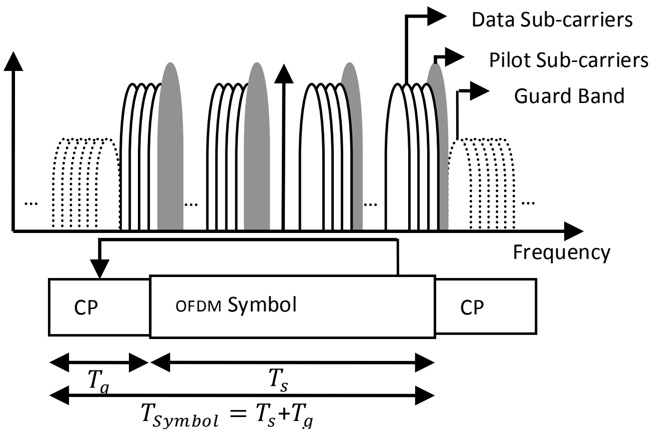


Figure 4. OFDM Symbol Structure.

	Channel Bandwidth	Number of Sub-Carriers	Data Sub-Carriers	Pilot Sub-Carriers	Guard Sub-Carriers
Fixed	2,5 MHz	256	192	8	56
Mobile	5,0 MHz	512	360	60	92

Table 1. OFDM modulation parameters for fixed and mobile Wimax.Viterbi Algorith

When data transmission is performed in a highly noisy environment, it becomes necessary to use identification and error correction techniques; some decoding algorithms have been developed such as [17]. The Viterbi algorithm [18], despite the high use of processing resources it is the most convolutional decoding algorithm [19], allows decoding convolutional codes since it recovers the originally transmitted data based on the received noisy bit stream.

The Viterbi encoder uses the convolutional codes diagram of a state machine to find the data path with the lowest accumulated error value which can be derived from the hamming distance measure between the possibly received data and the actually received one and finally go backwards through translating the bit stream to obtain the bits coded at the transmitter.

III. Hardware/Software partitioning for systemc modeling

The Hardware/Software Partitioning Problem:

Digital communication systems have been developed using DSPs or ASICs based architectures, so they are not able to satisfy design requirements such as flexibility in the ASICs case and performance in the DSP case. Different challenges are presented when integrating Hardware and Software design and modelling for specific purpose systems; more precisely, high processing speed, low developing cost and modularity. These are some of the desirable characteristics in communication systems; where there is a generalized need to provide high transmission speed and the possibility to integrate new processing techniques. Designers companies propose the use of single architecture with FPGAs, DSPs and CPUs components inside a dedicated chip [9] [10].

In order to find an appropriate Hardware/software structure, the following methodology was used: First task graph was made for observing which processes could be performed concurrently and which ones serial; according to the task graph, an algorithm was developed to simulate the required functions for performance profiling; finally, a series of decisions about system structure were made.

Figure 5 shows nine main parallel functions: Convolutional coding (F_t) and Viterbi decoding (F_v), QAM modulation (F_q) and demodulation (F_d), MIMO space-Time coding (F_a) and Zero Forcing estimation (F_z), OFDM modulation (F_{om}) and demodulation (F_{od}), and channel simulation (F_c), which in this case is used only to validate the communication system according to the coding and processing functions of MIMO-OFDM, for which most essential and robust processing elements are Fast Fourier Transform necessary for OFDM, and data coding on the FEC block. Looking into these functions other more specific tasks are found.

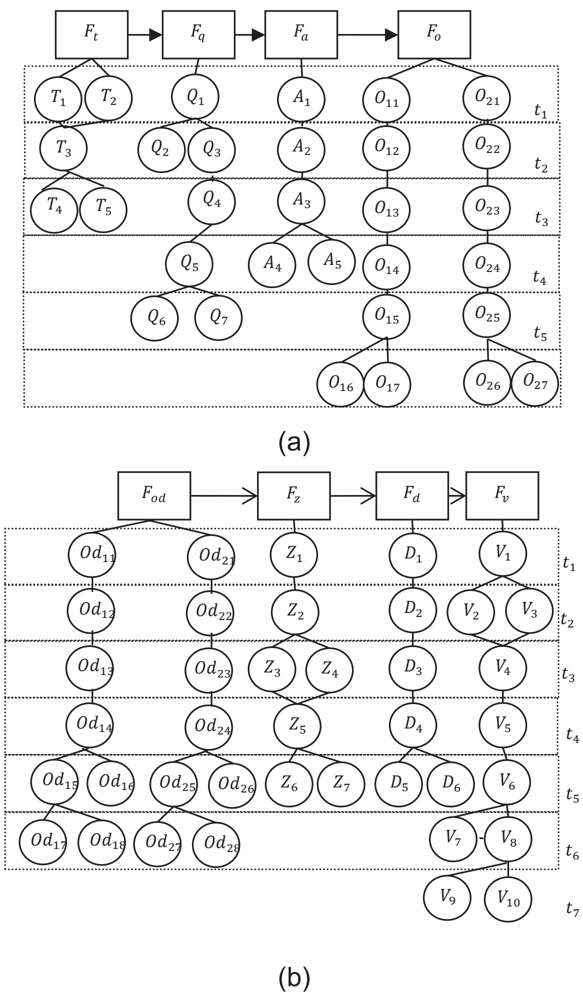


Figure 5. Task graph for MIMO-OFDM architecture, (a) transmitter, (b) receiver

Table II shows the description of each task specified in figure 5, according to the time slots t_n where it should be executed. $F_t, F_V, F_q, F_d, F_a, F_z, F_{om}, F_{od}$ And F_c, are the nine main functions; for the transmitter part, F_t (Randomization, Convolutional Coding and interleaving), and F_q (QAM modulation) for stage 1, F_a (MIMO Alamouti encoding) for stage 2 and

F_o (Pilot insertion, Fast Fourier Transform and cyclic prefix addition for both antenna 1 and antenna 2) for stage 3 (see figure 1). For the receiver part, F_{od} (cyclic prefix removal, Inverse Fast Fourier Transform, pilot removal) for stage 3, F_z (Zero Forcing Estimation) for stage 2 and finally, F_d (QAM demodulation) and F_v (de-interleaving, Viterbi decoding and de-randomizer) for stage 1.

T_n, Q_n, A_n, O_{1n}, O_{2n} represent the explicit task for each function.

Task	Function
T_1, Q_1, A_1, Om_{11}, Om_{21}, Od_{11}, Od_{21}, Z_1, D_1, V_1	Memory request and vector initialization for Functions.
T_2	Conversion of generator polynomial
T_3	Bit shifting and module-2 adding for data coding
T_4, Q_6, A_4, Om_{16}, Om_{26}, Od_{17}, Od_{27}, Z_6, D_5	Data storage in function channel
T_5, Q_7, A_5, O_{17}, Om_{27}, Od_{18}, Od_{28}, Z_7, D_6, V_{10}	Memory release
Q_2	Constellation matrix initialization
Q_3	Bit clustering by symbol size
Q_4	Binary to Decimal conversion
Q_5	Modulated data arrangement
A_2	Conjugate numbers and data arrangement for space coding
A_3	Data storage for antenna 1 and 2
Om_{12}, Om_{22}	Pilot symbol insertion in data stream for antenna 1 and 2
Om_{13}, Om_{23}	IFFT plan creation
Om_{14}, O_{m24}	IFFT operation executed
Om_{15}, Om_{25}	Cyclic prefix adding
Om_{18}, Om_{28}	IFFT operation destruction
Od_{12}, Od_{22}	Cyclic prefix removal
Od_{13}, Od_{23}	FFT plan creation for antenna 1 and 2
Od_{14}, Od_{24}	FFT operation execution for antenna 1 and 2
Od_{15}, Od_{25}	Pilot symbol removal and storage
Od_{16}, Od_{26}	FFT plan destruction
Z_2	Arrangement of data from antenna 1 and 2
Z_3	Inverse and hermitian matrix operations for channel weight

Esta tabla continúa en la siguiente página ⟶

Task	Function
Z_4	Perform conjugate operations for received data
Z_5	Hard decision decoding
D_2	Constellation initialization
D_3	Imaginary and real data demodulation
D_4	Decimal to binary conversion
V_2	Data arrangement for de-interleaving
V_3	State machine and possible output table initialization
V_4	Bitwise hard decision decoding (Hamming distance metric)
V_5	Accumulated and path error metric storage for all possible states across input bit stream
V_6	State number selection according to smallest accumulated error metric
V_7	Error counting
V_8	State path selection and decoding f input
V_9	De-randomization operation

Table 2. Task graph functions.

The second step was the profiling, where processing time and Multiply-Accumulate (MA) operations have been taken under consideration, in order to establish performance for the main functions in each system modules. These values were obtained using the Visual Studio profiling tool. The performance test was completed for fixed values, of data stream size, FFT size (N), coding rate (R) and modulation scheme (M), chosen as an example for transmission in the Wimax standard. Results are presented in table III. The functions that most MA operations performed were Convolutional coding, Viterbi decoder, and the constellation map construction for QAM modulation (Constellation Alphabet Construction, binary to decimal conversion). Highest execution time was showed by IFFT and FFT operations created using *fftw* library because of the creation plan necessary for operation executing.

Function	Multiply–accumulate operation
Trellis Coding	7507
Interleaver	685
Constellation Construction	1404
Binary to decimal conversion	1464
Modulation	288
Alamouti encoding	864
Conjugate op.	1
Pilot Insertion	950
Ifft	Fma* = 128
CP Adding	768
CP Removal	768
Fft	Fma* = 128
Pilot removal	600

Esta tabla continúa en la siguiente página —>

Function	Multiply–accumulate operation
Zero Forcing Weight arrangement	34
MIMO hard decision decoding	384
Decimal to Binary conversion	1156
Demodulation	1538
Viterbi algorithm	103296
De-interlaver	6349
Randomization	5760

Table 3. Performance statistics.
For Fixed data stream size = 384, R = ½, M = 16, N = 256

FMA = floating point multiply-accumulate operations.

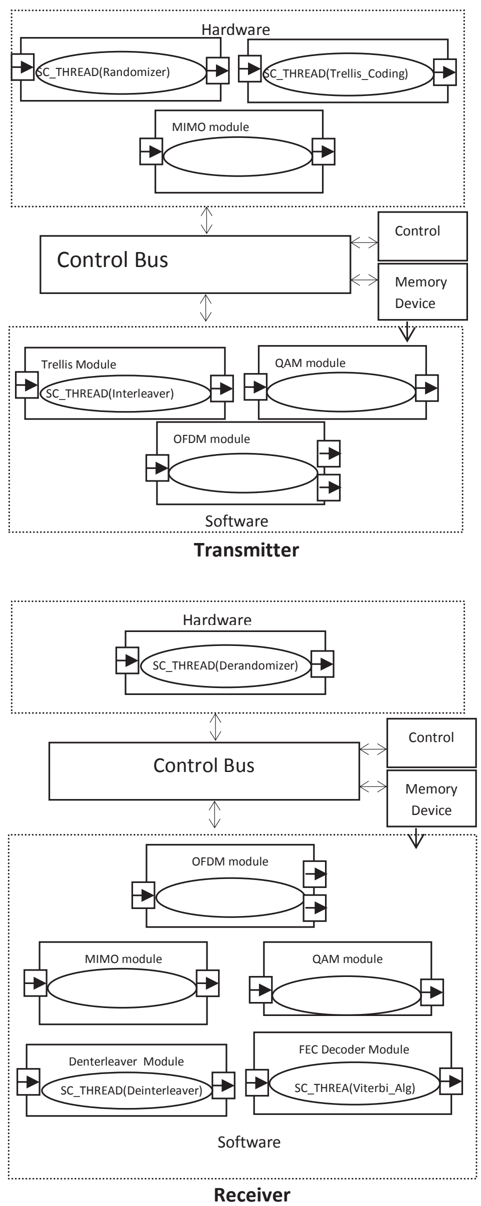


Figure 5. SystemC design of MIMO-OFDM system.

On the third step, with the collected information, a partitioning was proposed together with a reconstruction of the developed algorithms, in order to increment system performance:

1. Memory request and release will be performed only at the beginning and the end of the entire process.
2. Bit shifting and module two adding for trellis coding function will be performed by hardware, since it shows lower processing time and ease of development as well as Alamouti coder and randomizer.
3. Bit interleaver will be performed by a mathematic function with only bit rearrangement operations.
4. Constellation matrix will be stored permanently in a memory block in order to decrease processing time.
5. FFT function from OFDM module will be performed with *fftw* library since it shows the fastest processing among other algorithms using floating point arithmetic.
6. Modulation, demodulation, MIMO decoding and Viterbi algorithm will be performed by hardware.

SystemC Description:

The MIMO-OFDM communication transmitter and receiver have been constructed using SystemC module classes, using fifos to interconnect each module to the next one. A sc_fifo is a SystemC primitive channel, found as an object in sc_fifo class. Figure 5 shows the connection between processing modules that where described according to the partitioning problem results.

In order to validate transmission and reception system behaviour we simulated a Rayleigh fading wireless channel, as equation (3) shows, and generated a series of digital signals for modulation and coding processing.

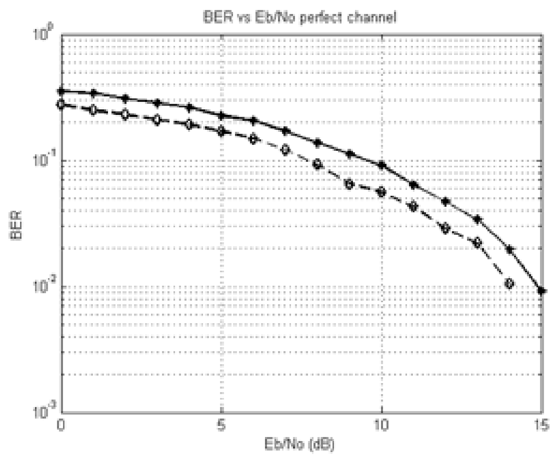
$$\boldsymbol{Y} = \boldsymbol{H} * \boldsymbol{X} + \boldsymbol{N} \quad (3)$$

Where $\boldsymbol{H}$ represents the MIMO channel gains, $\boldsymbol{X}$ is the transmitted signal, $\boldsymbol{Y}$ the received signal and $\boldsymbol{N}$ is additive Gaussian noise.

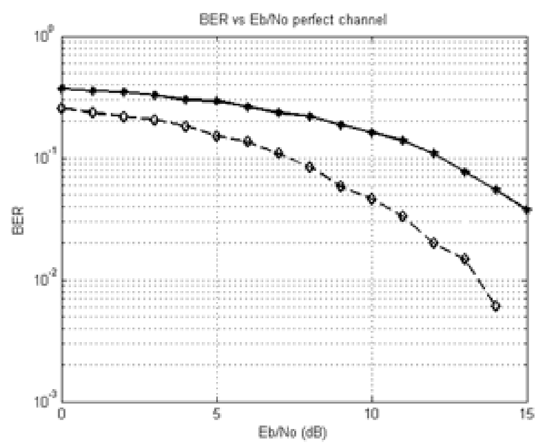
Figure 6 shows the transmitted signal degradation when passing through the wireless channel; Bit Error Rate increases according to Eb/No (energy per bit to noise power spectral density ratio) for modulation schemes 16-QAM, 64-QAM and 256-QAM. In the figure, the dotted line represents performance assuming perfect channel knowledge and the solid line represents system performance for estimated channel.

system performances shows that degradation increases when the M value of M-QAM modulation increases, which means that even tough 256-QAM modulations presents a higher transmission rate, it is more sensitive to noise added to the signal. This information is useful for link adaptation techniques.

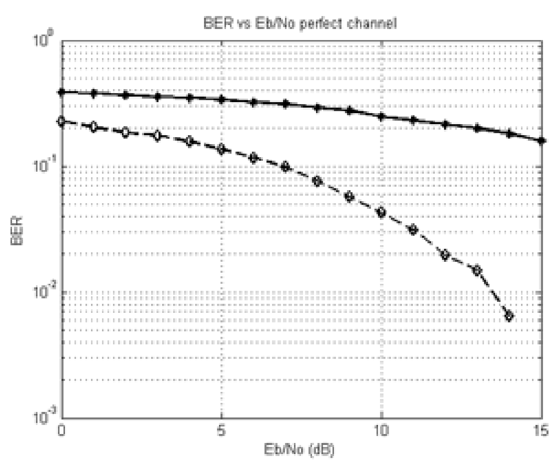
When errors in transmitted bits were found, the Viterbi algorithm developed was able to reconstruct the transmitted signal if SNR was lower than 15dB.



(a)



(b)



(c)

Figure 6. BER vs Eb/No. (a) 16-QAM, (b) 64-QAM, (c) 256-QAM

IV. Conclusions and future work

In this paper was presented the System Level Hardware-Software partition of a MIMO-OFDM transmission and reception system for SystemC modelling. We presented a strategy for establishing a solution for the Hardware/Software partitioning problem, which starts with analysing the structure of the system and the tasks needed to perform transmission and reception of binary signals and the level of concurrency of these tasks. The SystemC description is important for the simulation of wireless communication systems techniques, such as modulation and coding schemes testing for link adaptation and channel prediction.

Hardware/software partitioning is one of the most important tasks at the early stage of system modelling, and incorporating SystemC in modelling of wireless communication systems is an appropriate design methodology since it allows validating different technologies from a single platform. The required functions of the system under design were considered for analysing performance, using Multiply-Accumulate operations and concurrency between processes in order to establish an appropriate Hardware/Software partitioning.

Results validate system behaviour, as developed according to the Hardware/Software partitioning problem and the solution considered in this work, and serve as a background for the application of link adaptation techniques in order to find an appropriate relationship between transmission rate and Bit Error Rate to increase system performance.

Future work is framed on the construction of MIMO-OFDM communication system with link adaptation techniques, for testing channel prediction algorithms.

References

[1] Hongwei Yang, Ltd, «A Road to Future Broadband Wireless Access: MIMO-OFDM-Based Air Interface», Vol. 43, 2005,Issue 1, *Communications Magazine,* IEEE, pp. 53-60.

[2] Sampath, H., Talwar, S., Tellado, J., Erceg, V. and Paulraj, A., «A Fourth Generation MIMO-OFDM Broadband Wireless System: Design, Performance and Field Trial Results», Vol. 40 2002, Issue 9, *Communications Magazine*, IEEE, pp. 143-149.

[3] Peter Arato, Sandor Juhasz, 201th Adam Mann, Andras Orban, David Papp, Hardware-software partitioning in embedded system design, Intelligent Signal Processing, 2003 IEEE International Symposium on, pp. 107-202, 2003.

[4] Raj, S. Mitra and Anupam Basu, «Hardware/Sowftarw Partitioning: a Case of Constraint satisfaction», *Intelligent Systems and their Applications*, IEEE , Vol. 15, 2000, Issue 1, pp. 54-63.

[5] Dongwan Shin, Andreas Gerstlauer, Rainer Dömer and Daniel D. Gajski, «System-on-Chip Communication Modeling Style Guide», Technical Report CECS-TR-03-45, Center for Embedded Computer Systems, University of California, Irvine,2004.

[6] Chuang, G.C.H. «A MIMO WiMAX SoC in 90nm CMOS for 300km/h mobility" Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2011, pp. 134-136.

[7] Lajos Hanzo, Yosef (Jos) Akhtman and Li Wang, «MIMO-OFDM for LTE, Wi-Fi and Wimax: Coherent versus Non-coherent and Cooperative Turbo Transceivers», John Wiley & Sons Ltd, 2010.

[8] Yong Soo Cho, Jaekwon Kim, Won Young Yang and Chung G. Kang, «MIMO-OFDM Wireless Communications with MATLAB®», John Wiley & Sons, 2011.

[9] Intel, «WiMAX Base Station Proof of Concept», 2009.

[10] Fujitsu Microelectronics America, Inc, «The Fujitsu WiMAX 802.16-2004 SoC Programming Guide», 2005.

[11] Yuan Li, Ying-Chang Liang, Sumei Sun and Rui Zhang, «Adaptive Trellis and Bit-Interleaved Codedmodulation for Ordered MIMO-OFDM Channels», IEEE 16th International Symposium on Personal, Indoor and Mobile Radio Communications, pp. 226-230, 2005.

[12] Xiaodong Li, and James A. Ritcey, «Trellis-Coded Modulation with Bit Interleaving and Iterative Decoding», IEEE Journal On Selected Areas In Communications, vol. 17, 1999, n.° 4, pp. 715-724.

[13] I. S. Reed Solomon, «Polynomial Codes Over Certain Finite Fields», *Journal of the Society for Industrial and Applied Mathematics*, Vol. 8, 1960, n.° 2.

[14] G. D. Forney, Jr., «Convolutional codes I: Algebraic structure», *IEEE Trans. Information Theory*, Vol. IT-16, 1970, pp. 720-738.

[15] C. Berrou, A. Glavieux and P. Thitimajshima. «Near Shannon limit error- correcting coding and decoding: Turbo codes», Proc. 1993 Int. Conf. Comm., 1993, pp. 1064-1070.

[16] S. Alamouti, «Space block coding: A simple transmitter diversity technique for wireless communications», *IEEE J. Select. Areas Commun*, Vol. 16, pp. 1451-1458, Oct. 1998.

[17] R. M. Fano, «A Heuristic Discussion of Probabilistic Decoding», *IEEE Trans. Inf. Theory*, Vol. 9, n.° 2, pp. 64-74, Apr. 1963.

[18] Viterbi, A. «Error Bounds For Convolutional Codes and an Asymptotically optimum Decoding Algorithm», *IEEE Trans. Inf. Theory*, Vol. 13, Apr. 1967, Issue 2, pp. 260-269.

[19] G. David Forney, Jr. «The Viterbi Algorithm: A Personal History», Viterbi Conference, University of Southern California, Los Angeles, March 8, 2005.