Digital Predistortion Circuit using Vivado HLS

PES University, Ring Road Campus, e-mail:- PES1UG23EC266@pesu.pes.edu , PES1UG23EC274@pesu.pes.edu , PES1UG23EC280@pesu.pes.edu , PES1UG23EC276@pesu.pes.edu

Abstract:-

The increasing demand for energy efficiency in wireless communication systems makes power consumption a critical design metric for power amplifiers (PAs) and their associated linearization techniques. Non-linearity in PAs leads to significant spectral regrowth and distortion, which degrades the quality of the output signal. Digital Predistortion (DPD) circuits address these issues by applying inverse distortion. However, optimizing DPD hardware for power, performance, and area (PPA) trade-offs through traditional Register-Transfer Level (RTL) design is a labor-intensive and iterative process, regardless of the DPD method used (e.g., memory polynomial, lookup tables, neural networks).

This paper presents an efficient and rapid implementation of a Digital Predistortion (DPD) circuit using FPGA technology, specifically leveraging Vivado High-Level Synthesis (HLS). The core innovation of this work is a learning algorithm designed to generate inverse distortion based on real-time feedback from the Power Amplifier (PA) with high accuracy. The circuit features a modular architecture, which is systematically managed by a top-level function, allowing for precise correction of PA distortions that are accurately modeled using the Saleh Model. Our comprehensive signal processing chain takes raw digital input, applies predistortion, converts the signal to analog, up-converts it to radio frequency (RF), and amplifies it. A critical feedback loop captures the distorted output, down-converts it, and digitizes the signal. This setup enables continuous adaptation and optimization of the DPD coefficients, ensuring optimal linearity..

The system's outputs, which include final I/Q values, linear gain, and gain measured in dB, clearly demonstrate that input signals attain a high level of linear amplification after digital pre-distortion. This work emphasizes the effectiveness of HLS in speeding up the development of advanced digital pre-distortion (DPD) systems. As a result, we can expect improved signal quality and performance in future wireless devices, while also simplifying the hardware design process significantly.

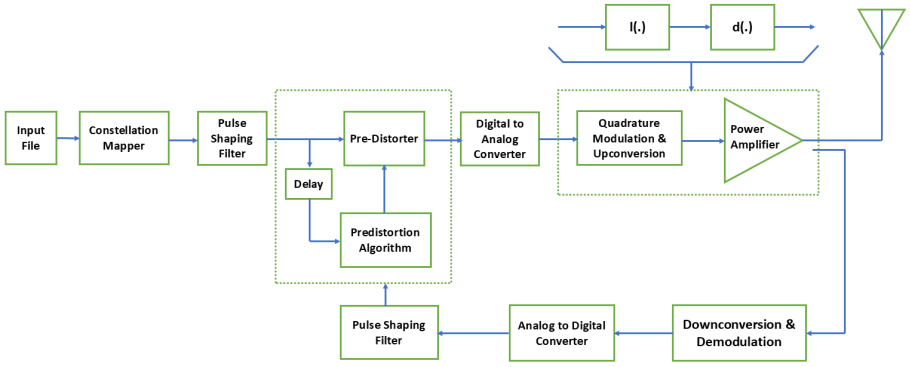
Introduction:-

Digital Predistortion (DPD) is a widely used technique for linearizing power amplifiers in wireless communication systems. This method is particularly appealing due to its digital implementation, which can be applied in both mobile units and base stations. At the heart of this approach is a look-up table (LUT) that is strategically placed before the amplifier. The LUT modifies the input signal by applying the inverse characteristics of the amplifier, ensuring that the overall system—comprising the predistorter and the amplifier—operates linearly. The main goal of using this circuit is to reduce the inherent nonlinearities associated with power amplifiers. Traditionally, efforts to linearize power amplifiers have focused on minimizing spectral growth, primarily driven by regulatory requirements aimed at limiting out-of-band emissions.

The objective of this paper is to simulate a Digital Predistortion Circuit using Vivado High-Level Synthesis (HLS). This will be achieved by developing a top-level function that integrates and organises all the various models within the circuit. The circuit contains Models such as Constellation Mapper, Pulse Shaping Filter, Predistorter (Predistortion Algorithm), Digital-to-Analog Converter(DAC), Analog-to-Digital Converter(ADC), Quadrature Modulation and Upconversion (QM), Power Amplifier(PA) and Downconversion and Demodulation(DDC). In our project, we utilise high-level programming languages, such as C++, to design each model within a centralised top-level function. This approach enhances code organisation and readability, allowing for efficient model management. The content is organized into four sections. Section I offers an overview of the circuit. Section II discusses the algorithms utilized and provides an explanation of the DPD module. Section III confidently presents the results of the study, accompanied by clear explanations of the graphs that illustrate these findings and Section IV covers potential applications and conclusion.

Section I: Overview of the Circuit

Fig.1 illustrates the block diagram of the Digital Predistortion Circuit, showcasing its essential components and functionality.



Digital Predistortion Circuit

Fig 1. The Block Diagram of the pre-distortion circuit is followed. The diagram is modelled after K. J. Muhonen, M. Kavehrad and R. Krishnamoorthy, "Look-up table techniques for adaptive digital predistortion: a development and comparison," in IEEE Transactions on Vehicular Technology, vol. 49, no. 5, pp. 1995-2002, Sept. 2000, doi: 10.1109/25.892601.

Let's begin with the constellation mapper, where the inputs are represented using Quadrature Phase Shift Keying (QPSK). Both the I (In-phase) and Q (Quadrature) components are mapped to their respective values. Specifically, '0' is mapped to -0.7071, while '1' is mapped to 0.7071Ref.[6]. The outputs from the Constellation mapper are processed by the Pulse Shaping Filter (PSF), where they are filtered and adjusted using the Raised Cosine Filter (RCF), ensuring that the values remain within the appropriate range.

In the initial epoch, predistortion is not used, and input signals are sent directly to the power amplifiers (PAs). This allows the amplifiers to operate based on the original signal characteristics, which can lead to nonlinearities if the PAs are not optimally configured. Skipping predistortion at this stage enables analysis of the amplifiers' raw performance before applying corrective measures later. Starting from the second epoch, predistortion is applied to reduce errors and achieve ideal outputs according to the RLS method. Next, the signal undergoes conversion from discrete digital values to discrete analog values. The discrete analog format enables efficient implementation of Quadrature Amplitude Modulation for the input signal. The discrete analog form facilitates the operation of the Quadrature Amplitude Modulator (QAM). QAM involves local oscillator-generated sine and cosine values stored in an LUT, which are then applied to the input signal. The upconverter modulates real signal values, which then serve as inputs to the PA, where it was modeled after a Saleh amplifier. The next output from PA is processed by a pulse shape filter through the feedback loop, and is also sent for transmission. After that, the amplified (non-linear) output is first demodulated and then downconverted, and this signal is passed to the DPD module. The DPD module aims to reduce the error function by comparing previous outputs and inputs while recursively calculating z(n). This adaptation model is based on the Recursive Least Squares method (RLS). Therefore, the predistortion circuit significantly reduces distortion, enabling the achievement of a precise and linear gain, which enhances overall system performance. As observed in Fig. 1, the output of the power amplifier exhibits a non-linear characteristic function, represented by d(.). The term I(.) refers to the section of the analog linear circuit of the transmitter. The overall implementation of the predistortion circuit is carried out using Vivado HLS. This project is organized into a main function along with several sub-functions that perform specific tasks based on the requirements of the models and testbench. The main function calls the other sub-functions, one of which implements the predistortion algorithm. An explanation of this algorithm will be provided in the next section.

Section II: Digital Pre-Distortion Module:-

This section discusses the design of the Digital Pre-Distortion (DPD) module, which utilizes the Recursive Least Squares (RLS) method. The DPD module serves as the central component of the Pre-Distortion Circuit, facilitating linear gain within the system. It comprises three key components: the Pre-Distortion block, the DPD Algorithm block, and a delay element. The RLS algorithm is employed to compare and generate the inverse distortion. The model of orthogonal polynomial based DPD is expressed in an equation below:-

$$z(n) = \sum_{m=1}^{K} \sum_{q=0}^{Q} w_{mq}\ \varphi_m(x(n-q))$$

Where $'K'$ is the order of nonlinearity, $'Q'$ is the depth of memory, $'w_{kq}'$ is the coefficient of the model, and $\emptyset_k(x)$ is the orthogonal polynomial, $x(n)$ is the input baseband OFDM signal (Orthogonal Frequency Division Multiplexing) , $z(n)$ is the signal after DPD. Using an orthogonalization procedure, we obtain the required polynomials. The weighted coefficients of the algorithm are determined by calculating the orthogonal polynomials.

The form of orthogonal polynomial is given by

$$\varphi_m(z) = \sum_{m=1}^{M} b_m |z|^{m-1}\ z$$

Fig 2. Formula for calculation of Phi (Orthogonal polynomial coefficients) H. Huaming, T. Liang, Z. Chunsheng, Y. Bin, Y. Kai and B. Zhiyong, "An adaptive pre-distortion method based on orthogonal polynomials," Proceedings of the 31st Chinese Control Conference, Hefei, China, 2012, pp. 5573-5576

Where M is nonlinear. The coefficients of DPD are obtained by using indirect learning architecture.

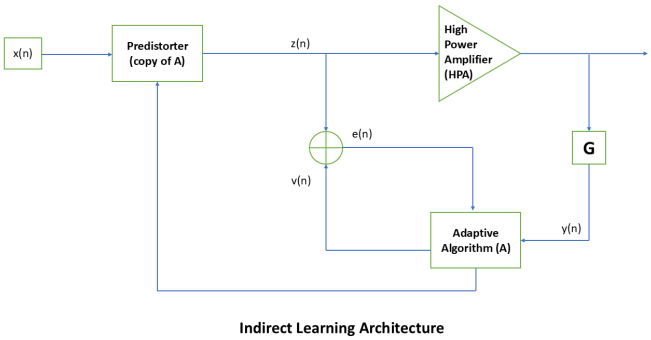


Fig 4. Indirect Learning architecture used error comparison and RLS implementation for digital pre distortion based off H. Huaming, T. Liang, Z. Chunsheng, Y. Bin, Y. Kai and B. Zhiyong, "An adaptive pre-distortion method based on orthogonal polynomials," Proceedings of the 31st Chinese Control Conference, Hefei, China, 2012, pp. 5573-5576.

According to Figure 4, z(n) represents the output of the applied pre-distortion. This output is determined by calculating the error function, which is obtained by subtracting the output of the pulse amplifier (PA), y(n), from the desired signal, v(n), which is considered as i_ref and q_ref. The Digital Predistortion (DPD) module employs an Indirect Learning Architecture. In this architecture, the polynomial coefficients are generated as shown in Figure 2. The error function and z(n) are calculated recursively. The generated weight coefficients are then used in the pre-distorter, where they are applied to the input signal through complex multiplication. The generation and application of these coefficients follow the memory polynomial model. Therefore, the architecture effectively combines both pre-distortion and post-distortion for the input signals.