INTEGRATION, the VLSI journal 49 (2015) 49–64

Contents lists available at ScienceDirect

INTEGRATION, the VLSI journal

journal homepage: www.elsevier.com/locate/vlsi

Modeling memory effects in RF power amplifiers applied to a digital pre-distortion algorithm and emulated on a DSP-FPGA board

José Ricardo Cárdenas-Valdez [a], José Cruz Núñez-Pérez [a], José Alejandro Galaviz-Aguilar [a], Andrés Calvillo-Téllez [a], Christian Gontrand [b], J. Apolinar Reynoso-Hernández [c], Esteban Tlelo-Cuautle [d,*]

[a] Digital Technology Research and Development Center, National Polytechnic Institute (CITEDI-IPN), Av. del Parque 1310, Mesa de Otay, 22150 Tijuana, Baja California, Mexico
[b] Institut des Nanotechnologies de Lyon (INL), UMR 5270 CNRS, Institut National des Sciences Appliquées de Lyon (INSA-Lyon), Bâtiment Blaise Pascal 7, Avenue Jean Capelle 69621, Villeurbanne Cedex, France
[c] Center for Scientific Research and Higher Education of Ensenada (CICESE), Ensenada, Mexico
[d] National Institute of Astrophysics, Optics and Electronics (INAOE), Luis Enrique Erro No. 1, Santa María Tonatzintla, San Andrés Cholula, 72840 Puebla, Mexico

ARTICLE INFO

Article history:
Received 5 March 2014
Received in revised form
16 August 2014
Accepted 16 December 2014
Available online 25 December 2014

Keywords:
Amplitude modulation
Field programmable gate arrays
Memory modeling
Nonlinear circuits
Power amplifier

ABSTRACT

An emulation tool with the capability of modeling the nonlinearity order and memory effects for real power amplifier's (PAs) conversion curves, is introduced. The proposed tool comprises special cases of Volterra series as the Memory Polynomial Model and a learning technique like artificial neural networks. The proposed system is a novel integrated one able to model the real behavior of radio frequency (RF)-PAs under different memory models. The developed system starts from a test bed able to acquire AM–AM and AM–PM distortion curves measurements. This procedure can predict the behavior and improve the analysis using it as a design tool.

1. Introduction

With the explosive growth of the high data-rate transmission, the radio frequency (RF) Spectrum is becoming an increasingly scarce resource. Such scarcity is due to existing licensing regiment. Additionally, there is an increasing concern that higher performing requirements drives to use faster frequencies for digitally modulated signals, such as in wideband wireless communication systems, e.g. wideband code division multiple access (WCDMA), Worldwide interoperability for microwave access (WiMAX), etc. [1]. In those systems, the transmitter chain includes a power amplifier (PA) as the main device. Unfortunately, its inherent nonlinearity causes in-band distortion and intermodulation products that spreads out over neighbor channels. That way, to mitigate undesired intermodulation and distortion effects, recent advances related to linearization techniques have been developed, mainly to compensate memory effects induced by real PAs [2,3]. These issues have been partially solved introducing modeling techniques for PAs, such as: memory or memoryless models [4,5], polynomial models [6], neural networks [7,8], others techniques taking into account memory effects based on Volterra series [9–13], etc., all of them providing a proper way to capture the nonlinearity order and memory depth.

Important efforts related to PAs linearization improving memory effects have been developed in [14–19]. Other works make use of commercially available design tools, e.g. DSP Builder [20–23], exhibiting a proper way to model nonlinear behaviors. The existing research opens a bridge between the modeling through software and emulation in field programmable gate arrays (FPGAs) [24,25], which flexible characteristics give a precise estimation of real PAs behavior. In this manner, this work introduces a modeling approach based on a special case of Volterra series as memory polynomial model (MPM), which is successfully compared with a learning technique, as artificial neural networks (ANNs). The system has the flexibility to model a real PA behavior and a digital pre-distortion (DPD) module implementation that are fully controlled by Matlab.

* Corresponding author. Tel.: +52 222 266 3100; fax: +52 222 247 0517.
E-mail addresses: jcardenas@citedi.mx (J. Ricardo Cárdenas-Valdez), nunez@citedi.mx (J. Cruz Núñez-Pérez), jgalaviz@citedi.mx (J. Alejandro Galaviz-Aguilar), calvillo@citedi.mx (A. Calvillo-Téllez), christian.gontrand@insa-lyon.fr (C. Gontrand), apolinar@cicese.mx (J. Apolinar Reynoso-Hernández), etlelo@inaoep.mx (E. Tlelo-Cuautle).

http://dx.doi.org/10.1016/j.vlsi.2014.12.005
0167-9260/

The first trial is performed from measurements of three real PAs, making unnecessary its physical presence as device under test (DUT), as well as the transceiver in a DPD chain. The proposed modeling approach is based on a test bed able to achieve the AM–AM and AM–PM conversion curves [26]. The main contribution is the ability to model memory effects and nonlinearity order previous to a DPD chain proving the performance of a DPD algorithm.

The paper is organized as follows: Section 2 describes the general behavioral modeling theory by using Volterra series and special truncations. Section 3 describes the test bed to obtain RF-PAs conversion curves, the MPM, ANN and DPD module implementations. Section 4 presents the main experimental results. Finally, Section 5 summarizes the conclusions.

2. Behavioral modeling of nonlinear RF power amplifiers

Nonlinearity is an inherent property of a PA in wideband applications. This undesirable effect means that the output response is stimulated not only by the actual state of the input signal but also by its previous states. Volterra series provide a precise way to analyze this behavior taking into account the undesirable effects as memory. However, the number of parameters to be estimated increases exponentially with the degree of nonlinearity and memory depth, creating high computational complexity that is impractical in some real applications. Therefore, MPM performs an effective truncation of Volterra series and can be used to mitigate such complexity. That way, special truncation of Volterra series as the MPM can be projected to determine the input and output ratio. The general representation of Volterra series is given by,

$$y(n)=\sum_{m_1=0}^{M}h_1(m)x(n-m)+\cdots+\sum_{m_1=0}^{M}\sum_{m_2=0}^{m}h_2(m_1,m_2)x(n-m_1)x(n-m_2)$$
$$+\cdots+\sum_{m_1=0}^{M}\sum_{m_2=0}^{M}\sum_{m_3=0}^{M}h_2(m_1,m_2)x(n-m_1)x(n-m_2)x(n-m_3) \tag{1}$$

where $y(n)$ is the output signal of the PA, $x(n-m_i)$ is the different moment of the input signal, $h_k(m_1,\ldots,m_k)$ is the k_{th}-order of the Volterra kernel, K is the nonlinear order of the model and M is the memory depth [27].

2.1. Memory polynomial model

The equivalent baseband PA model, considering memory effects and nonlinearity order, can be represented with a MPM considering only the diagonal terms in the Volterra series, by

$$y(n)=\sum_{q=0}^{Q}\sum_{k=1}^{K}a_{2k-1}|x(n-q)|^{2(k-1)}x(n-q) \tag{2}$$

where $x(n)$ is the input complex baseband signal, $y(n)$ is the output complex baseband signal, $a_{k,q}$ are complex valued parameters or MPM coefficients, Q is the memory depth and K is the polynomial order.

Eq. (2) can be rewritten in terms of the memory depth to obtain:

$$y(n)=\sum_{q=0}^{Q}F_q(n-q)=F_0(n)+\mathrm{F}_1(\mathrm{n}-1)+\ldots+\ F_{\mathrm{q}}(\mathrm{n}-\mathrm{Q}) \tag{3}$$

where $F_q(n)$ can be expressed by (4) or by (5),

$$y(n)=\sum_{k=1}^{K}a_{2k-1}|x(n-q)|^{2(k-1)}x(n-q) \tag{4}$$

$$F_q(n)=a_{1,q}x(n)+a_{3,q}\ |x(n)|^{\ 2}x(n)+\cdots+\ a_{2k-1}\ |x(n)|^{\ 2(k-1)}x(n) \tag{5}$$

Each MPM stage can be subdivided in terms of the sampling time. Fig. 1 shows the internal structure and the delay time for each step; for additional memory analysis is necessary to add other layers in order to represent the general structure of MPM as blocks.

The least square error method (LSM) is used to find the coefficients from a given analytical function and a set of measurement data. The procedure of LSM is explained in [28,29] and internal Matlab functions are included during the modeling stage.

2.2. Artificial neural network

ANN is another way to compute the information based on a biological model. It is quite useful for fitting processes pattern recognition and classification. In general, a neuron consists of a system with various inputs and a single output. Fig. 2 shows the constitution of an artificial neuron. An ANN is composed by: a set of inputs $x_j(t)$ and synaptic weights w_{ij}, and a propagation law as depicted by,

$$h_i(\mathrm{t})=\sigma(w_{ij},x_j(t));h_i(t)=\sum w_{ij}x_j \tag{6}$$

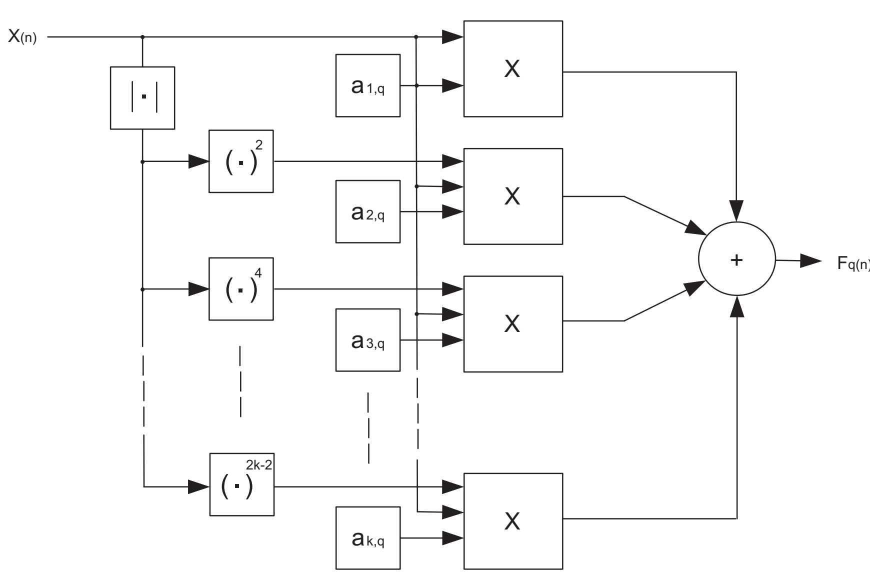

Fig. 1. Implementation of F_q with Volterra kernels.