A Thesis Report

On

DIGITAL PREDISTORTION IN WCDMA POWER AMPLIFIER USING EMBEDDED PROCESSOR

Submitted in the partial fulfillment of requirement for the award of the Degree of

MASTER OF TECHNOLOGY

IN

VLSI DESIGN AND CAD

Submitted by

Pawan Kumar Singh

60661017

Under the guidance of

Dr. Sanjay Sharma

Assistant Professor, ECED

Electronics and Communication Engineering Department

Thapar University

Patiala-147004 (INDIA)

June, 2008

Certificate

I hereby certify that the work which is being presented in the thesis entitled, "***Digital Predistortion in WCDMA Power Amplifier Using Embedded Processor***", in partial fulfillment of the requirements for the award of degree of **Master of Technology in VLSI Design and CAD** at **Thapar University**, Patiala, is an authentic record of my own work carried out under the supervision of **Dr. Sanjay Sharma** and refers other researcher's works which are duly listed in the reference section.

The matter embodied in this thesis has not been submitted for the award of any other degree of this or any other university.

Pawan Kumar Singh 27/06/08
(PAWAN KUMAR SINGH)

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

Dr. SANJAY SHARMA
Assistant Professor
ECED
Thapar University,
Patiala.

30.6.08.
Dr. A. K. CHATTERJEE
Professor and Head,
Electronics & Communication Engg. Department,
Thapar University,
Patiala- 147004.

Dr. R.K. SHARMA
Dean of Academic Affairs,
Thapar University
Patiala -147004

Acknowledgement

At the very outset, I wish to place on record my deep sense of gratitude and indebtedness my worthy supervisor **Dr Sanjay Sharma (Assistant Professor)**, Electronics and communication Engineering Department at Thapar University, Patiala. His dynamism and diligent enthusiasm have been highly instrumental in keeping my spirits high. His flawless & forthright suggestions blended with an innate intelligent application have crowned my task with success.

I am highly obliged to **Prof. A.K.Chatterjee**, H.O.D Electronics and Communication Engineering Department, Thapar University, Patiala for allowing me to carry out my thesis work in this University.

I would also like to offer my sincere thanks to all faculty, teaching and non-teaching, of Electronics & Communication Engg. Deptt. (ECED), and staff of central library, TU, Patiala for their assistance.

I am also thankful to the authors whose works I have consulted and quoted in this work.

Last, but not the least, very special thanks to my parents and my friends for their constant encouragement and blessings. Their patience and understanding without which this study would not have been in this present form, is greatly appreciated.

DATE **PAWAN KUMAR SINGH**

(60661017)

Abstract

Power amplifiers are essential components in communication systems and are inherently nonlinear. The nonlinearity creates spectral growth (broadening) beyond the signal bandwidth, which interferes with adjacent channels. It also causes distortions within the signal bandwidth, which decreases the bit error rate at the receiver. Newer transmission format, such as wideband code division multiple access (WCDMA) is especially open to the nonlinear distortions due to their high peak-to-average power ratios (PAPRs). If we simply back-off the input signal to achieve the linearity required for the power amplifier, the power amplifier efficiency will be very low for high PAPR signals. Another choice is to linearize a nonlinear power amplifier so that overall we have a linear and reasonably efficient device. Digital predistortion is one of the most cost effective ways among all linearization techniques. However, most of the existing designs treat the power amplifier as a memoryless device.

For wideband or high power applications, the power amplifier exhibits memory effects, for which memoryless predistorters can achieve only limited linearization performance. The memory polynomial predistorter can correct both the nonlinear distortions and the linear frequency response that may exist in the power amplifier. It is a robust predistorter, which has demonstrated good performance on several nonlinear system models.

The predistorter models considered in this dissertation include both even- and odd order nonlinear terms. Here, the benefits to include even-order nonlinear terms in both the baseband power amplifier and predistorter models are described. By including these even-order nonlinear terms, a richer basis set us obtained, which offers appreciable improvement. The ideal performance of digital predistortion certainly relies on robust predistorters that can completely compensate for the nonlinearities of the power amplifier. Then there developed a method to construct compensators for the imbalance and dc offset. This compensation technique helps to correct for the analog imperfections, which in turn improve the overall predistortion performance.

List of Figures

Dedicated to

My Grand Father **Late Shri Rajdev Singh**

Table of Contents

CHAPTER ONE

Introduction

1.1 Motivation:

Power amplifiers are indispensable components in a communication system and are inherently nonlinear. The nonlinearity generates spectral regrowth, which leads to adjacent channel interference and violations of the out-of-band emission requirements mandated by regulatory bodies. It also causes in-band distortion, which degrades the bit error rate (BER) performance. To reduce the nonlinearity, the power amplifier can be backed off to operate within the linear portion of its operating curve. However, newer transmission formats, such as wideband code division multiple access (WCDMA) and orthogonal frequency division multiplexing (OFDM), have high peak to average power ratios, i.e., large fluctuations in their signal envelopes. This means that the power amplifier needs to be backed off far from its saturation point, which results in very low efficiencies, typically less than 10% ; i.e., more than 90% of the dc power is lost and turns into heat. Considering the large number of wireless base stations deployed worldwide, improved power amplifier efficiency can substantially reduce the electricity and cooling costs incurred to the service providers. To improve the power amplifier efficiency without compromising its linearity, power amplifier linearization is essential. Among all linearization techniques, digital predistortion is one of the most cost effective [1].

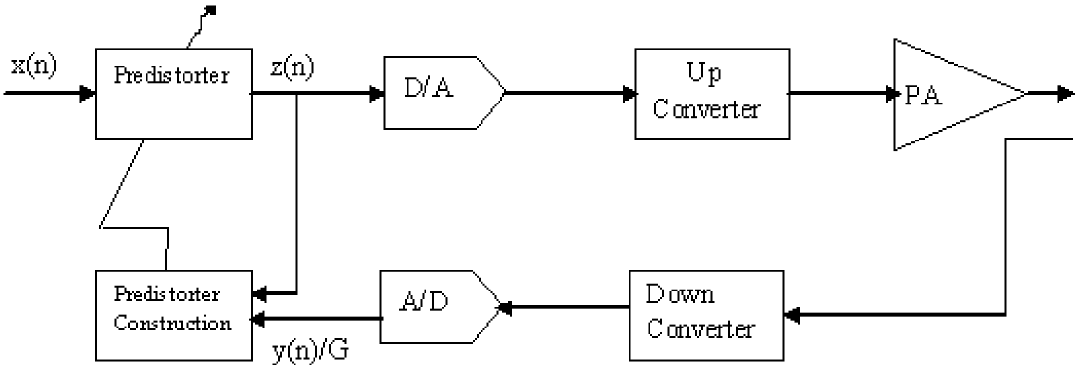


Figure 1.1: Digital Predistortion System Diagram [1]

It adds a digital predistorter in the baseband to create an expanding nonlinearity that is complementary to the compressing characteristic of the power amplifier. Ideally, the cascade of the predistorter and the power amplifier becomes linear and the original input is amplified by a constant gain. With the predistorter, the power amplifier can be utilized up to its saturation point while still maintaining a good linearity, thereby significantly increasing its efficiency. In reality, the power amplifier characteristics may change over time because of temperature drift, component aging, etc. Therefore, the predistorter should also have the ability to adapt to these changes [2].

Digital predistortion implementations in the current literature mostly focus on the power amplifier that has a memoryless nonlinearity; i.e., the current output depends only on the current input through a nonlinear mechanism. This instantaneous nonlinearity is usually characterized by the AM/AM and AM/PM responses of the power amplifier, where the output signal amplitude and phase deviation of the power amplifier output are given as functions of the amplitude of its current input. There has been intensive research on predistortion techniques for memoryless power amplifiers during the past decade. As the signal bandwidth gets wider, such as in WCDMA, power amplifiers begin to exhibit memory effects. This is especially true for those high power amplifiers used in wireless base stations. The causes of the memory effects can be attributed to thermal constants of the active devices or components in the biasing network that have frequency dependent behaviors. As a result, the current output of the power amplifier depends not only on the current input, but also on past input values [2].

In other words, the power amplifier becomes a nonlinear system with memory. For such a power amplifier, memoryless predistortion can achieve only very limited linearization performance .Therefore, digital predistorters also need to have memory structures. This dissertation investigates robust predistorter models that are capable of linearizing power amplifiers with memory effects. It also investigates system implementation issues related to these wideband digital predistortion systems.

1.2 Objectives:

The objective of this dissertation is to develop digital predistortion systems for linearization of power amplifiers with memory effects. Our research efforts focus on three areas;

- Predistorter models with memory structures;
- Digital compensation techniques of analog imperfections in the transmitters;
- Wide band digital predistortion testbed.

The ideal performance of digital predistortion certainly relies on robust predistorters that can completely compensate for the nonlinearities in the power amplifier. In reality, however, the performance can also be affected by the analog imperfections in the transmitter, which are introduced by the analog components, such as mixers, analog filters, and quadrature modulators. The second focus of this dissertation is to investigate modeling and compensation techniques for these imperfections. In this dissertation, a wideband digital predistortion testbed is also developed to evaluate the performance of digital predistortion systems on real power amplifiers.

The W-CDMA system is one of the leading wideband cellular technologies used in the 3G Cellular market. At the heart of the W-CDMA system is the 3G base station. The early W-CDMA trial system, initially proposed and developed by NTT DoCoMo, and the UMTS Terrestrial Radio Access (UTRA), developed by Siemens and others, served as the basis for consideration by the International Telecommunications Unions IMT2000 initiative. The initiative used these two efforts as a baseline to eventually merge the development efforts into one standard (W-CDMA) with two implementations: Frequency Division Duplex (FDD) and Time Division Duplex (TDD), governed by the Third Generation Partnership Project (3GPP) organization. The system is sometimes referred to as 3GPP W-CDMA to differentiate it from the earlier wideband CDMA versions.

As the importance of IP-based services increases, there are growing demands to improve coverage and throughput and reduce the delay of the uplink. Applications that can benefit from an enhanced uplink include services such as video clips, multimedia, e-mail, telematics, gaming, and video streaming. This application note focuses on the FDD mode of the W-CDMA specification, although most of the content is also applicable to TDD.

Whenever the term W-CDMA is used, it is referring to the 3GPP. With the deployment of the 3G-wireless infrastructure gaining momentum, equipment manufacturers and network operators are searching for a highly optimized base band processing solution with the greatest freedom in the areas of cost, flexibility, and scalability to meet increasing data service and time-to-market requirements. This application note provides an overview of the base band processing of a typical W-CDMA base station, along with the associated implementation challenges faced by W-CDMA equipment manufacturers, including the silicon cost, flexibility, and scalability trade-offs. It also provides a summary of an optimal solution to meet the extensive processing requirements of the W-CDMA specifications while retaining flexibility and minimizing the overall cost.

CHAPTER TWO

Power Amplifiers Basics

2.1 Power Amplifier Fundamentals:

Power Amplifiers are devices used to amplify signals in order to obtain high signal powers necessary for transmission via a propagation medium. They are indispensable in wireless communications. The following section is a brief introduction of some fundamental Power Amplifier features.

2.2 Gain and Output Power:

In mobile communications each system has its specifications which must be fulfilled. Obtaining output powers high enough for various applications is a very important task achieved by Power Amplifiers. In general the information signal is first modulated and up converted, and then sent to a PA. This input is multiplied with a gain factor and the desired output power is obtained. Gain is handled in dB and power in dBm throughout this thesis. Fig. 2.1 (a) and (b) show example PA output and gain versus input power characteristics of a linear PA respectively. PA output versus input power characteristics shown in fig. 2.1 (a) is also called AM/AM characteristics of the PA As it can be seen from the figures the gain is constant for low input powers and it reduces with approaching its saturation region. Saturation region is easily visible from the output power curve where the output power stays constant with further increase of the input power. In the fig. 2.1 (a) 1 dB compression point is also shown, which refers to the output power level at which the amplifier's transfer characteristics deviates from the ideal one by 1 dB [3]. This is a widely used measure of amplifier linearity revealing roughly which linear output power value is achievable with the device under test (DUT).

2.3 Linearity:

Linearity is one of the key issues in Power Amplifiers used in new generation mobile communication systems. The linearity of a Power Amplifier is easily visible in its gain and phase characteristics.

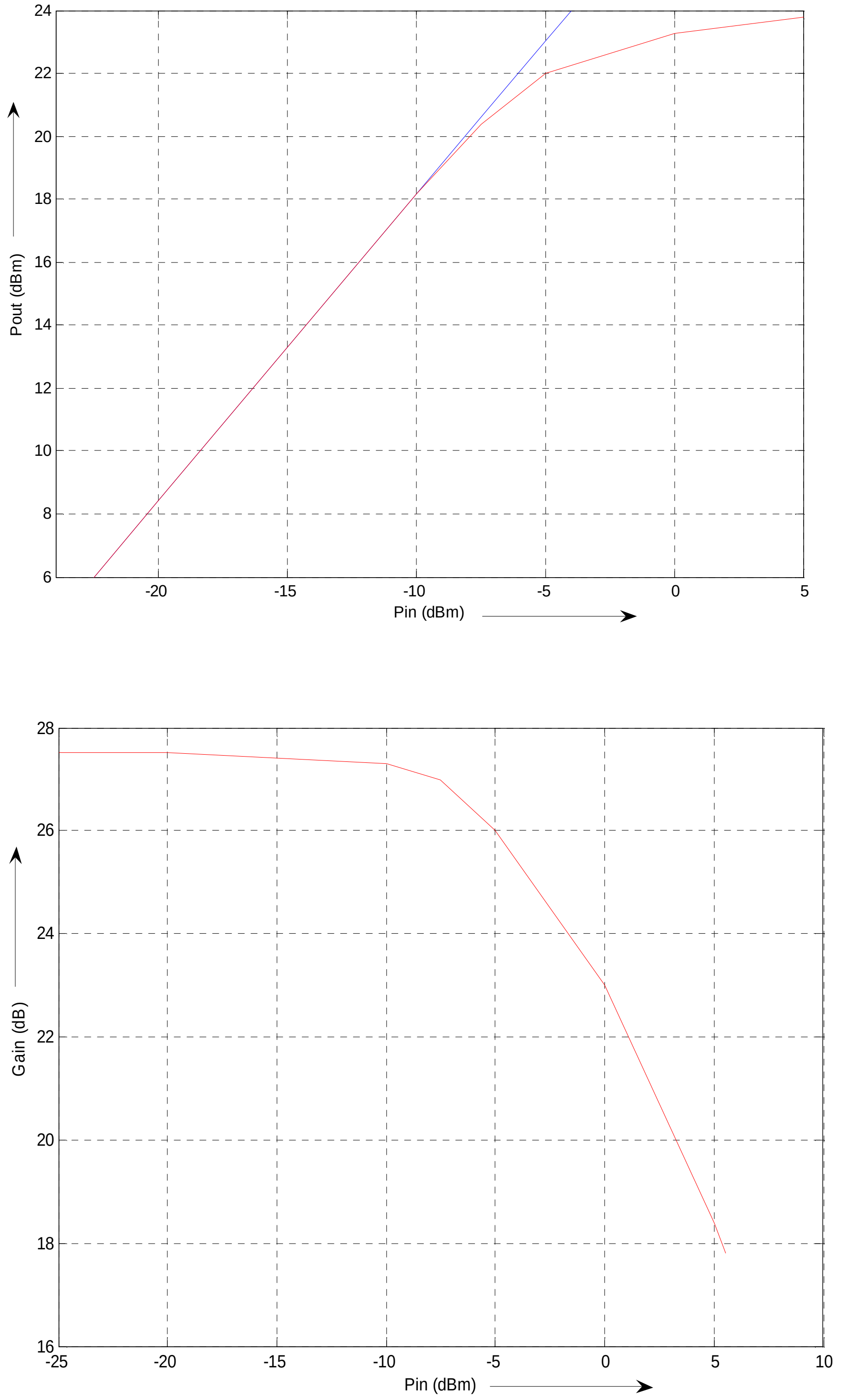


Figure 2.1: (a) Output power and (b) gain characteristics of a PA. [3]

If an amplifier has a constant gain and phase response for an input power region, then the amplifier is said to be linear for this region. Fig. 2.2 (a) and (b) show typical and desired amplifier gain and phase characteristics respectively. Solid lines are gain and phase characteristics of a memoryless PA and dashed lines indicate the ideally linear PA gain and phase characteristics. In general after reaching a relatively high output power value the amplifier gain decreases gradually with increasing input power because the PA reaches its saturation point. Phase nonlinearity increases also with increasing input power.

Amplifier phase characteristics shown in fig. 2.2 (b) are also called as AM/PM characteristics. The other way of determining PA nonlinearity is using second and third order intercepts points. The advantage is that it is a fixed quantity from which the distortion level at a particular operating point may be predicted. There are some conventional analog techniques used to design linear Power Amplifiers by optimizing linearity and efficiency through bias and matching adjustments. However, these analog techniques have their limits and achieving a highly linear gain and phase response simultaneously is very difficult. Currently these methods are widely used and achievable performance is close to its limits. Therefore some other sophisticated solutions are necessary to solve the problem.

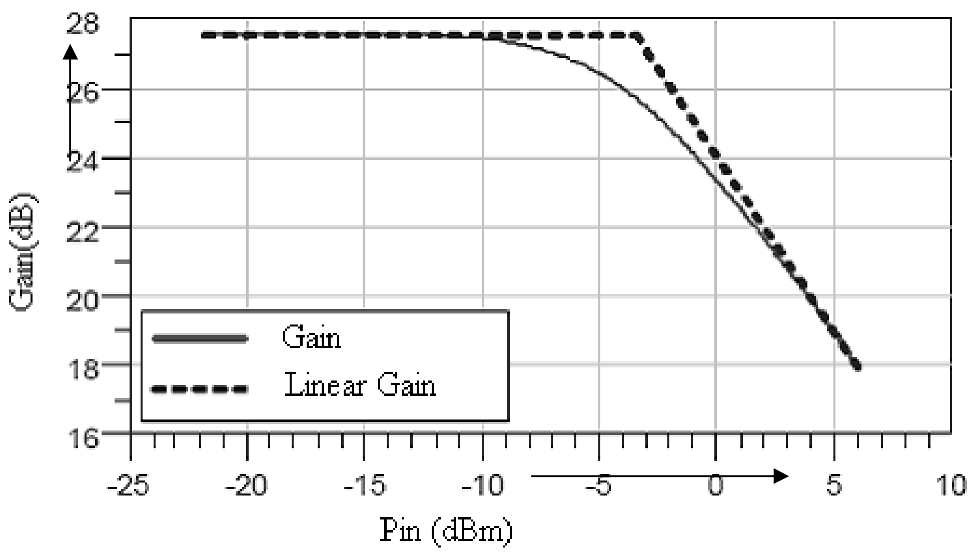


Figure 2.2 (a): Nonlinear and ideally linear PA gain characteristics. [3]

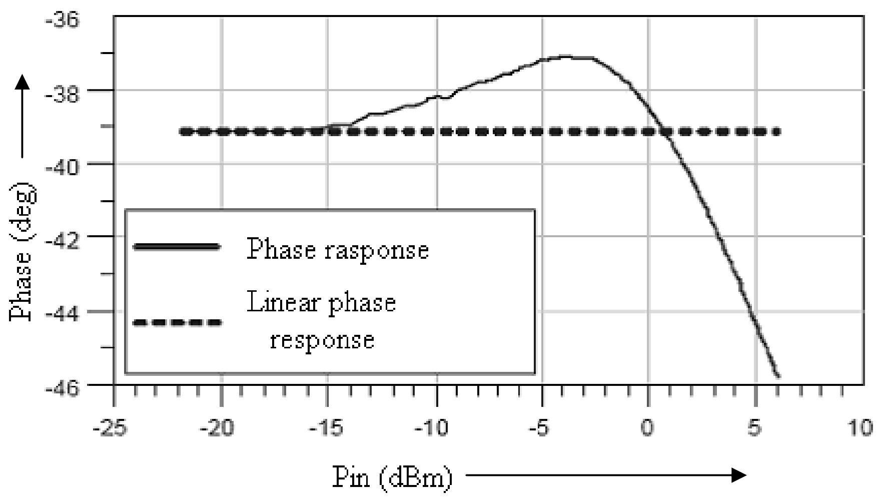


Figure 2.2 (b): Nonlinear and ideally linear PA phase characteristics. [3]

The reason why the linearity is so important is the varying signal envelopes in spectrum efficient modulation types used in new generation mobile communication systems. If signals have constant envelopes like in FM (Frequency Modulation) or GMSK (Gaussian Minimum Shift Keying) then PA linearity is not an important issue because the instantaneous input power stays constant and therefore there are no gain and phase variations for a specific operation point. However, if the signal envelope varies, then the instantaneous input power changes continuously. As a result the signal at the PA output is distorted if the amplifier gain and phase response are not linear. This distortion can be measured in terms of IMD (Inter modulation Distortion), ACPR (Adjacent Channel Power Ratio) or EVM (Error Vector Magnitude) Fig. 2.3 show possible degradation of PA output ACPR (for WCDMA) and EVM due to its nonlinearity [3,5]. If a two-tone signal is applied to a nonlinear device, then a large number of harmonics and IMPs (Inter modulation Products) are generated depending on the nonlinearity degree of the device. The odd-order IMPs (3*rd*, 5*th*, 7*th*, etc) are the most important ones because they fall into the neighborhood of the main signal and therefore not easily filterable. The most commonly used measure of IMD is the ratio of the largest IMP to the amplitude of one of the two equal tones. ACPR is caused by IMPs falling in

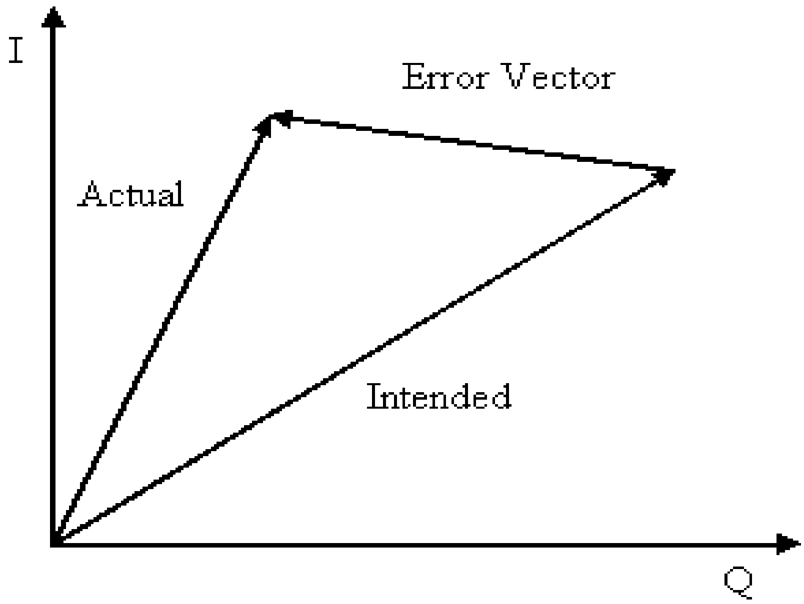


Figure 2.3: Error Vector Magnitude

the signal neighborhood in the case of complex modulated signals composed of a lot of spectral components. It is defined as the ratio of the distortion signal power falling in the adjacent channels to the carrier power (main channel power) in dB. In fig. 2.3 (a) PA input and output signals are compared. Under ideal conditions PA output is a shifted version of the input in the vertical direction by an amount equal to PA gain. However, the PA output in fig. 2.3 (a) has some unwanted distortion elements in the neighbor channels indicating PA nonlinearity. ACPR is a very critical issue in mobile communications. A transmitter must fulfill the specifications and not to disturb dedicated neighbor channels because they are usually used by other transmitters. EVM can be defined as the distance between the desired and actual signal vectors (error vector), normalized to a fraction of the signal amplitude [5]. In fig. 2.3 (b) the degradation in output signal phasor is shown which corresponds the signal constellation. The actual value of the constellation point can deviate from the ideal one significantly depending on PA nonlinearity. EVM can be defined for each symbol k as

$$EVM(k) = \frac{|E(k)|}{\sqrt{\frac{1}{N}\sum_{k=1}^{N}|S(k)|^2}}$$

where E(k) is the error vector for symbol k, S(k) is the ideal signal vector of the symbol k and N is the number of symbols. Root-mean-square (RMS) value of EVM for a number of symbols is a widely used measure of system linearity and it can be defined as

$$EVM_{RMS} = \frac{\sqrt{\sum_{k=1}^{N} |E(k)|^2}}{\sqrt{\sum_{k=1}^{N} |S(k)|^2}}$$

EVM is an in band distortion causing high bit error rates during reception of the transmitted data. Therefore EVM specifications must also be fulfilled in order to have proper communication.

2.4 Efficiency:

Efficiency is another key issue in mobile communications [3, 10], especially for battery operated mobile terminals. It has two widely used definitions, drain (or collector) efficiency and PAE (Power Added Efficiency). Drain efficiency is the ratio of output radio frequency (RF) power to input DC power

$$\eta = Pout\ RF\ /PDC$$

and PAE is the overall efficiency obtained by subtracting input drive power from output RF power and divides it by input DC power [6].

$$\text{PAE} = (PoutRF - Pdrive)/PDC$$

If the gain of a PA is high then its drain efficiency and PAE are close and simply drain efficiency can be used in calculations.

CHAPTER THREE

Power Amplifier Linearization Methods

Power amplifier linearization is currently one of the most promising techniques for linearity and efficiency improvement in mobile communication systems. There are numerous techniques which have different levels of complexity, various advantages and limitations [2]. Different linearization methods may fit to different communication systems. For example more sophisticated high performance systems may be used for base station PAs whereas the systems usable in handsets should have low complexity, low cost and high efficiency. Although in general the main reason to implement these systems is to linearize the PA, they improve also the efficiency because a linearized PA can be driven closer to compression (operation with low back-off). In the following sections several PA linearization methods are explained which can be classified mainly as feedback, feedforward and predistortion systems.

3.1 Feedback:

Feedback linearization methods are relatively simple compared to feedforward and conventional predistortion. The idea is to force the PA output to follow its input. There are different types of feedback linearization topologies classified mainly as RF feedback and modulation feedback which can be divided again into two: polar and Cartesian feedback. In modulation feedback the modulation components (I&Q or R&θ) of PA input and output are compared whereas in RF feedback the RF signals are compared. Feedback systems can be implemented at RF, IF or baseband frequencies. A major issue in feedback linearization is the stability due to delays in feedback which is critical, especially in systems with discrete components. The group delay increases significantly due to PA matching circuits or filters and couplers in the loop. Especially high order filters like SAWs (Surface Acoustic Wave filter) can not be used due to large group delays in the order of several hundred nanoseconds [2]. In RF power amplifiers it is difficult to have a high loop gain and good stability with RF feedback. Increasing operation frequency in RF feedback or modulation BW in modulation feedback makes stability issues even more critical. Feedback methods are not suitable for wide band applications in the order of several MHz because the stability and correction capability of a feedback loop is limited by its gain-bandwidth product (GBW). Feedback systems can not distinguish between nonlinearities of PA and other nonlinearities in the forward or feedback paths. Fortunately the forward path nonidealities like nonlinearities, gain and phase mismatches in quadrature modulators can be corrected sufficiently by a close to perfect feedback path. Consequently, the feedback path must be as linear as possible, because it can not be corrected [2, 37].

3.1.1 RF feedback:

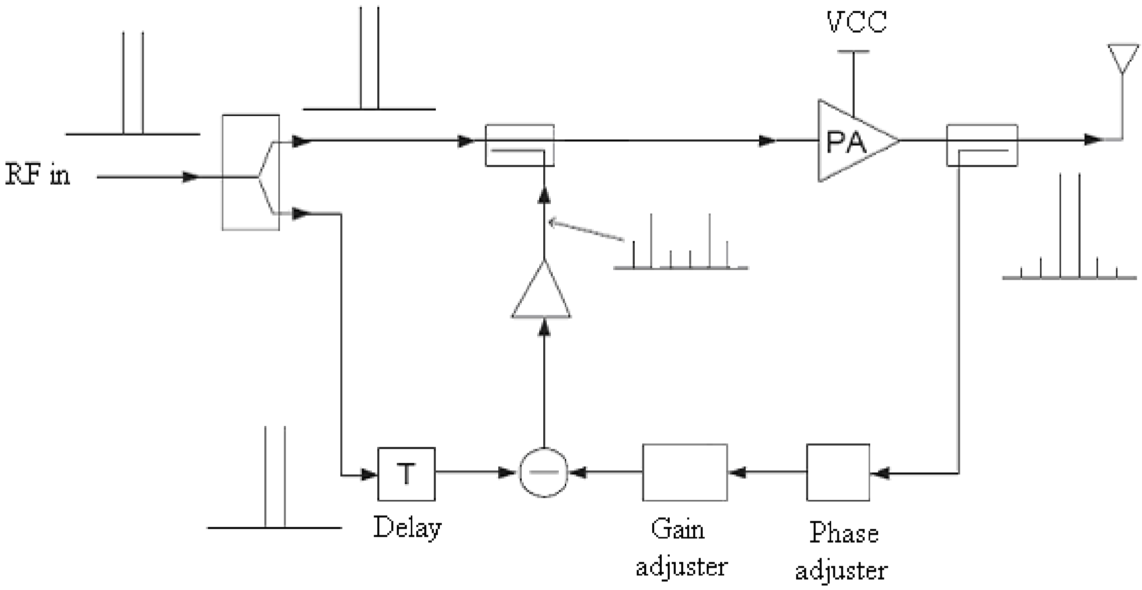


Figure 3.1: RF feedback linearization topology [2].

In this linearization type the RF output is subtracted directly from the input without demodulation or down conversion. Various topologies are possible in RF feedback. They are mainly used to linearize an individual stage rather than a transmitter whereas modulation feedback techniques can be applied to complete transmitter systems. Gain stabilization is an important application of RF feedback. There are some RF feedback topologies applied to systems containing complete transmitters. However, the delay in the loop has to be small to ensure the stability, which may be achieved with MMIC (Monolithic Microwave Integrated Circuit) devices rather than discrete elements. Fig. 3.1 is an example of such an RF feedback topology called distortion feedback. The input signal is divided into two parts. One part is compared with the distorted PA output signal to obtain an error signal canceling the error generated by the PA. This error signal is added to the second part of the input signal and sent to the PA input. The system looks like a predistortion system but uses real time feedback.

3.1.2 Polar loop:

In polar loop linearization AM/AM and AM/PM nonlinearities are corrected by two individual loops as shown in fig. 3.2. The feedback is done in general at IF but RF Implementation is also possible. In [2] a polar loop concept is presented having low complexity and being applicable to EDGE. In the previous chapter PLTx has been introduced which is an efficiency enhancement method, rather than linearity improvement method but it has also a good degree of linearity.

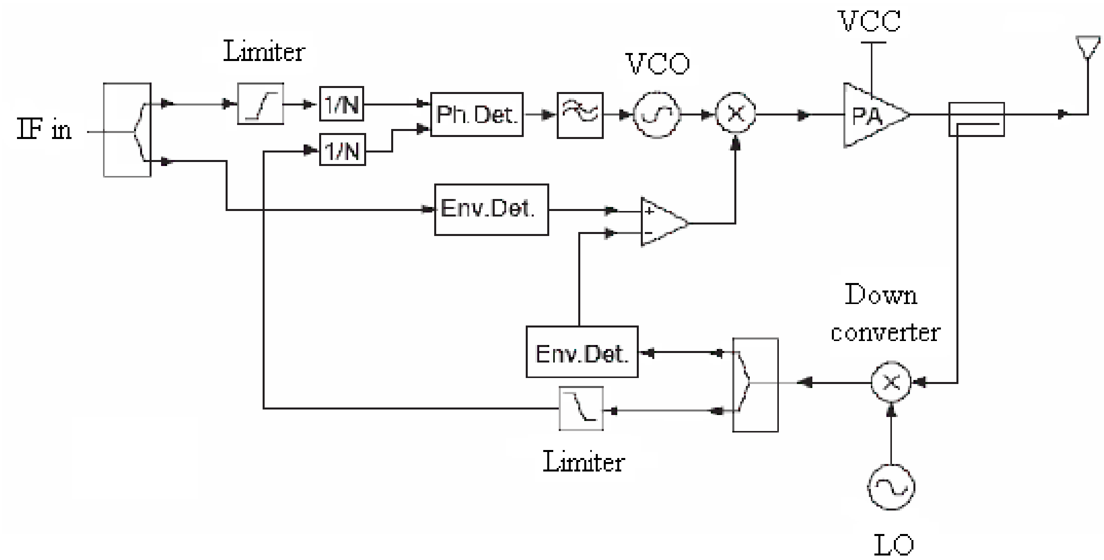


Figure 3.2: Polar loop linearization topology [2].

The topology shown in fig. 3.2 is different from PLTx in the sense that the amplitude loop does not modulate the PA supply voltage or bias, but modulates the envelope of the PA input signal. The input is assumed to be an envelope and phase modulated IF signal. It is first split into two parts. Envelope of the first one is detected in order to obtain AM data and the second one is sent to a limiter to obtain just the phase modulated carrier. A portion of the PA output is taken using a coupler and it is down converted. This signal is also split into two parts. Again one part is envelope detected to obtain AM data and the other is sent to a limiter to obtain the phase modulated carrier, both including PA nonlinearities. The two phase modulated signals from input and feedback are sent to frequency dividers and a PLL in order to force phase modulation to cancel the AM/PM nonlinearity of the PA. Also, the envelope detected input and feedback signals are sent to a differential amplifier to obtain an error signal canceling the AM/AM nonlinearity. VCO and differential amplifier output signals are sent to a mixer in order to obtain an amplitude and phase modulated PA input. In addition to fig. 3.2 there are various implementation possibilities. For example in some cases the required amount of linearization is low and it is sufficient to correct just AM/AM nonlinearities with envelope feedback. Another difference can be related to the phase loop where a phase shifter controlled by the phase error between input and feedback signals can be used instead of the PLL. The topology shown in fig. 3.2 can be realized at RF instead of IF with the result of reduced circuit complexity. As a rule of thumb the correction loop bandwidth in a feedback system should be at least 5 times higher than the signal bandwidth in order to achieve a stable operation [6]. If a modulated signal with pulse shaping done in Cartesian coordinates is separated into its magnitude and phase components, then these have considerably higher bandwidths than the original signal. This additional increase in the bandwidth makes the stability issue in polar loop linearization more critical because the loop bandwidth should be accordingly higher for a proper operation. If we assume that the possible linearization bandwidth at cellular frequencies is typically less than 1 MHz with polar loop [6] (this is the bandwidth of magnitude or phase signal), then the maximum bandwidth of the signal to be corrected is much lower than 1 MHz, which is not sufficient for CDMA or WCDMA systems. Another disadvantage of the polar loop is locking problems of the PLL in the phase loop due to small instantaneous powers and abrupt phase jumps [2, 17].

3.1.3 Cartesian loop:

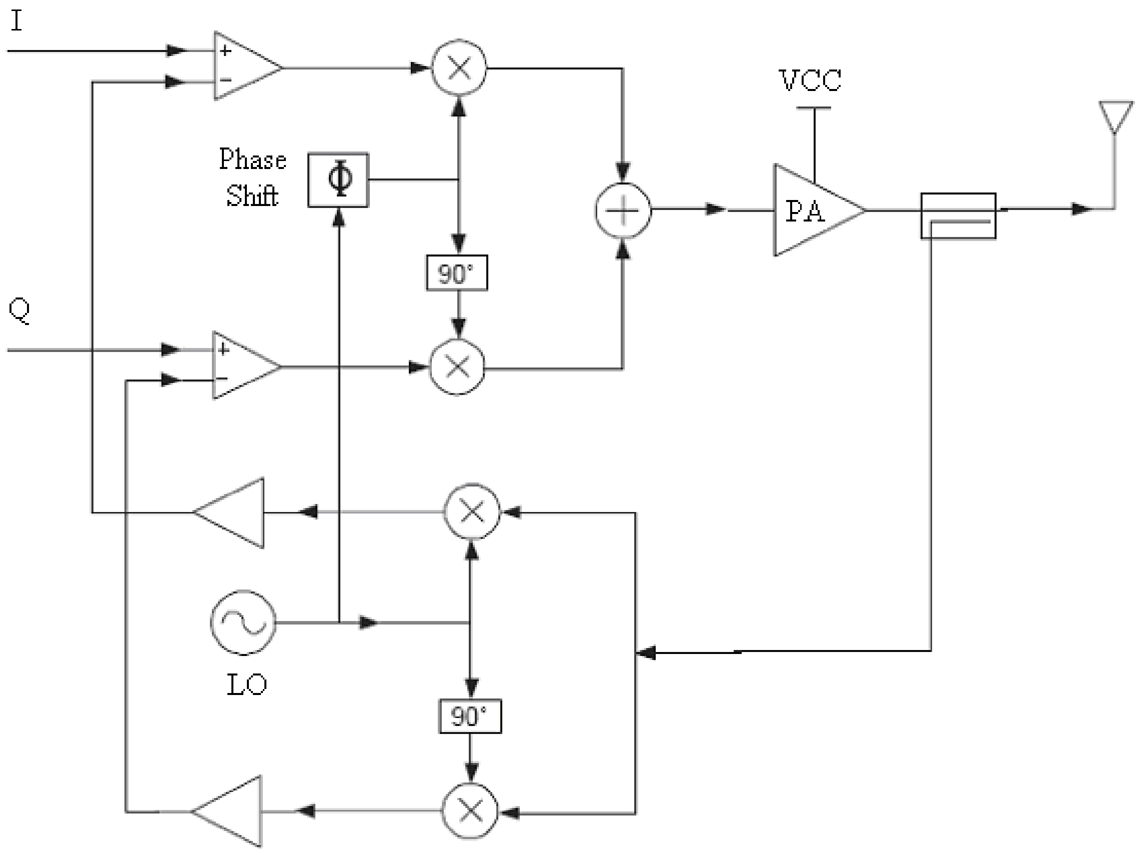


Figure 3.3: Cartesian loop linearization block diagram.

The problem of bandwidth widening in polar loop can be solved by Cartesian loop linearization technique. Fig. 3.3 shows a general block diagram. In Cartesian feedback I- and Q-signals are used to correct amplifier nonlinearities instead of R and θ as in polar loop. I and Q have similar spectral properties as RF signal whereas R and θ have much larger bandwidths [5]. In fig. 3.3 a demodulator is used in the feedback path in order to obtain distorted Cartesian modulation components from PA output. Demodulated and scaled I- and Q-signals are subtracted from the input I- and Q-signals to obtain the PA input baseband signals resulting in linear operation The circuit in fig. 3.3 is implemented in RF because IF Cartesian loop is undesirable due to the additional delay added to feedback loop. Compared to polar loop, Cartesian loop is easier to implement, because in general digital circuits generate the baseband signals in I&Q form. Cartesian loop has been already implemented in narrow band systems [26]. For narrow band signals achievable degree of linearity is high. For wide band systems, however, it is not suitable due to high loop delays resulting in stability problems. A phase adjuster is necessary for LO signal in Cartesian loop in order to maintain the relationship between the input and feedback signals. Since LO phase shift depends on temperature, power level and process variations which make the stability issue difficult, a control loop for phase shift may be necessary in order to maintain the synchronization in the system.

3.2 Feed-forward:

The feed forward method allows high linearization performance and is currently used in base stations of mobile communications systems. The idea is to extract the distortion at the PA output, amplify it and add it to the PA output in opposite phase in order to cancel the distortion. Out of all linearization methods, only feedforward systems provide a very good distortion reduction over a wide bandwidth. The drawback of these systems is the low power efficiency due to high power requirement of the error amplifier operated in class A mode and losses due to couplers and delay lines in the system.

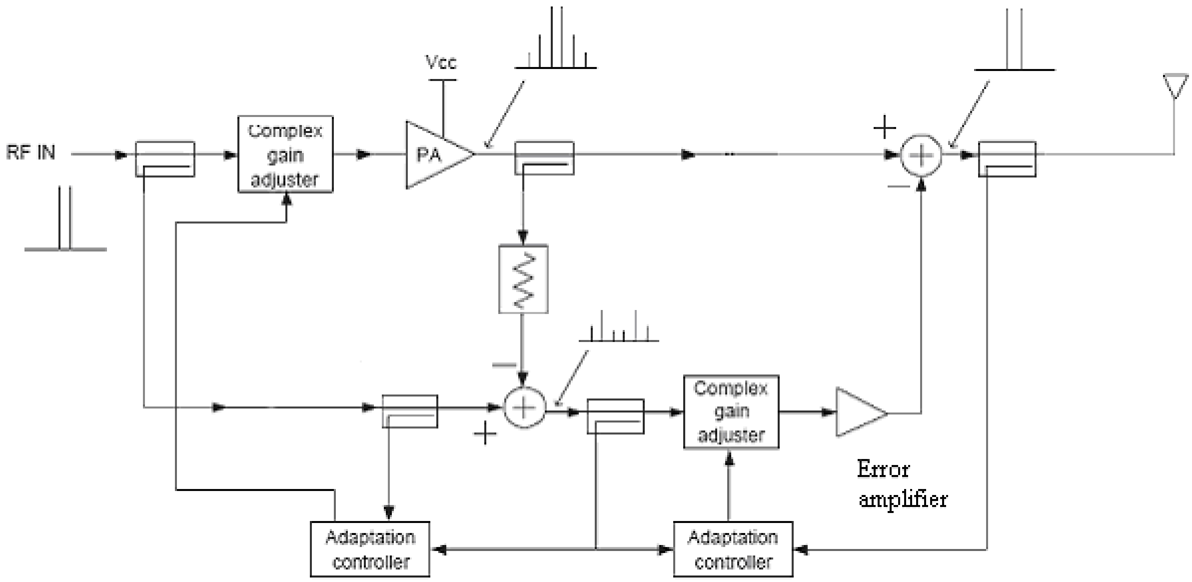
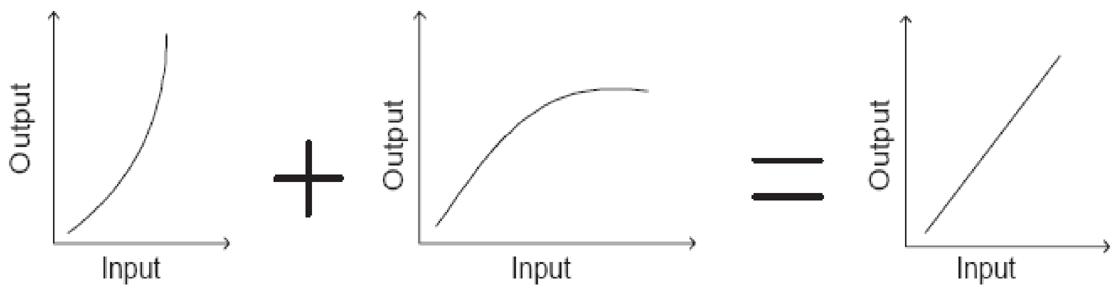


Figure 3.4: Feedforward linearization block diagram [39].

Fig. 3.4 shows the basic block diagram of feedforward comprising a PA and a high number of external components. The system is composed of two main parts: signal cancellation and error cancellation loops. The signal cancellation loop works as follows. First, a portion of the PA input signal is taken using a coupler, which is used as reference signal in the loop. A portion of the PA output is also taken and its power is reduced to the level of the reference signal from input by using an attenuator. The subtraction of these two signals gives the error signal generated at the PA output which is going to be used to cancel the PA distortion at the system output. A delay line is implemented in the reference signal path in order to compensate time delay differences between these two paths. Since feedforward is based on subtraction and addition of almost equal signals, it must have a precisely controlled adaptation compensating the possible changes of the system due to changing environmental or operating conditions. Therefore the adaptation controller of the signal cancellation loop adjusts the complex gain adjuster in front of the PA in order to obtain just the error signal at the output of the loop. The controller uses the reference and error signals to do this adjustment. The error cancellation loop works in a way similar to the signal cancellation loop. The error signal obtained from the signal cancellation loop is phase shifted using a complex gain adjuster in front of the error amplifier. It is amplified by the error amplifier and added to the delayed PA output. Also in error cancellation loop a precise control of the complex gain adjuster is required. This is achieved by a second controller using the error signal from the signal cancellation loop and the resultant system output. A second delay line is used at the PA output in order to eliminate time mismatch between upper and lower paths in the error cancellation loop and achieve a proper error cancellation at the output of the final combiner. For relatively narrow band signals delay lines are good enough solutions to adjust the phases such that the reference and error signals are sufficiently cancelled in the loops. However, for very wide bandwidth signals this is difficult to achieve. A possible solution is presented in [11] using a phase equalizer to obtain a better phase match at each frequency in the band of interest. Adaptive feedforward linearization is able to handle wide bandwidths with continuous adjustment for component drifts and power level changes. The adaptation is done by using pilot signals or gradient methods. The main strength of feedforward is its high level of inter-modulation reduction capability for wideband signals with unconditional stability [11]. It has no stability problem because all signals in the system flow in the same direction meaning no feedback loops causing conditional stability. Distortion due to memory effects can also be compensated with feed forward because these effects are also included in the error signal and this is subtracted from the PA output [39]. The main issues to be considered in feedforward are improving system efficiency and reducing complexity. This Proposes a method immigrating a part of the analog signal processing into the digital domain, which reduces the power requirement of the error amplifier meaning better efficiency and eliminates the output delay line. However, feedforward needs some more improvement in terms of efficiency in order to become usable in battery operated handsets.

3.3 Predistortion:

The idea behind predistortion is to expand the input signal prior a PA in such a way that the nonlinearities due to the PA are compensated. It is realized by implementing a nonlinear block in front of the nonlinear PA generating input signal level dependent distortion elements opposite of the distortion caused by the PA. As a result the cascade of these nonlinear blocks has a linear response.



Predistorter | Power Amplifier | Predistorter+Power Amplifier

Figure 3.5: Basic steps of Predistortion

The figure shows AM/AM correction done by the predistorter. In general the similar correction is also done for phase (AM/PM) by adding a phase to the input signal opposite of the phase shift due to the PA. Predistortion can be realized in two different ways, analog or digital implementation. Analog predistortion is realized by creating the required AM/AM and AM/PM nonlinearities canceling the effects of PA using analog components. It can be implemented at RF, IF or baseband. In the digital predistortion case however the system is realized with digital components and it is implemented usually at baseband. In predistortion systems first the PA characteristics must be obtained in order to calculate the required inverse nonlinearity compensating the PA nonlinearity. For a good system performance precise AM/AM and AM/PM characteristics are required. Although in some cases they can be assumed not to depend on frequency (static case), actually they do. In the case of static AM/AM and AM/PM characterization the PA output is assumed to depend just on the momentary input signal [6]. However, in general the output depends on the momentary and previous input signal values. These previous values can change depending on signal bandwidth. This means the PA behaves in different ways if modulation signals with different bandwidths (for example two tone signals with same power but different tone spacing) are applied to it. This phenomenon is called memory effect. In this thesis digital predistortion systems will be divided into two: systems taking memory effects into account and systems neglecting it such as MDP. Memoryless predistorters are much simpler compared to predistorters with memory and if memory effects in a PA are not strong, then memoryless predistortion is a reasonable solution with high performance. However, the amount of linearity improvement is limited if the PA exhibits strong memory effects [6].

3.4 Analog Predistortion:

The advantages of analog predistortion are its relatively simple circuitry, low cost, low power consumption, wideband signal handling capability and integrity. However, these systems can have in general just a moderate linearization performance and they introduce insertion loss. Moreover if they are implemented adaptively, then system complexity may increase significantly. There are various ways to implement analog predistortion. It can be a simple circuit composed of diodes or transistors as in RF predistorters, or it can be composed of multipliers to realize polynomial nonlinearities.

The system can be adaptive or fixed depending on environmental conditions and system specifications. However, a reliable system must have a kind of adaptation adjusting the predistorter according to the environmental conditions especially in today's mobile communication systems, which may operate under extreme conditions and still must fulfill the specifications. High linearity systems based on RF predistortion are extremely difficult to achieve and are not widely available. A typical ACPR improvement of about 10 dB is possible. A simple predistorter based on a diode circuit is presented which improves ACPR performance by about 5 dB. In analog predistortion the gain and phase flatness of the predistorter and of the PA limit the operating bandwidth, and the memory effects in both of them limit the linearization performance.

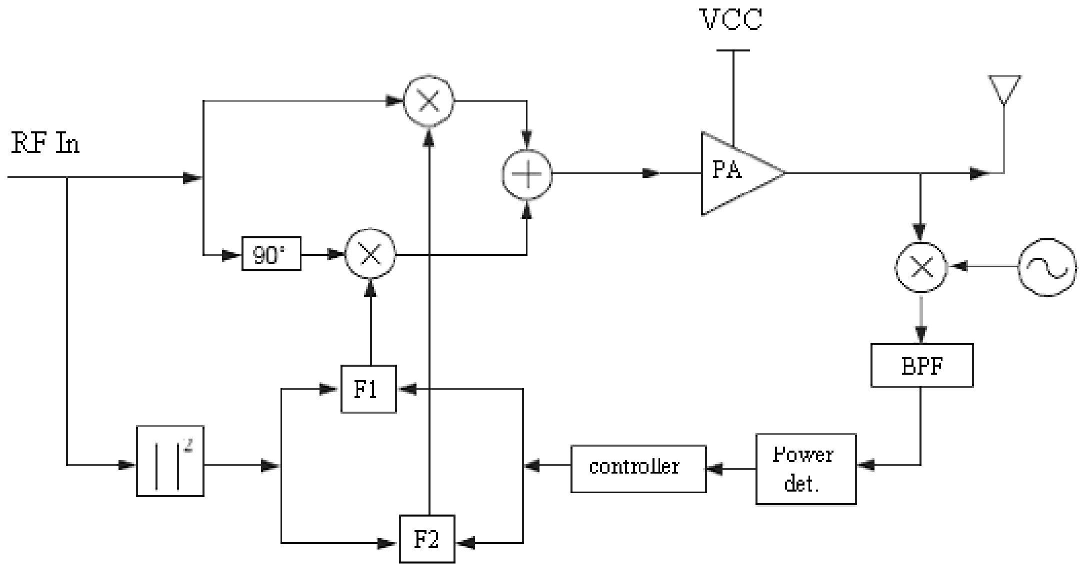


Figure 3.6: Block diagram of an adaptive predistorter

RF predistortion operates just in front of the PA at high frequency. Fig. 3.1 shows the block diagram of a possible implementation. The predistortion is done using a circuit changing the PA input signal according to polynomial functions. These polynomials are implemented using analog circuits and their coefficients are supplied and changed by a digital control circuit. The system has an adaptation based on adjacent channel emission. The measured power in adjacent channels of PA output and squared magnitude of the system input signal are used in order to calculate the required polynomial multiplicands $F1$ and $F2$. After multiplying the input signal with $F1$ and $F2$, the signal is predistorted in a way that PA output linearity is improved.

The predistorter shown in fig. 3.5 is implemented at RF, but IF implementation is also possible. IF predistortion is a reasonable way for linearization because then it is possible to use the system for more than one frequency by using local oscillators for desired up- and down conversions. In the presented system the feedback path utilizes signal down conversion to simplify the design of the high selectivity BPF which is supposed to give a signal related to the adjacent channel power emission at the PA output. An IF predistorter is presented which is similar to the explained RF predistorter. The down converter in its feedback and up converter in its forward paths do the required frequency conversions.

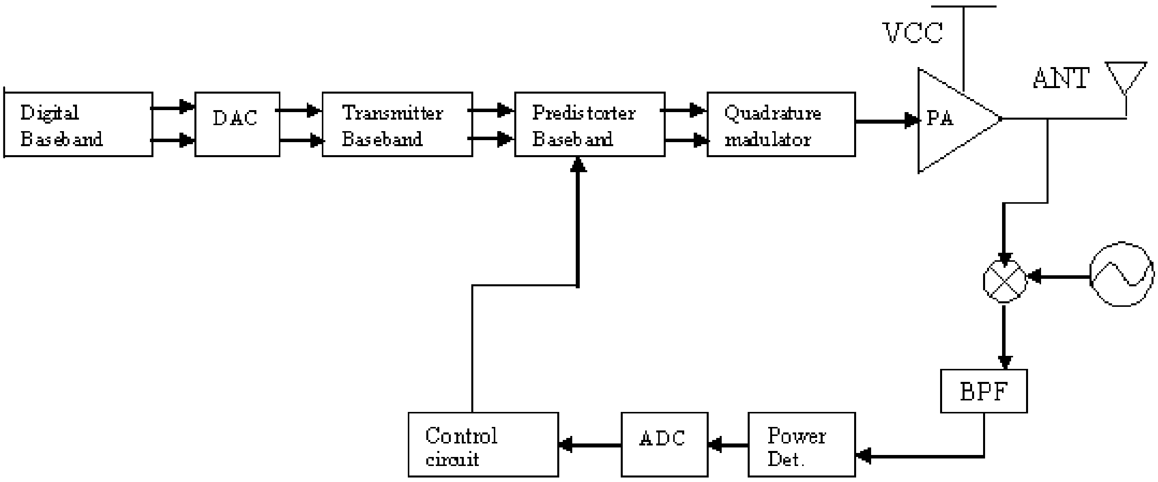


Figure 3.7: Block diagram of an adaptive baseband predistorter

Fig.3.6 shows a possible implementation of adaptive baseband analog predistortion. The control circuit in the feedback path is similar to the presented RF predistorter. A portion of the output signal is down converted, the power in the adjacent channel is detected and the required polynomial coefficients are calculated. Actually different coefficient sets can be stored in a LUT and the required coefficients can be read from this table during operation. The adaptation may be used for coefficient update in this LUT. The predistortion takes place in the block named as"Predistorter Baseband" in fig. 3.6.

The circuit inside this block responsible for polynomial predistortion at baseband is shown in fig. 3.6 Two 5*th* order polynomial predistortion signals *Fi* and *Fq* are generated and these are multiplied with the input baseband signals *Iin* and *Qin* using a complex multiplier. The coefficients ci, cq, c3i, c3q, c5i and c5q are either read from a LUT or taken from the"Control Circuit" in fig 3.5. The similar predistorters are presented, which can be implemented both in IF and baseband.

3.5 Digital Predistortion:

DP is usually implemented in digital baseband but it is also possible to do it at IF. The theory behind is the same as in analog predistortion. This method is in general used for base stations in mobile communication systems in order to improve linearity, which is very important in systems with wide bandwidths. A significant improvement can be achieved for class B and AB amplifiers in applications requiring high linearity. DP is simple compared to feed forward linearization widely used in base stations. It is unconditionally stable and a precise linearization is possible. Since the implementation is done with digital components, the system is more robust and flexible compared to analog circuits [12]. The method is robust against drifts in PA characteristics due to component variations, temperature, aging, if it is implemented adaptively.

This is at the expense of increased complexity due to significant amount of added hardware and software. The main drawback of DP is its limited bandwidth. Since all data samples must be modified continuously, a high amount of mathematical operations in digital domain is required. If the sample frequency or bandwidth of the modulation signal increases, then the required clock frequency and the amount of operations also increase. Higher clock frequency or higher amount of operations means higher power consumption. However, there is a continuous improvement in digital IC technology. Power consumption de creases as the integration level increases. Therefore DPD seems to be one of the most promising linearization techniques for future applications.

We can divide adaptive DPD into two parts: the forward and feedback paths. Fig. 3.3 shows a simplified block diagram of the system. The forward path is actually a transmitter chain with an additional block where predistortion is applied. The feedback path however is composed of additional components required to update the predistorter in order to track PA characteristics changes and ensure a reliable operation. Complexity level of the feedback path can change for different applications. It can be like a complete receiver or be in a simpler form compromising system performance and complexity. The contents of"Analog Feedback Circuitry" block in fig. 3.3 can be a demodulator and the required baseband circuitry, or an envelope and a phase detector, or a circuit detecting the power in the adjacent channels. For all these cases different adaptation circuits are required in order to update the predistorter block.

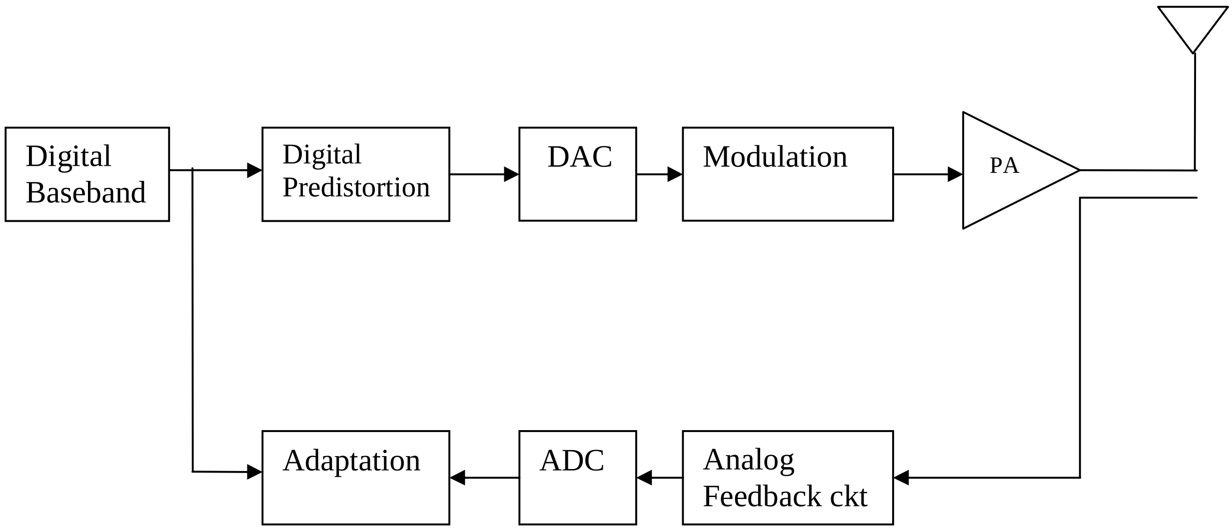


Figure 3.8: Simplified block diagram of adaptive digital predistortion.

The predistorter in the forward path must operate continuously whereas the feedback path can be used just when it is required or it can be programmed to operate with a given duty cycle. Therefore the forward path is very critical, and the maximum operation bandwidth is limited by the clock frequency and the amount of required operations.

The amount of mathematical operations in the adaptation unit is in general quite high compared to the predistorter in the forward path but since the adaptation unit does not need to operate continuously, power consumption can be kept reasonably low. In some systems like TDMA, where transmission and reception are not done simultaneously, the additional feedback path may be avoided by using the available receiver path for adaptation. However, if there is simultaneous transmission and reception as in CDMA and WCDMA systems, then the receiver path can not be used and additional components for adaptation path are required. the main advantages and drawbacks of feedback, feed forward, analog and digital predistortion systems are summarized. Advantages of DPD make it the favorite method applicable in terminal applications.

3.6 Look-up-Table (LUT) Based Predistortion:

LUT based predistorter is simpler compared to polynomial one in the critical forward path. A magnitude or squared-magnitude calculation of a complex signal in the addressing unit and a complex multiplication unit are required whereas in polynomial predistorter significantly more multiplication and addition operations may be required depending on the order of polynomial. Fig. 3.5 shows a LUT based adaptive DP system is going to be explained later in more detail. The predistorter is composed of an addressing unit, a LUT, a delay unit and a complex multiplier.

Nagata has proposed a LUT based adaptive DP having a big LUT size because of addressing by both input I- and Q-signals. This is a two dimensional LUT (mapping predistorter). If I- and Q-branches have k quantization levels, then the size of the LUT is k2. However, the amplifier nonlinearity does not depend on the input signal phase but just its amplitude if it is assumed to be memoryless, which gives the possibility to reduce the size of the LUT to k. In this case the addressing unit calculates the amplitude of the input signal and LUT addressing is done accordingly [9]. This is called gain based predistorter. Instead of the magnitude, magnitude square of the input can also be used which is proportional to the signal power. In the following the LUT based adaptive digital predistorter will be explained in detail. PA characterization, LUT coefficient calculation, system operation and its adaptation will be handled separately.

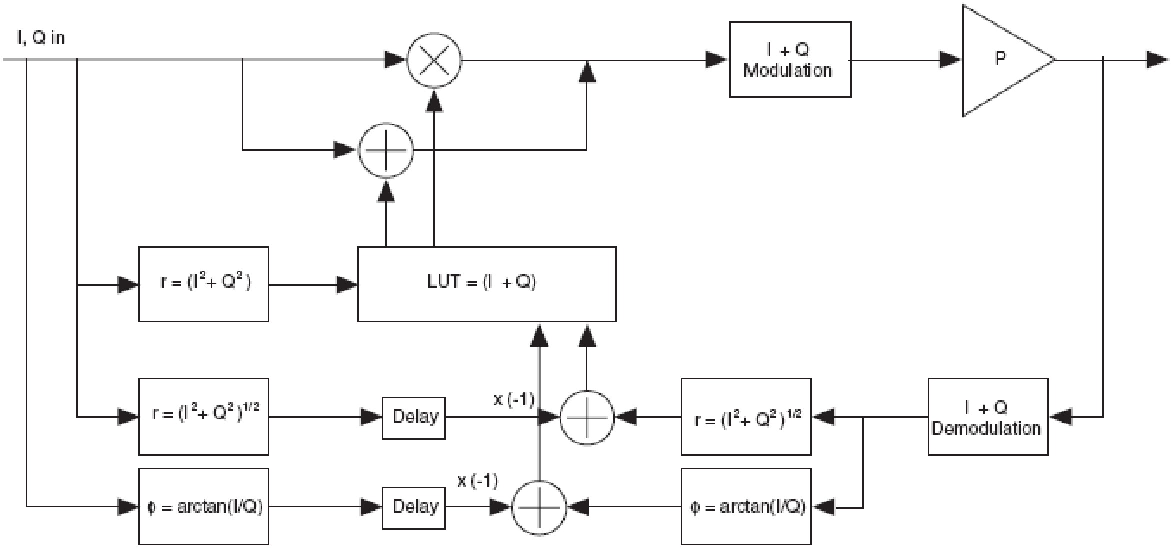


Figure 3.9: LUT based digital predistortion [8]

3.7 System Description:

For predistorting an input signal applied to a radio frequency (RF) power amplifier, the apparatus comprising: an RF phase discriminator, for generating output signals indicative of phase and amplitude differences between two input signals ; means for coupling a portion of an RF amplifier input as a first input signal to the RF phase discriminator; means for coupling a portion of an RF amplifier output as a second input signal to the RF phase discriminator, wherein the coupled portion of the RF amplifier output is scaled to be comparable with the RF amplifier input; an analog-to-digital converter for converting difference signals generated by the RF comparator to digital form; a computation module for deriving from the difference signals corresponding values of gain compression and RF amplifier output phase; means for coupling a portion of the RF amplifier input to the analog-to-digital converter, to derive RF amplifier input values in digital form; a digital memory, for storing in association with each other, values of RF amplifier input, gain compression and RF amplifier output phase; and an amplifier predistorter, for adjusting the amplitude and phase of the RF amplifier input based on the stored values of gain compression and RF amplifier output phase, to compensate for distortion in the RF amplifier.

In the RF phase discriminator comprises: first and second input terminals for receiving the first and second inputs derived from the amplifier input and output signals, respectively; and first and second output terminals for outputting in-phase (I) and quadrature (Q) output signals containing phase difference and amplitude difference information. Wherein the analog-to-digital converter receives as parallel inputs the I component and Q output signals and generates the digital equivalents of these difference signals. And further comprising: means for coupling a portion of the RF amplifier input to a third parallel input of the analog-to-digital converter, which generates the digital equivalent of the RF amplifier input. In the digital memory comprises a lookup table having data fields for storing RF amplifier input values and corresponding values of amplifier gain compression and amplifier output phase; and the amplifier predistorter includes means for accessing the lookup table based on a current value of RF amplifier input, and retrieving the corresponding values of amplifier gain compression and amplifier output phase. Wherein the values of amplifier gain compression and amplifier output phase are stored in the lookup table as running averages, whereby any changes in amplifier distortion characteristics are automatically reflected in the lookup table and used to apply predistortion. In the digital memory comprise additional lookup tables, each corresponding to a different RF amplifier operating frequency [9]. A method for predistorting an input signal applied to a radio frequency (RF) power amplifier, to compensate for amplifier distortion, the method comprising the steps of: coupling a portion of an RF amplifier input as a first input signal to an RF phase discriminator; coupling a portion of an RF amplifier output as a second input signal to the RF phase discriminator, wherein this coupling step includes scaling the RF amplifier output to be comparable with the RF amplifier input; generating, in the RF phase discriminator, output signals indicative of phase and amplitude differences between the first and second input signals ; converting, in an analog-to-digital converter, the RF comparator output difference signals from analog to digital form; computing from the digital difference signals corresponding values of RF amplifier gain compression and RF amplifier output phase; storing, in association with each other in a lookup table, values of RF amplifier input, amplifier gain compression and amplifier output phase; and predistorting the RF amplifier input in amplitude and phase, based on stored values of gain compression and amplifier output phase associated with a desired value of RF amplifier input.

In the step of converting the difference signals to digital form comprises receiving as parallel inputs to the analog-to-digital converter in-phase (I) and quadrature (Q) difference signals from the RF phase discriminator and generating the digital equivalents of these difference signals and further comprising: coupling a portion of the RF amplifier input to a third parallel input of the analog-to-digital converter; and generating the digital equivalent of the RF amplifier input the storing step comprises storing in the lookup table data values for the RF amplifier inputs and corresponding values of amplifier gain compression and amplifier output phase; and the predistorting step includes accessing the lookup table based on a current value of RF amplifier input, and retrieving the corresponding values of amplifier gain compression and amplifier output phase.

The storing step stores running averages of the amplifier gain compression and amplifier output phase in the digital memory comprises additional lookup tables, each corresponding to a different RF amplifier operating frequency, and wherein the storing step includes selecting a lookup table based on RF amplifier frequency, and storing in the selected lookup table values for RF amplifier input, amplifier gain compression and amplifier output phase in the step of predistorting comprises automatically compensating for changes in amplifier characteristics that cause corresponding changes in amplifier distortion [9].

This block diagram illustrates the software/hardware boundary for the adaptive linearization circuit. In addition to a Power Amplifier, it also requires a coupler, quadrature modulator and demodulator as well as an A/D and D/A converter. Note that the same oscillator is used in the Up and down conversion for coherence; some methods require a phase shifter for achieving stability. The linearizer creates a predistorted version of the desired modulation.

The predistorter consists of a complex gain adjuster which controls the amplitude and phase of the input signal. The amount of predistortion is controlled by the entries of a Look-up Table that interpolate the AM/AM and AM/PM nonlinearities of the power amplifier. Note that the envelope of the input signal is an input to the Look-up table. The feedback path samples the distorted signal for which the DSP adjusts the Look-up Table entries so as to minimize the level of distortion.

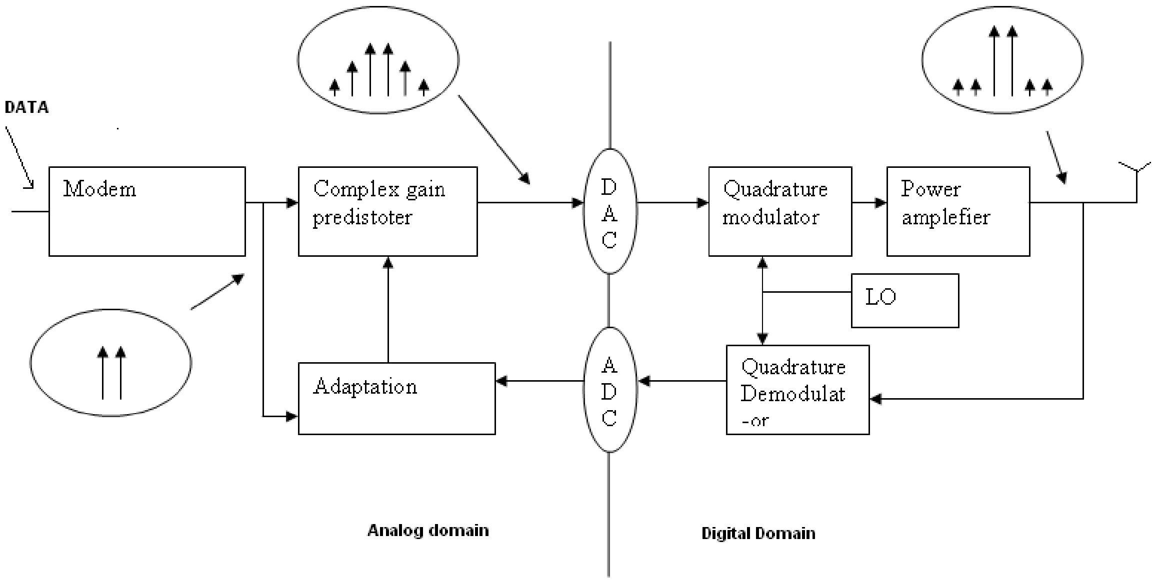


Figure 3.10: Predistortion at base station

We can observe the spectral response at various nodes in the digital baseband predistorter given a two tone input signal. The function of the adaptation block is to extract the amplitude modulation of the input RF signal and adjust the Look-up Table entries. The entries of the Look-up Table will be indexed by the level of the input signal envelope. The complex gain adjuster, once optimized, will provide the inverse nonlinear characteristics to that of the power amplifier. Thus one can observe at the input mode to the power amplifier, the spectral growth from the predistorter. Ideally the inter modulation products will be of equal amplitude but in anti-phase to those created as the two tones pass through the power amplifier. The function of the DSP adaptation block is to adjust the Look-up Table entries so as to minimize the difference between the input signal and a scaled version of the output signal from the power amplifier. Thereby minimizing the level of distortion.

3.8 Calculation of LUT Coefficients:

The next step after obtaining PA characteristics is LUT coefficient calculation. In a memoryless PA the output depends just on the momentary input value which makes LUT coefficient calculation very simple compared to systems with memory and it results in a small size LUT. The process is the same after obtaining AM/AM and AM/PM characteristics either with static or dynamic characterization. Fig. 3.11 shows a possible nonlinear AM/AM characteristics given as f(x) and the target linear AM/AM as $g(x)$. Two points $y1$ and $y2$ have been selected from the figure such that

$$y1 = f(x1)$$

$$y2 = f(x2)$$

Where xi and yi are magnitude or squared magnitude of the PA input and output respectively.

$$y2 = g(x1)$$

is the desired function having a linear response. If we define the coefficient $cx1$ as

$$cx1 = x2/x1$$

multiply it with $x1$ and apply to the function f, then it is possible to obtain the linear Function g as shown in the equation below.

$$y2 = f(x2) = f(cx1x1) = g(x1)$$

This means for each xi value a coefficient cxi must be calculated and stored in a LUT in order to cover the complete input signal range and linearize the PA accordingly. The size of LUT depends on quantization levels of the input signal. cxi is magnitude of the complex coefficient stored in the LUT for xi, which is responsible for AM/AM correction [9]. The phase of each coefficient should also be calculated and stored in the LUT using nonlinear AM/PM characteristics of the PA in order to correct also AM/PM nonlinearities. As it is seen from in fig. 3.11 a predistorter can correct distortion up to the full saturation level of the PA where any further increase of the input does not increase the output. Fig. 3.11 shows possible improvement in the deliverable maximum linear output power with DPD. A is a point where the PA has a highly linear response without DPD. This is achieved due to its high back-off. In a linear system there is no compression and input and output PARs are equal. If the input is increased such that the PA operates at point B, then a compression reducing the output PAR occurs which must be corrected using DPD [10].

Calculating the phase of LUT coefficients is a relatively easy process compared to amplitude calculation. This is an extension of the fig. 3.11 with nonlinear and target AM/PM characteristics of the PA. Phase of the output is ph1 for input of $x1$ and ph2 for $x2$. However, the phase of LUT coefficient should be stored for $x1$ is not (pht−ph1) but (pht−ph2) because if the input is $x1$, then it is multiplied with the magnitude coefficient $cx1$ and as a result the input becomes $x2$. This means the instantaneous phase is shifted from ph1 to ph2 and a phase shift equal to (pht − ph2) is required in order to obtain a constant phase at the output for this input value. The calculation of LUT coefficients must be done according to the preferred addressing. Two mostly used types are magnitude and power addressing. The difference between these two is explained. Power addressing is the easiest one and it requires squaring the signals in I- and Q-branches and adding them up in the addressing block.

The result is squared magnitude of the complex input signal which corresponds to its power. Since the LUT coefficients are calculated for input signal values equispaced in power, there is a high coefficient density (low spacing between coefficients) near to saturation, which means good linearization for low back-off operation. However, if the PA has also nonlinearities in the high back-off region as in class B, then the correction may not be sufficient in this region. Magnitude addressing gives in general a better IMD suppression due to its uniform coefficient distribution equispaced in voltage for both high and low back-off. However, for magnitude addressing a square-root calculation unit is required in addition to the operations in power addressing method which may be a computationally intensive block. One other possibility is using optimum addressing. In this case the coefficients are dense in the regions where the PA nonlinearity is high and less dense in low nonlinearity regions. This means the coefficients are closely spaced in highly nonlinear regions in order to increase the performance. According to, the IMD performance of the magnitude addressing is better than the power addressing by about 10 dB but the best IMD performance is obtained with the optimum addressing which is 1-4 dB better than the magnitude addressing. However, optimum addressing is cumbersome and it depends on the amplifier, modulation signal and back-off during the operation. Therefore the amplitude addressing is a good overall compromise in terms of its performance, modulation format and PA type independency and simplicity. These issues are important especially for applications in battery operated small size handsets which should have a simple structure.

3.9 Measurement of AM/AM and AM/PM Characteristics:

In a predistortion system, if the measured AM/AM and AM/PM characteristics are not precise, the calculated LUT coefficients will not match the PA and performance of the linearization will be degraded. The easiest way of measuring AM/AM and AM/PM characteristics of a PA is using single tone excitation as shown in fig. 3.11 However, with this method just a static characterization is possible because the input signal envelope is constant, which means a constant instantaneous power during the excitation.

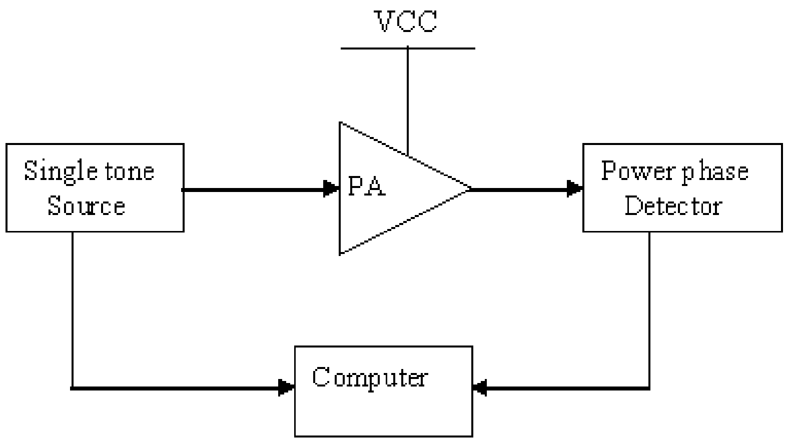


Figure 3.11: Static AM/AM & AM/PM measurement setup.

In single tone measurements the obtained characteristics are based on average power measurements. However, in memoryless predistortion systems the incoming signals are modified according to their instantaneous power or magnitude. Therefore a Power Amplifier characterization based on instantaneous power measurement should give better results compared to the static characterization. A Power Amplifier characterization method with multi tone excitation is proposed which gives a dynamic AM/AM and AM/PM characterization due to its varying input signal envelope. It has been observed that there is a significant difference between the characteristics obtained with average and instantaneous power measurements. The instantaneous power measurement may exhibit hysteresis loops in the gain and phase characteristics depending on the input power which is a measure of memory effects. If the obtained data is fitted to a polynomial function, then these Power Amplifier characteristics can be used in memoryless Digital Predistortion systems like static AM/AM and AM/PM characteristics and it is supposed to give better results compared to static characterization because it reflects the behavior of the Power Amplifier during operation with modulated input signals [6].

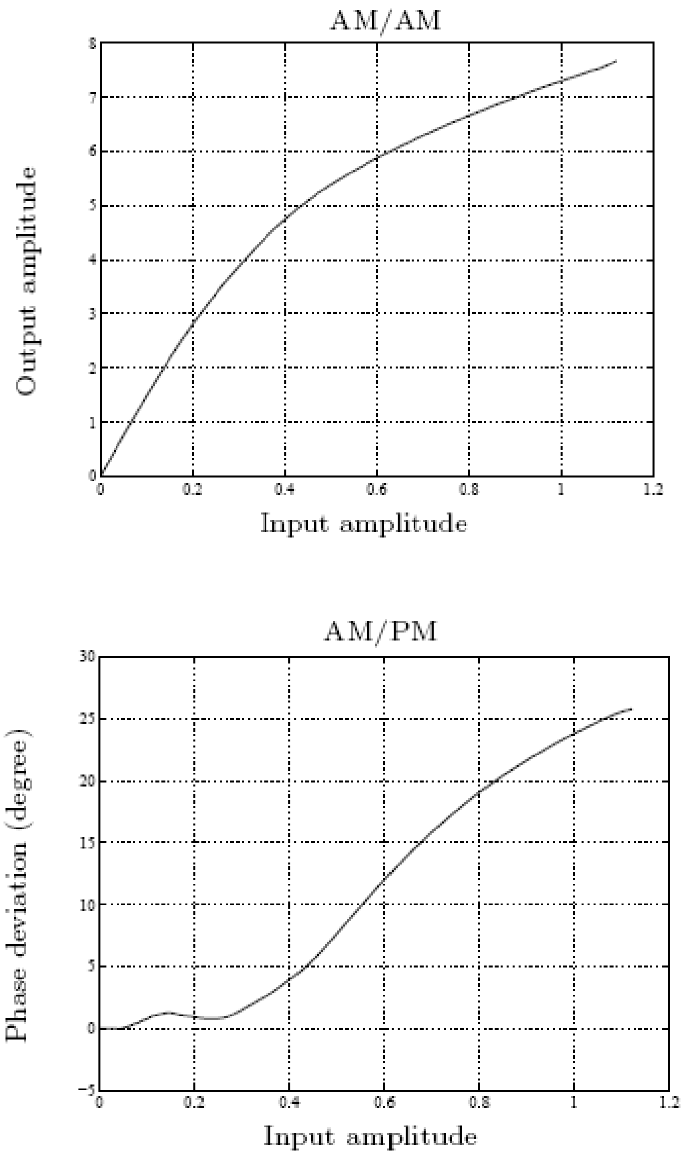


Figure 3.12: Static AM/AM and AM/PM characteristics before linearization

3.10 Comparison Between Different Linearizer:

Table 1: Comparison Between Different Linearizer

	Advantages	**Drawbacks**
Feedback	Adaptation to environmental Conditions	Useful for narrowband systems · Stability problems
Feed forward	High degree of linearization Wideband signal handling capability	High complexity Large size Low efficiency
Analog predistortion	Wideband signal handling Capability Moderate power consumption	Usable for weakly nonlinear Pas Moderate linearization High complexity if conventional adaptation is used
Digital predistortion	High degree of linearization High flexibility High reliability High integrability No stability problem Progress in digital ICs	Limited bandwidth High power consumption and clock frequencies for wideband systems High complexity if conventional adaptation is used

CHAPTER FOUR

Introduction to FPGA Design

4.1. Field Programmable Gate Array (FPGA):

Field Programmable Gate Arrays are called this because rather than having a structure similar to a PAL or other programmable device, they are structured very much like a gate array ASIC. This makes FPGAs very nice for use in prototyping ASICs, or in places where and ASIC will eventually be used. For example, an FPGA maybe used in a design that need to get to market quickly regardless of cost. Later an ASIC can be used in place of the FPGA when the production volume increases, in order to reduce cost.

4.2. FPGA Architectures:

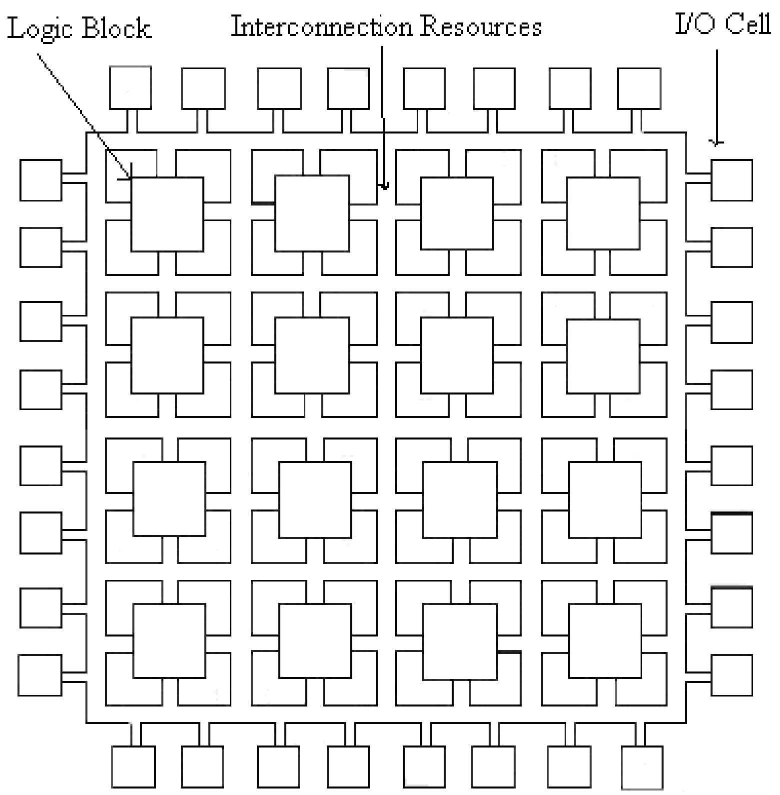


Figure 4.1: FPGA Architecture

Each FPGA vendor has its own FPGA architecture, but in general terms they are all a variation of that shown in Figure 4.1. The architecture consists of configurable logic blocks, configurable I/O blocks, and programmable interconnect. Also, there will be clock circuitry for driving the clock signals to each logic block, and additional logic resources such as ALUs, memory, and decoders may be available. The two basic types of programmable elements for an FPGA are Static RAM and anti-fuses.

4.3. Configurable Logic Blocks:

Configurable Logic Blocks contain the logic for the FPGA. In large grain architecture, these CLBs will contain enough logic to create a small state machine. In fine grain architecture, more like a true gate array ASIC, the CLB will contain only very basic logic. The diagram in Figure 4.2 would be considered a large grain block. It contains RAM for creating arbitrary combinatorial logic functions [25]. It also contains flip-flops for clocked storage elements, and multiplexers in order to route the logic within the block and to and from external resources. The multiplexers also allow polarity selection and reset and clear input selection.

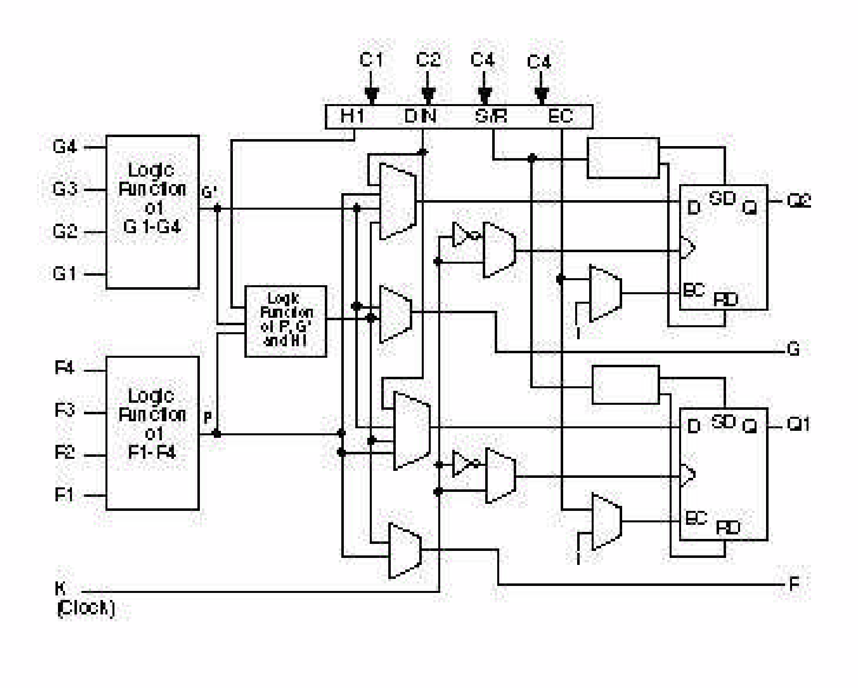


Figure 4.2: FPGA Configurable Logic Block [25]

4.4. Configurable I/O Blocks:

A Configurable I/O Block, shown in Figure 4.3, is used to bring signals onto the chip and send them back off again. It consists of an input buffer and an output buffer with three state and open collector output controls. Typically there are pull up resistors on the outputs and sometimes pull down resistors. The polarity of the output can usually be programmed for active high or active low output and often the slew rate of the output can be programmed for fast or slow rise and fall times. In addition, there is often a flip-flop on outputs so that clocked signals can be output directly to the pins without encountering significant delay. It is done for inputs so that there is not much delay on a signal before reaching a flip-flop which would increase the device hold time requirement.

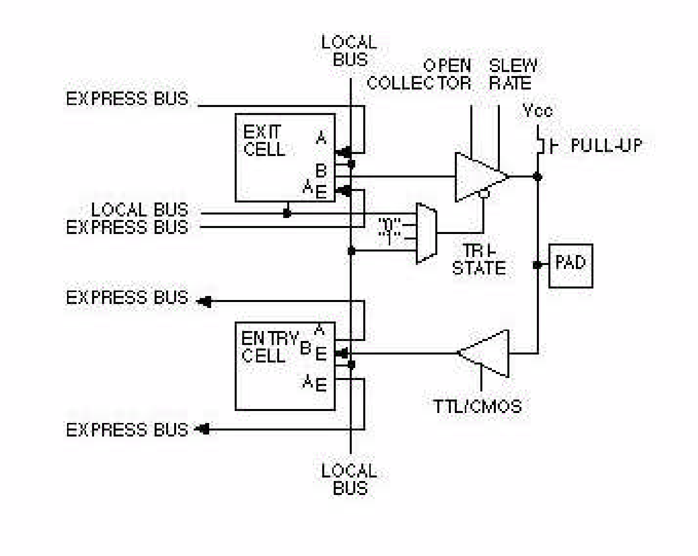


Figure 4.3: FPGA Configurable I/O Block

4.5. Programmable Interconnect:

Interconnect of an FPGA is very different than that of a CPLD, but is rather similar to that of a gate array ASIC. In Figure 4.4, a hierarchy of interconnect resources can be seen. There are long lines which can be used to connect critical CLBs that are physically far from each other on the chip without inducing much delay. They can also be used as buses within the chip. There are also short lines which are used to connect individual CLBs which are located physically close to each other. There are often one or several switch matrices, like that in a CPLD, to connect these long and short lines together in specific ways. Programmable switches inside the chip allow the connection of CLBs to interconnect lines and interconnect lines to each other and to the switch matrix [25]. Three-state buffers are used to connect many CLBs to a long line, creating a bus. Special long lines, called global clock lines, are specially designed for low impedance and thus fast propagation times. These are connected to the clock buffers and to each clocked element in each CLB. This is how the clocks are distributed throughout the FPGA.

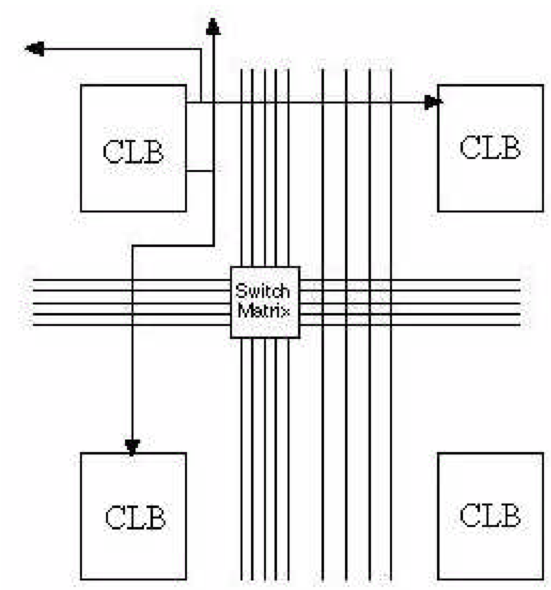


Figure 4.4: FPGA Programmable Interconnect

4.6. Clock Circuitry:

Special I/O blocks with special high drive clock buffers, known as clock drivers, are distributed around the chip. These buffers are connected to clock input pads and drive the clock signals onto the global clock lines described above. These clock lines are designed for low skew times and fast propagation times. As we will discuss later, synchronous design is a must with FPGAs, since absolute skew and delay cannot be guaranteed. Only when using clock signals from clock buffers can the relative delays and skew times are guaranteed.

4.7. Small v/s Large Granularity:

Small grain FPGAs resemble ASIC gate arrays in that the CLBs contain only small, very basic elements such as NAND gates, NOR gates, etc. The philosophies that small elements can be connected to make larger functions without wasting too much logic. In a large grain FPGA, where the CLB can contain two or more flip-flops, a design which does not need many flip-flops will leave many of them unused. Unfortunately, small grain architectures require much more routing resources, which take up space and insert a large amount of delay which can more than compensate for the better utilization.

Small Granularity	**Large Granularity**
Better utilization	Fewer levels of logic
Direct conversion to ASIC	Less interconnect delay

A comparison of advantages of each type of architecture is shown in Table. The choice of which architecture to use is dependent on your specific application.

4.8. SRAM v/s Anti-fuse Programming:

There are two competing methods of programming FPGAs. The first, SRAM programming, involves small Static RAM bits for each programming element. Writing the bit with a zero turns off a switch, while writing with a one turns on a switch. The other method involves anti-fuses which consist of microscopic structures which, unlike a regular fuse, normally make no connection. A certain amount of current during programming of the device causes the two sides of the anti-fuse to connect. The advantages of SRAM based FPGAs is that they use a standard fabrication process that chip fabrication plants are familiar with and are always optimizing for better performance. Since the SRAMs are reprogrammable, the FPGAs can be reprogrammed any number of times, even while they are in the system, just like writing to a normal SRAM [25]. The disadvantages are that they are volatile, which means a power glitch could potentially change it. Also, SRAM based devices have large routing delays. The advantages of Anti-fuse based FPGAs are that they are non-volatile and the delays due to routing are very small, so they tend to be faster. The disadvantages are that they require a complex fabrication process, they require an external programmer to program them, and once they are programmed, they cannot be changed.

4.9. Example of FPGA Families:

Examples of SRAM based FPGA families include the following:

- Altera FLEX family
- Atmel AT6000 and AT40K families
- Lucent Technologies ORCA family
- Xilinx XC4000 and Virtex families

Examples of Anti-fuse based FPGA families include the following:

- Actel SX and MX families
- Quick logic pASIC family

4.10. The Design Flow:

This section examines the design flow for any device, whether it is an ASIC, an FPGA, or a CPLD. This is the entire process for designing a device that guarantees that you will not overlook any steps and that you will have the best chance of getting backs a working prototype that functions correctly in your system. The design flow consists of the steps in Figure 4.5.

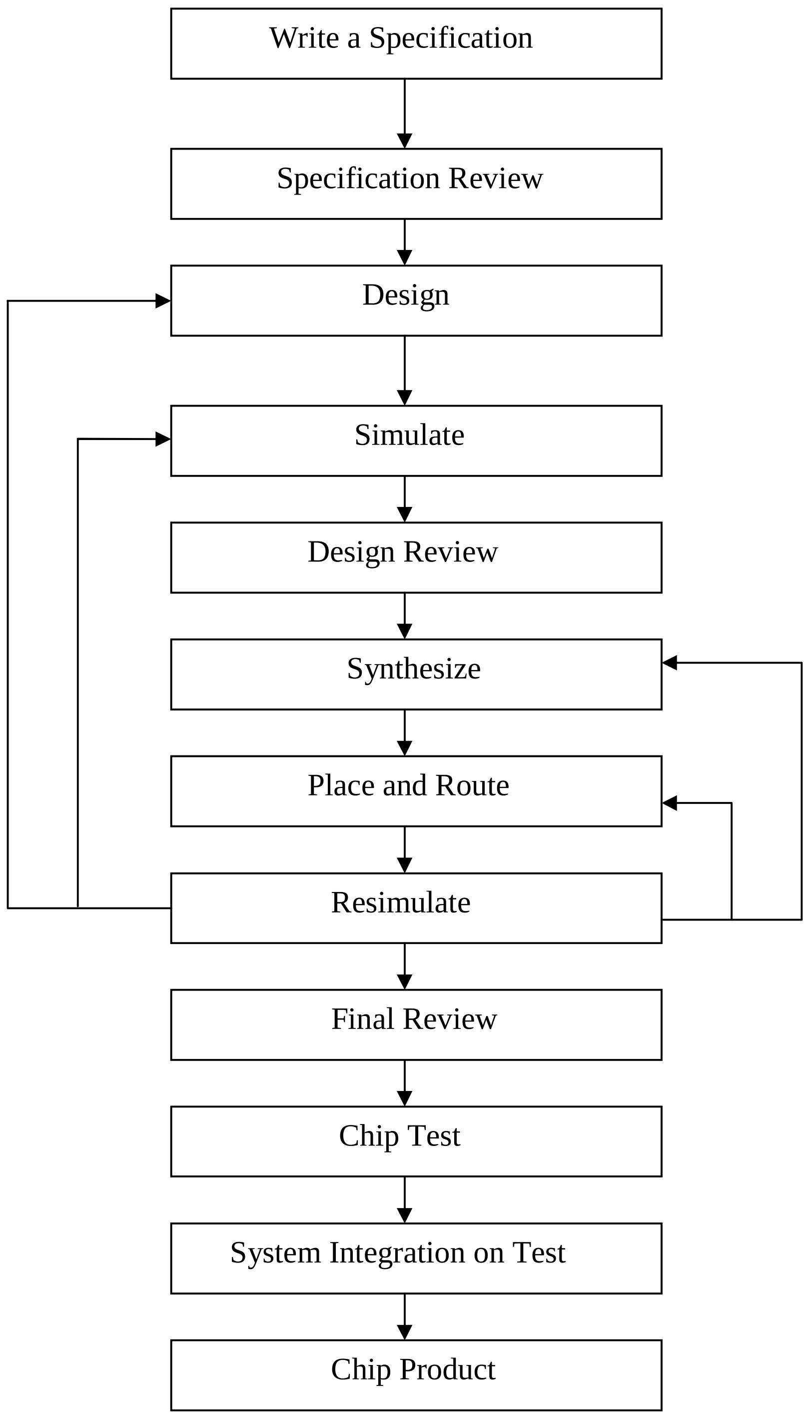


Figure 4.5: FPGA Design Flow

4.11. Writing a Specification:

The importance of a specification cannot be overstated. This is an absolute must, especially as a guide for choosing the right technology and for making your needs known to the vendor. As specification allows each engineer to understand the entire design and his or her piece of it. It allows the engineer to design the correct interface to the rest of the pieces of the chip. It also saves time and misunderstanding. There is no excuse for not having a specification.

A specification should include the following information:

- ■ An external block diagram showing how the chip fits into the system.
- ■ An internal block diagram showing each major functional section.
- ■ A description of the I/O pins including
- ■ output drive capability
- ■ input threshold level

➢ Timing estimates including

- ■ Setup and hold times for input pins
- ■ Propagation times for output pins
- ■ Clock cycle time
- ■ Estimated gate count
- ■ Package type
- ■ Target power consumption
- ■ Target price
- ■ Test procedures

It is also very important to understand that this is a living document. Many sections will have best guesses in them, but these will change as the chip is being designed.

4.11.1 Choosing a Technology:

Once a specification has been written, it can be used to find the best vendor with a technology and price structure that best meets your requirements.

4.11.2 Choosing a Design Entry Method:

You must decide at this point which design entry method you prefer. For smaller chips, schematic entry is often the method of choice, especially if the design engineer is already familiar with the tools. For larger designs, however, a hardware description language (HDL) such as Verilog or VHDL is used because of its portability, flexibility, and readability. When using a high level language, synthesis software will be required to “synthesize” the design. This means that the software creates low level gates from the high level description.

4.11.3 Choosing a Synthesis Tool:

You must decide at this point which synthesis software you will be using if you plan to design the FPGA with an HDL. This is important since each synthesis tool has recommended or mandatory methods of designing hardware so that it can correctly perform synthesis. It will be necessary to know these methods up front so that sections of the chip will not need to be redesigned later on. At the end of this phase it is very important to have a design review. All appropriate personnel should review the decisions to be certain that the specification is correct, and that the correct technology and design entry method have been chosen.

4.11.4 Designing the Chip:

It is very important to follow good design practices. This means taking into account the following design issues that we discuss in detail later in this chapter.

- Top-down design
- Use logic that fits well with the architecture of the device you have chosen
- Macros
- Synchronous design
- Protect against metastability
- Avoid floating nodes
- Avoid bus contention

4.11.5 Simulating - design review:

Simulation is an ongoing process while the design is being done. Small sections of the design should be simulated separately before hooking them up to larger sections. There will be much iteration of design and simulation in order to get the correct functionality. Once design and simulation are finished, another design review must take place so that the design can be checked. It is important to get others to look over the simulations and make sure that nothing was missed and that no improper assumption was made. This is one of the most important reviews because it is only with correct and complete simulation that you will know that your chip will work correctly in your system.

4.11.6 Synthesis:

If the design was entered using an HDL, the next step is to synthesize the chip. This involves using synthesis software to optimally translate your register transfer level (RTL) design into a gate level design that can be mapped to logic blocks in the FPGA. This may involve specifying switches and optimization criteria in the HDL code, or playing with parameters of the synthesis software in order to insure good timing and utilization.

4.11.7 Place and Route:

The next step is to lay out the chip, resulting in a real physical design for a real chip. This involves using the vendor's software tools to optimize the programming of the chip to implement the design. Then the design is programmed into the chip.

4.11.8 Re-simulating - final review:

After layout, the chip must be re-simulated with the new timing numbers produced by the actual layout. If everything has gone well up to this point, the new simulation results will agree with the predicted results. Otherwise, there are three possible paths to go in the design flow. If the problems encountered here are significant, sections of the FPGA may need to be redesigned. If there are simply some marginal timing paths or the design is slightly larger than the FPGA, it may be necessary to perform another synthesis with better constraints or simply another place and route with better constraints. At this point, a final review is necessary to confirm that nothing has been overlooked.

4.11.9 Testing:

For a programmable device, you simply program the device and immediately have your prototypes. You then have the responsibility to place these prototypes in your system and determine that the entire system actually works correctly. If you have followed the procedure up to this point, chances are very good that your system will perform correctly with only minor problems. These problems can often be worked around by modifying the system or changing the system software. These problems need to be tested and documented so that they can be fixed on the next revision of the chip. System integration and system testing is necessary at this point to insure that all parts of the system work correctly together. When the chips are put into production, it is necessary to have some sort of burn-in test of your system that continually tests your system over some long amount of time. If a chip has been designed correctly, it will only fail because of electrical or mechanical problems that will usually show up with this kind of stress testing.

4.12 Design Issues:

In the next sections of this chapter, we will discuss those areas that are unique to FPGA design or that are particularly critical to these devices.

4.12.1 Top-Down Design:

Top-down design is the design method whereby high level functions are defined first, and the lower level implementation details are filled in later. A schematic can be viewed as a hierarchical tree as shown in Figure 4.6. The top-level block represents the entire chip. Each lower level block represents major functions of the chip. Intermediate level blocks may contain smaller functionality blocks combined with gate-level logic. The bottom level contains only gates and macro functions which are vendor-supplied high level functions. Fortunately, schematic capture software and hardware description languages used for chip design easily allows use of the top-down design methodology.

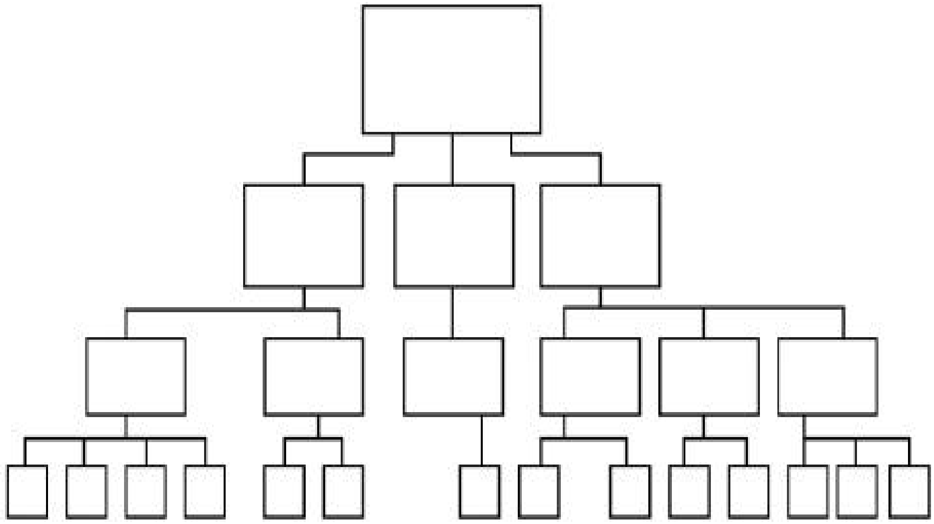


Figure 4.6: Top-Down Design

Top-down design is the preferred methodology for chip design for several reasons. First, chips often incorporate a large number of gates and a very high level of functionality. This methodology simplifies the design task and allows more than one engineer, when necessary, to design the chip. Second, it allows flexibility in the design. Sections can be removed and replaced with a higher-performance or optimized designs without affecting other sections of the chip. Also important is the fact that simulation is much simplified using this design methodology. Simulation is an extremely important consideration in chip design since a chip cannot be blue-wired after production. For this reason, simulation must be done extensively before the chip is sent for fabrication. A top-down design approach allows each module to be simulated independently from the rest of the design [25]. This is important for complex designs where an entire design can take weeks to simulate and days to debug. Simulation is discussed in more detail later in this chapter.

4.12.2 Keep the Architecture in Mind:

Look at the particular architecture to determine which logic devices fit best into it. The vendor may be able to offer advice about this. Many synthesis packages can target their results to a specific FPGA or CPLD family from a specific vendor, taking advantage of the architecture to provide you with faster, more optimal designs.

4.12.3 Synchronous Design:

One of the most important concepts in chip design, and one of the hardest to enforce on novice chip designers, is that of synchronous design. Once a chip designer uncovers a problem due to asynchronous design and attempts to fix it, he or she usually becomes an evangelical convert to synchronous design. This is because asynchronous design problems are due to marginal timing problems that may appear intermittently, or may appear only when the vendor changes its semiconductor process. Asynchronous designs that work for years in one process may suddenly fail when the chip is manufactured using a newer process. Synchronous design simply means that all data is passed through combinatorial logic and flip-flops that are synchronized to a single clock. Delay is always controlled by flip-flops, not combinatorial logic. No signal that is generated by combinatorial logic can be fed back to the same group of combinatorial logic without first going through a synchronizing flip-flop. Clocks cannot be gated - in other words, clocks must go directly to the clock inputs of the flip-flops without going through any combinatorial logic. The following sections cover common asynchronous design problems and how to fix them using synchronous logic.

4.13 Race conditions:

Figure 4.7 shows an asynchronous race condition where a clock signal is used to reset a flip-flop. When SIG2 is low, the flip-flop is reset to a low state. On the rising edge of SIG2, the designer wants the output to change to the high state of SIG1. Unfortunately, since we don't know the exact internal timing of the flip-flop or the routing delay of the signal to the clock versus the reset input, we cannot know which signal will arrive first - the clock or the reset. This is a race condition. If the clock rising edge appears first, the output will remain low. If the reset signal appears first, the output will go high. A slight change in temperature, voltage, or process may cause a chip that works correctly to suddenly work incorrectly. A more reliable synchronous solution is shown in Figure 4.8. Here a faster clock is used, and the flip-flop is reset on the rising edge of the clock. This circuit performs the same function, but as long as SIG1 and SIG2 are produced synchronously - they change only after the rising edge of CLK - there is no race condition.

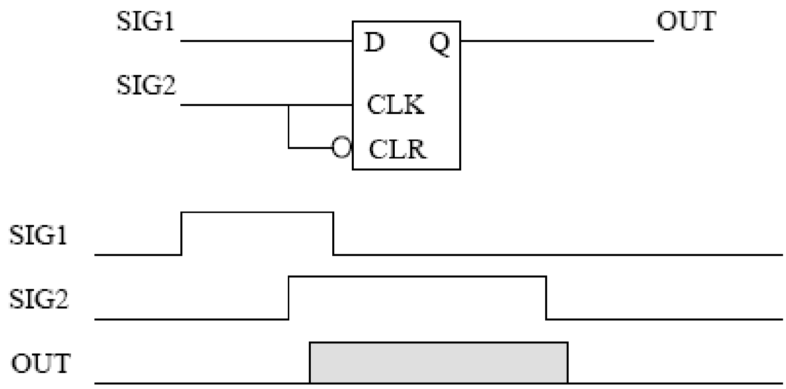


Figure 4.7: Asynchronous: Race Condition

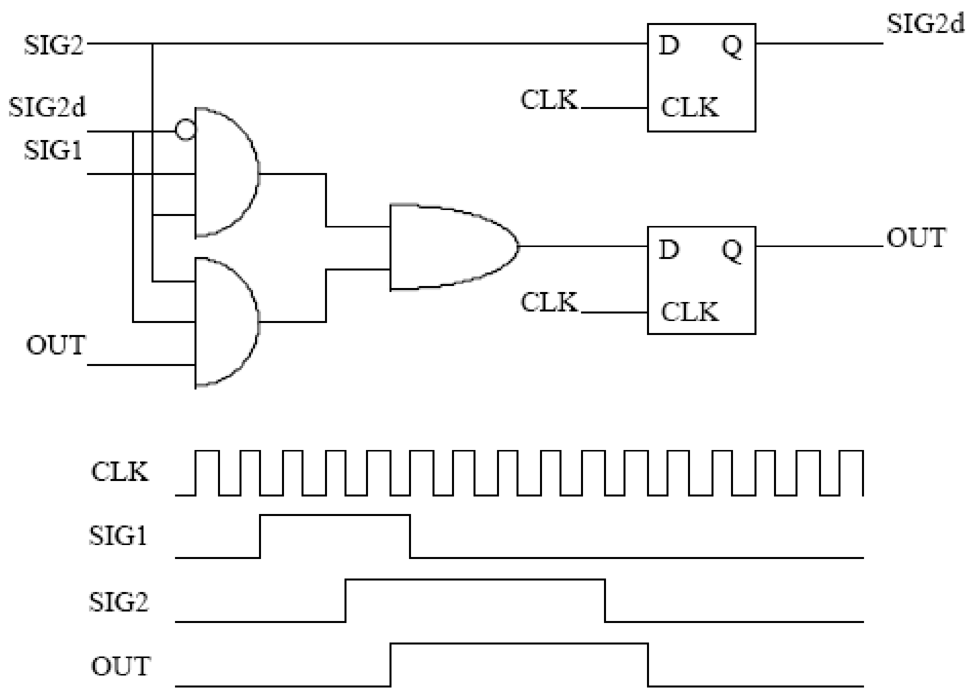


Figure 4.8: Synchronous: No Race Condition

4.14 Delay dependent logic:

Figure 4.9 shows logic used to create a pulse. The pulse width depends very explicitly on the delay of the individual logic gates. If the process should change, making the delay shorter, the pulse width will shorten also, to the point where the logic that it feeds may not recognize it at all. A synchronous pulse generator is shown in Figure 4.10. This pulse depends only on the clock period. Changes to the process will not cause any significant change in the pulse width.

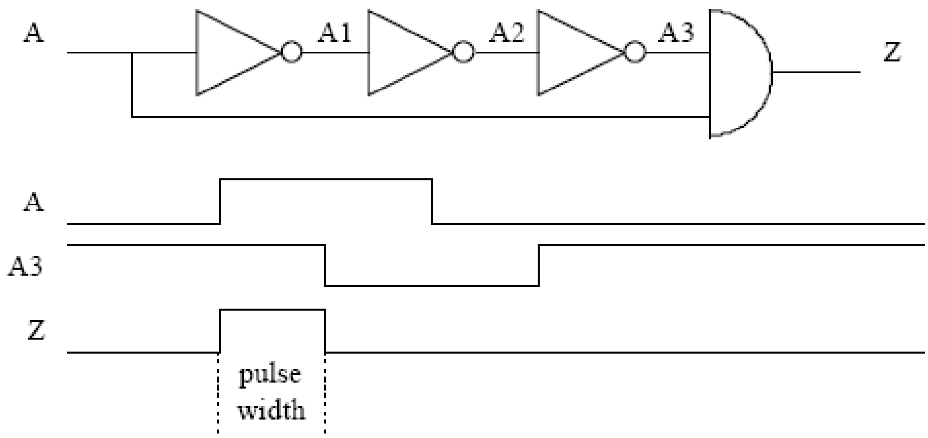


Figure 4.9: Asynchronous: Delay Dependent Logic

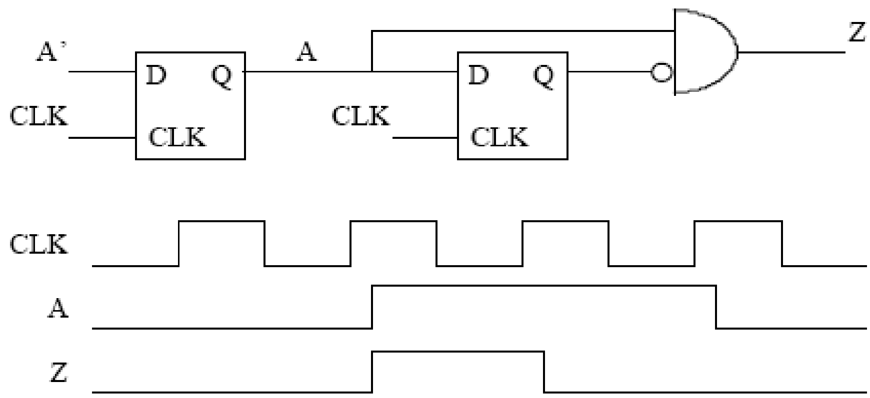


Figure 4.10: Synchronous: Delay Independent Logic

4.15 Hold time violations:

Figure 4.11 shows an asynchronous circuit with a hold time violation. Hold time violations occur when data changes around the same time as the clock edge. It is uncertain which value will be registered by the clock. The circuit in Figure 20 fixes this problem by putting both flip-flops on the same clock and using a flip-flop with an enable input. A pulse generator creates a pulse that enables the flip-flop.

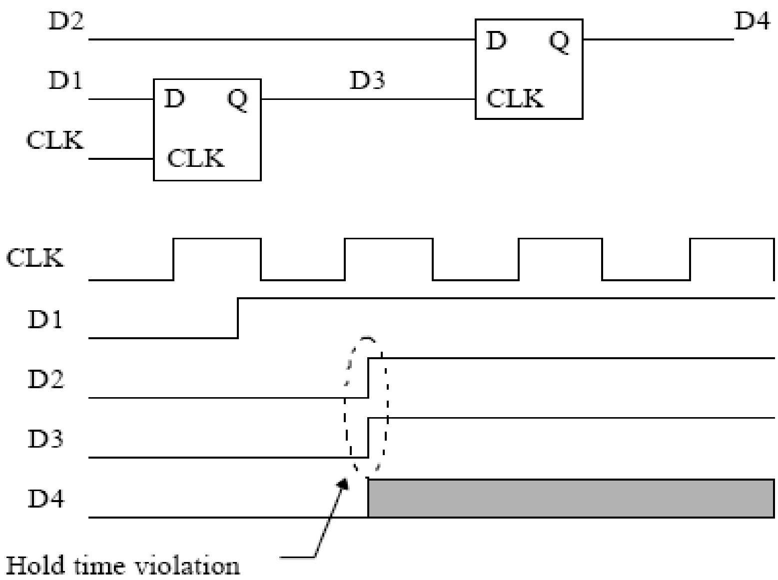


Figure 4.11: Asynchronous: Hold Time Violation

4.16 Glitches:

A glitch can occur due to small delays in a circuit such as that shown in Figure 4.12. An inverting multiplexer contains a glitch when switching between two signals, both of which are high. Yet due to the delay in the inverter, the output goes high for a very short

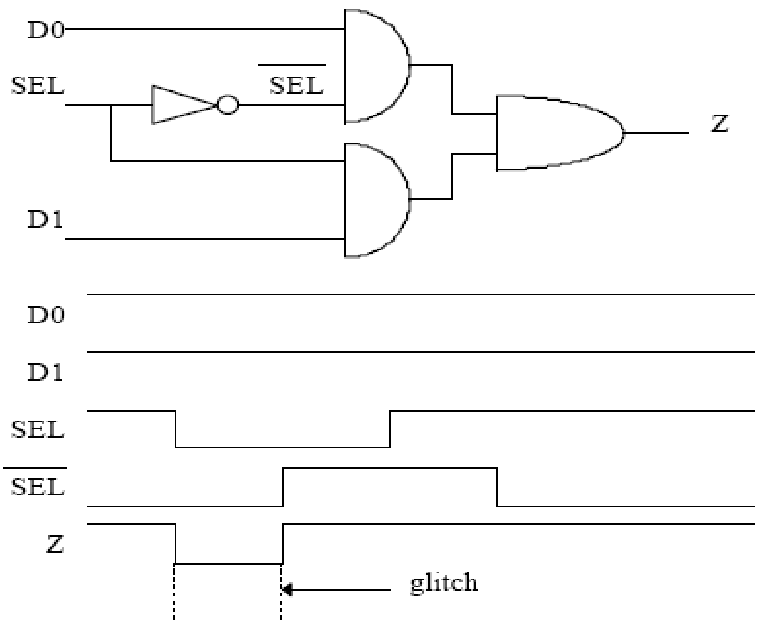


Figure 4.12: Asynchronous: Glitch

time. Synchronizing this output by sending it through a flip-flop as shown in Figure 4.13, ensures that this glitch will not appear on the output and will not affect logic further downstream.

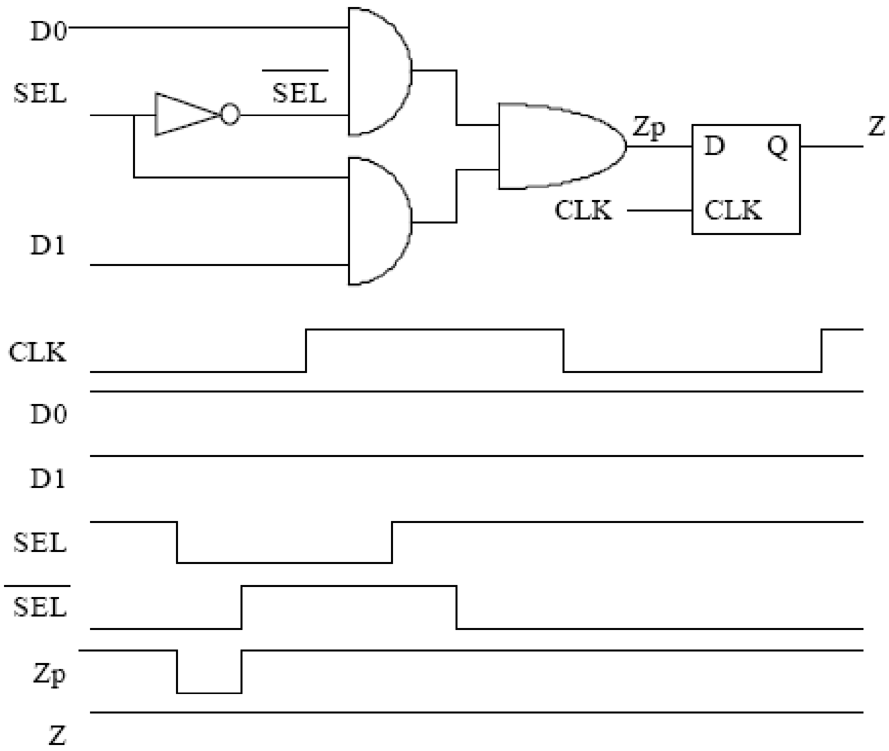


Figure 4.13: Synchronous: No Glitch

4.17 Bad clocking:

Figure 4.14 shows an example of asynchronous clocking. This kind of clocking will produce problems of the type discussed previously. The correct way to enable and disable outputs is not by putting logic on the clock input, but by putting logic on the data input as shown in Figure 4.15.

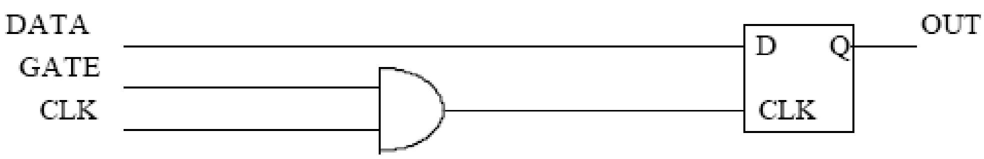


Figure 4.14: Asynchronous: Bad Clocking

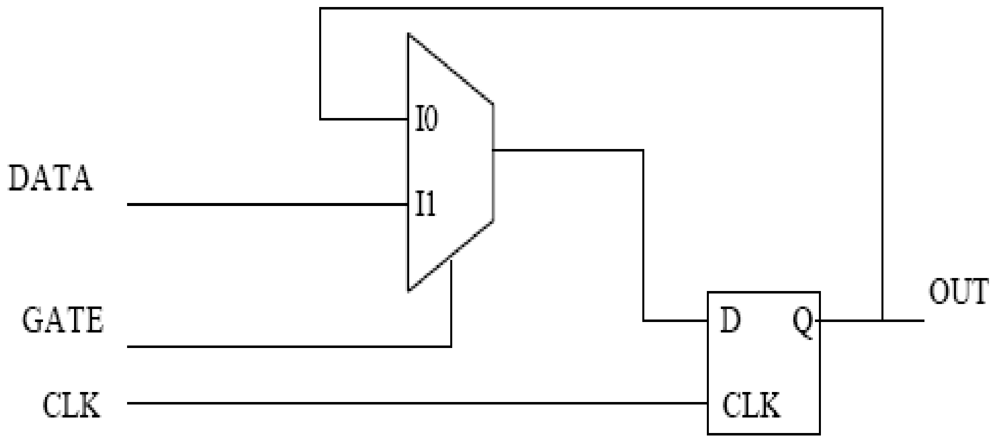


Figure 4.15: Synchronous: Good Clocking

4.18 Metastability:

One of the great buzzwords, and often misunderstood concepts, of synchronous design is metastability. Metastability refers to a condition which arises when an asynchronous signal is clocked into a synchronous flip-flop. While chip designers would prefer a completely synchronous world, the unfortunate fact is that signals coming into a chip will depend on a user pushing a button or an interrupt from a processor, or will be generated by a clock which is different from the one used by the chip. In these cases, the asynchronous signal must be synchronized to the chip clock so that it can be used by the internal circuitry. The designer must be careful how to do this in order to avoid metastability problems as shown in Figure 4.16. If the ASYNC_IN signal goes high around the same time as the clock, we have an unavoidable race condition [25]. The output of the flip-flop can actually go to an undefined voltage level that is somewhere between a logic 0 and logic 1. This is because an internal transistor did not have enough time to fully charge to the correct level. This meta level may remain until the transistor voltage leaks off or “decays”, or until the next clock cycle. During the clock cycle, the gates that are connected to the output of the flip-flop may interpret this level differently. In the figure, the upper gate sees the level as logic 1 whereas the lower gate sees it as logic 0. In normal operation, OUT1 and OUT2 should always be the same value. In this case, they are not and this could send the logic into an unexpected state from which it may never return. This metastability can permanently lock up your chip.

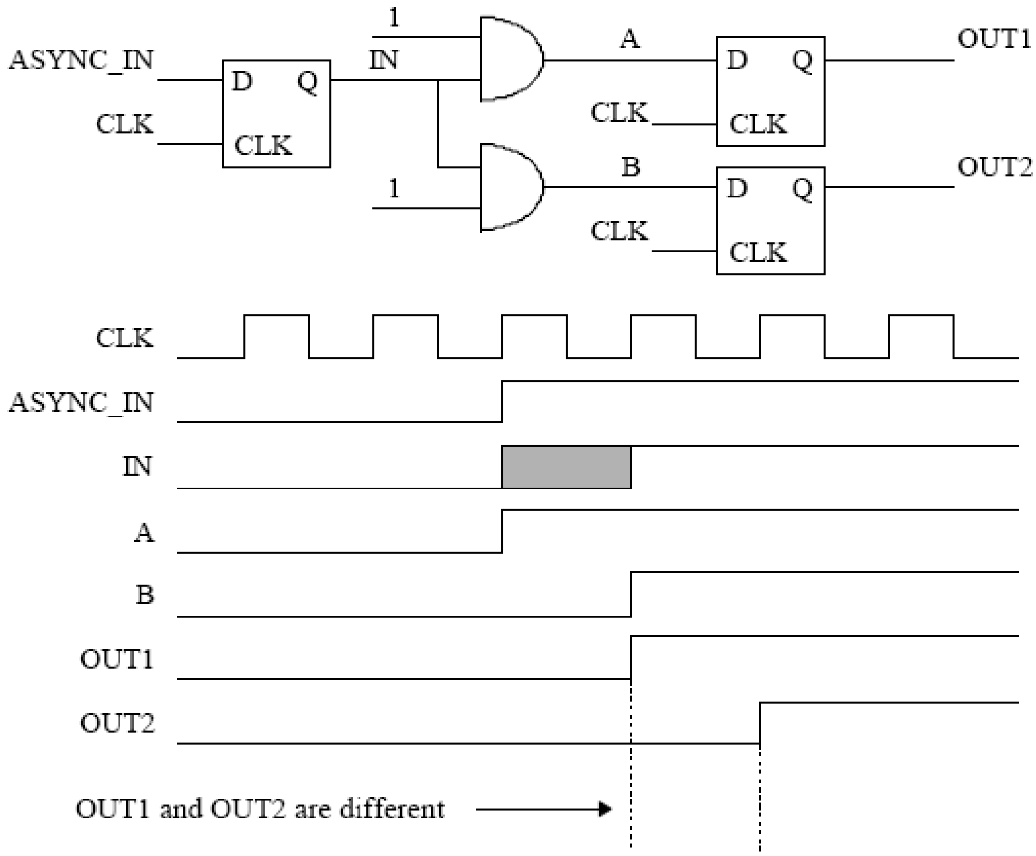


Figure 4.16: Metastability - The Problem

The “solution” to this metastability problem by placing a synchronizer flip-flop in front of the logic, the synchronized input will be sampled by only one device, the second flip-flop, and be interpreted only as logic 0 or 1. The upper and lower gates will both sample the same logic level, and the metastability problem is avoided. Or is it? The word solution is in quotation marks for a very good reason. There is a very small but non-zero probability that the output of the synchronizer flip-flop will not decay to a valid logic level within one clock period. In this case, the next flip-flop will sample an indeterminate value, and there is again a possibility that the output of that flip-flop will be indeterminate. At higher frequencies, this possibility is greater. Unfortunately, there is no certain solution to this problem. Some vendors provide special synchronizer flip-flops whose output transistors decay very quickly. Also, inserting more synchronizer flip-flops reduces the probability of metastability but it will never reduce it to zero. The correct action involves discussing metastability problems with the vendor, and including enough synchronizing flip-flops to reduce the probability so that it is unlikely to occur within the lifetime of the product.

4.19 Timing Simulation:

This method of timing analysis is growing less and less popular. It involves including timing information in a functional simulation so that the real behavior of the chip is simulated. The advantage of this kind of simulation is that timing and functional problems can be examined and corrected. Also, asynchronous designs must use this type of analysis because static timing analysis only works for synchronous designs. This is another reason for designing synchronous chips only. As chips become larger, though, this type of compute intensive simulation takes longer and longer to run. Also, simulations can miss particular transitions that result in worst case results. This means that certain long delay paths never get evaluated and a chip with timing problems can pass timing simulation. If you do need to perform timing simulation, it is important to do both worst case simulation and best case simulation. The term “best case” can be misleading. It refers to a chip that, due to voltage, temperature, and process variations, is operating faster than the typical chip. However, hold time problems become apparent only during the best case conditions.

CHAPTER FIVE

System Implementation

A system level block diagram of a digital predistortion system is shown in the following figure. The entire digital predistortion system is housed in the transmitter and requires the addition of several components to the standard transmitter components. The primary components augmenting the modulator, upconverter, and the power amplifier are:

- Complex Gain Adjust block in the baseband signal path.
- Look-Up Table for storing a set of complex coefficients.
- Demodulator and Down converter whose input is drawn from the output of the power amplifier.
- Adaptation Algorithm processing block that calculates the complex coefficients that are stored in look-up-table.

With the inclusion of digital predistortion, the digital complex baseband input signal samples are multiplied prior to the DAC by complex coefficients drawn from the look-up table. The look-up table coefficients implement the predistortion function, F. The adaptation algorithm determines the values of the coefficients by comparing the feedback signal and a delayed version of the input signal. Ideally, the feedback signal should be a scaled reproduction of the input signal (i.e. linear amplification), thus, the role of the adaptation algorithm is to derive a set of coefficients that forces the error between the scaled feedback signal and the delayed input signal to zero.

A number of critical design parameters/issues can be readily identified from the digital predistortion system shown in the preceding figure:

- Sampling rate used in the DSP portions of the transmitter
- Number of bits used for quantization by DAC and ADC
- Shape of the reconstruction filter at the output of the DAC
- Bandwidth of the low pass filter at the input of the ADC
- Stability of the feedback loop/adaptation algorithm
- Precision (fixed-point) of the look-up table and DSP processing, if any
- Memory available for the look-up table and adaptation algorithm

The intent of implementing the digital predistortion system in ADS is to be able to capture and analyze the impact on performance of some of these design parameters.

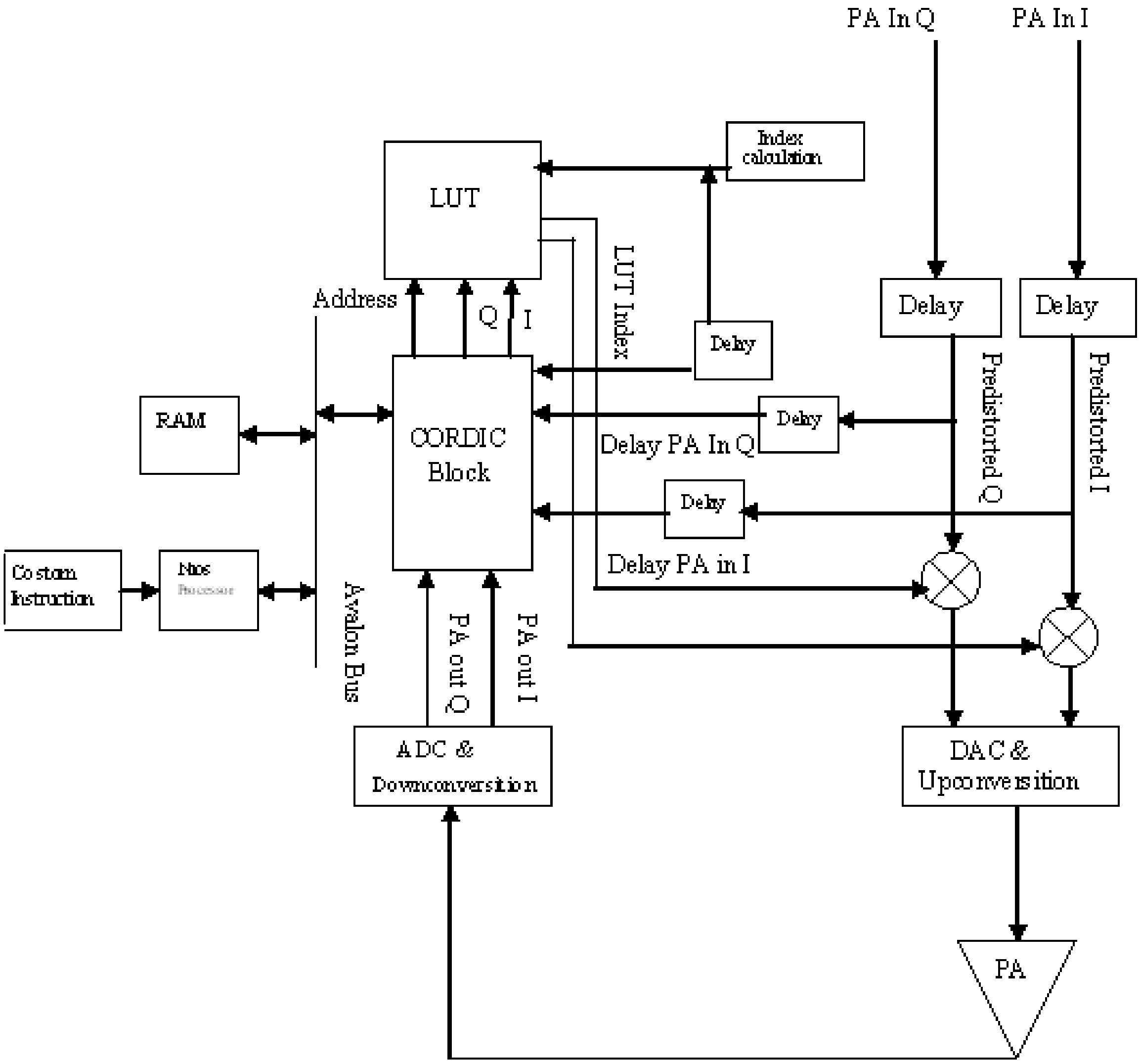


Figure 5.1: Digital predistortion system block Diagram [8]

5.1 Implementing the Predistortion Function:

The Complex Gain Adjust and Look-Up Table blocks of the preceding figure are shown in more detail in the following illustration.

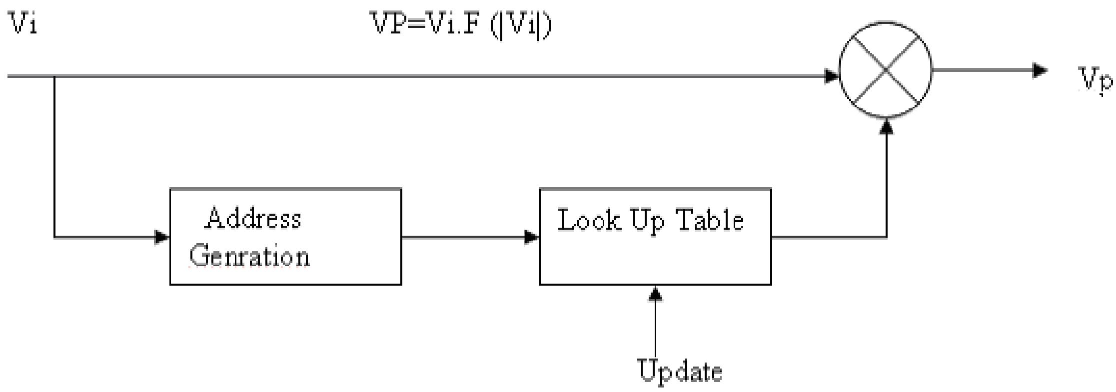


Figure 5.2: Implementing a predistortion function {27}

The predistorting function, F, is assumed to be a function of the magnitude of the input signal. The predistorting function is implemented using a complex multiplier, a Look-Up Table and an Address Generation block that selects the appropriate coefficient from the look-up table, given the magnitude of the input signal. The coefficients stored in the look-up table are the value of the predistortion function at certain input signal magnitudes. Thus, the predistortion function is not implemented in an analytic manner; rather, it is only calculated at a specified number of points. The size of the look-up table employed determines the number of points at which the predistortion function is calculated. In addition, the distribution of the predistortion function points need not necessarily be evenly distributed across the range of the input signal magnitude. Instead, it may be desirable to distribute the predistortion function points across the range of the input signal magnitude using a squared (power) or logarithmic relationship [27]. The following plots illustrate how the predistortion function may be indexed in the look-up table by the magnitude of the input signal and by the power of the input signal. Both plots show the same information, however, the scale of the x-axis is in dBm in the left hand plot and in Volts in the right-hand plot.

For look-up table entries, the plots show the cumulative distribution of the entries over the range of the input signal magnitude for the magnitude and power distributions. Overlayed upon the distributions is the amplifier response, illustrating where the entries fall along the amplifier response. The last entry in both distributions falls at the maximum input power level that can be linearized (approximately 0.4 dBm). The look-up table entries are equally spaced over the range of the magnitude of the input signal in the case of magnitude indexing, whereas, more look-up table entries are distributed at the higher end of the range in the case of power indexing.

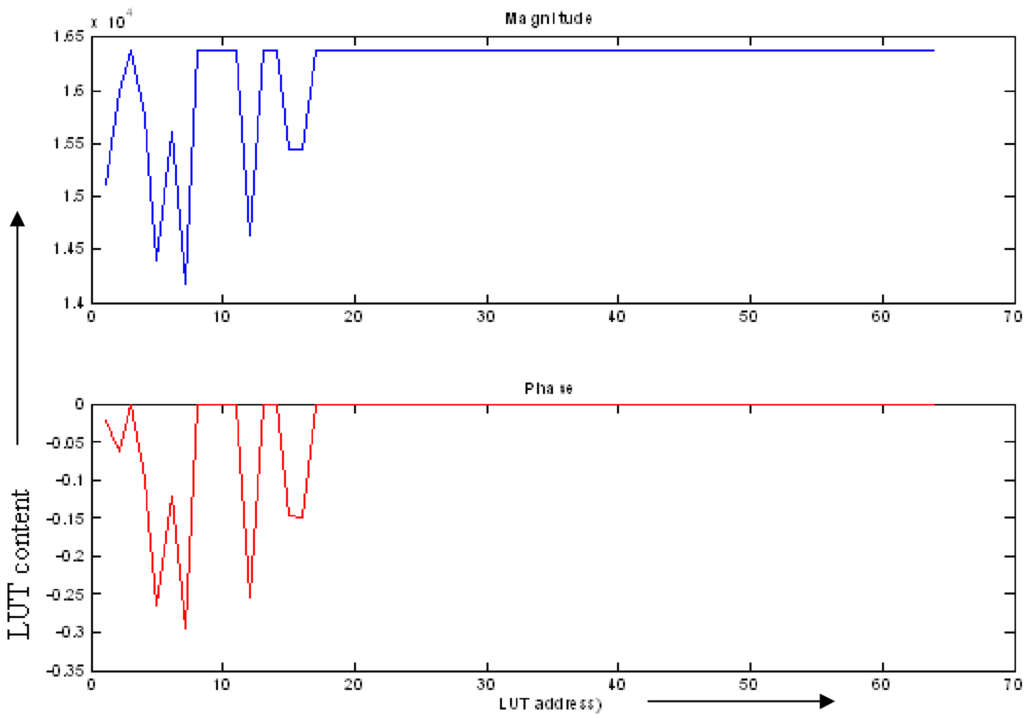


Figure 5.3: Magnitude and phase of the LUT address-content plot

5.2 Adaptation Algorithm:

The function of the adaptation algorithm is to derive the predistortion function, F, i.e. the inverse characteristic of the amplifier response. The predistortion function may be derived using either a modulated signal input (random signal) or a known training signal input. The adaptation algorithm and its implementation are fundamentally different depending upon which type of input signal is utilized. The algorithms that are based upon the use of a modulated signal employ statistical signal processing and, typically, some type of curve fitting algorithm to generate a smooth predistortion function. The complexity of the adaptation algorithm and its implementation can be significantly simplified, however, by using the alternative input signal - a known training signal.

5.3 Training Signal:

The adaptation algorithm chosen for implementation is based upon the use of a training signal. The training signal is a single tone having a frequency equal to the carrier frequency and whose power is ramped up over the duration of the training period. The power of the tone is set to zero at the start of the training period and will typically peak at, or just below, the maximum correctable input power of the amplifier. The use of a single tone whose power is ramped as a training signal greatly simplifies the adaptation algorithm and its implementation. However, the use of the training signal does require that the modulated signal being transmitted be interrupted while the training signal is transmitted. In addition, because the training signal is a single tone, the digital predistorter is only correcting for the operation of the amplifier at a single frequency, not across the entire transmission bandwidth. If the amplifier's pass-band is quite flat, the use of a single tone training signal in this manner will enable the predistortion function to be determined accurately. In any event, the use of a single complex coefficient to correct for distortion of the amplifier at a particular power level presupposes that any amplifier memory effects are minimal, i.e. the amplifier has a flat pass-band [27].

5.4 Implementation:

The following illustration is a block diagram of the adaptation algorithm that is employed. The algorithm is based upon the determination of the open loop gain, H, of the predistorter and amplifier combination at the power level associated with each look-up table entry.

Recall that the desired linear response of the predistorter and amplifier cascade requires that F(|Vi|) G(|Vp|) = k for all inputs. Hence, if Glin is set to be equal to k, the desired open loop gain of the system is unity. If the calculated open loop gain is not equal to unity, the predistortion function must be adjusted in a manner to drive the open loop towards unity. This is can be achieved as illustrated in the following manner.

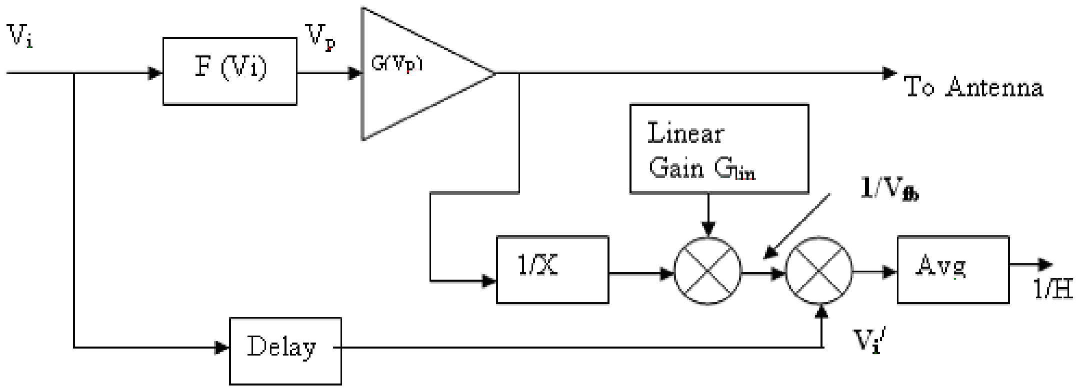


Figure 5.4: Implementation of the Adaptation Algorithm [27]

$$\mathbf{1/H= V_i^{/} / V_{fb} = G_{lin} / F(|Vi|).G(|Vp|) ===> 1+J*0}$$

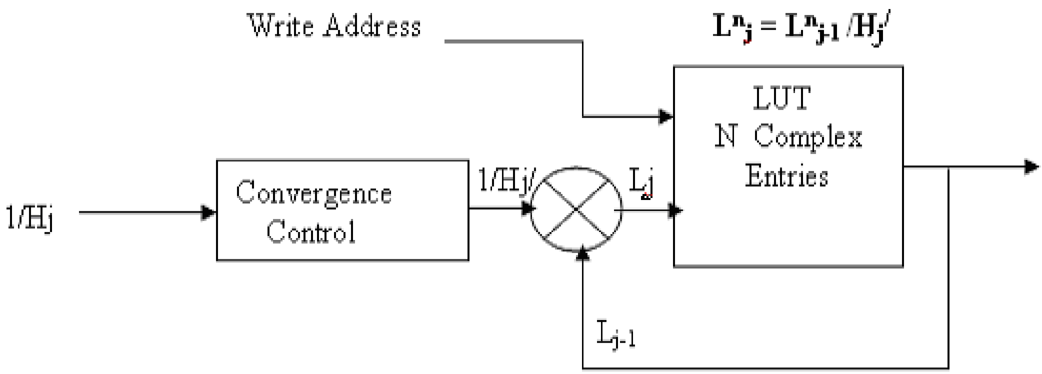


Figure 5.5: Calculation of New Predistortion Function [27]

The predistortion function is defined by a set of coefficients stored in the look-up table, Ln, where each n corresponds to an input signal magnitude which is mapped to a look-up table address. In order to drive the open loop gain to unity, the predistortion function coefficients are updated by dividing each coefficient by the calculated open loop gain (or by the calculated open loop gain adjusted to slow the rate at which the coefficients will change).

5.5 Synchronization:

The accuracy of the open loop gain calculation is dependent upon the accuracy of the estimation of the delay in the feedback path. As shown in figure 5.4, the input signal, Vi, must be delayed precisely by an amount equal to the delay in the feedback path. The delay in the feedback path is estimated by calculating the correlation between the magnitude of the input signal and the magnitude of the feedback signal. The use of the magnitude of the signals has the benefit of not requiring phase synchronization in the feedback path. Because the delay in the feedback path will not necessarily be equal to an integer number of DSP sample periods, interpolation is employed to more precisely align the input and feedback signals [27]. The correlation between the input and feedback signal is performed on a modulated signal that precedes the training signal because the gain compression of the amplifier makes the accuracy of the correlation over the training signal suspect. In addition, because the envelope of the modulated signal will typically have a PDF such that it spends much of its time within the linear operating region of the amplifier, correlation using the modulated signal becomes more reliable. However, because the modulated signal is stochastic, the statistics of the modulated signal, as well as the size of the data block over which the correlation operation is performed will impact the accuracy of the delay estimation. In general, the accuracy of the estimation improves as the block size increases. Unfortunately, a larger block size requires more memory and takes longer to perform the estimate.

5.6 Adaptive Coefficient Update Using Embedded Processing:

In many typical applications the PA characteristics do not change rapidly with time. The PA characteristics vary as a function of temperature drift and component aging, parameters that have long time-constants. The previous section described a predistorter design that employed a dedicated customized datapath constructed using the logic fabric and embedded multipliers. To implement the DPD coefficient update. Depending on system requirements, and in particular the required rate of coefficient adoption, an FPGA embedded processor could be employed to realize the update. In this approach a buffer of the samples y (n) and z (n) in Figure 3.12 are prepared and processed offline. This lowers the overall implementation requirements of the system. State-of-the-art FPGAs like Virtex-II Pro and Virtex-4 include embedded Power PC 405 (PPC405) processing cores. The adaptive algorithm can be coded in C and executed on the PPC405 [1]. When a new coefficient vector is available the PPC405 can transfer this data to the coefficient memory in the Volterra filter. The PPC could also be used for other tasks in the system, in addition to periodically servicing the DPD processor .

5.7 CORDIC Block:

The CORDIC block is used to convert from polar to Cartesian and from Cartesian to polar. CORDIC is hardware efficient algorithm that allows trigonometric function to be performing using only shifts and additions.

5.7.1 Operation of CORDIC Block:

1. Cartesian PA input and output values are fed in to the CORDIC block.
2. The polar PA input and output values are read from the CORDIC block by the NIOS processor.
3. The NIOS processor implements the algorithm to calculate the new LUT values in polar form.
4. The NIOS processor writes this LUT value to CORDIC block.
5. The CORDIC block converts the LUT values in to Cartesian form and then writes it to LUT.

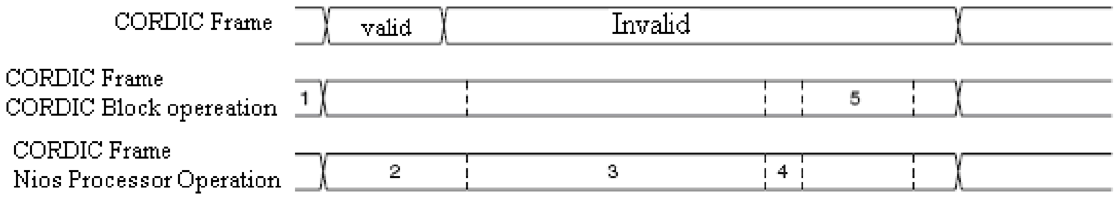


Figure 5.6: shows the synchronization between Nios and CORDIC [8]

5.7.2 Nios Processor:

1. The Nios Processor communicates with the CORDIC block bia Avalon bus the Nios Processor reads PA inputs and outputs in polar forms from CORDIC through Avalon. Using these values the Nios Processor implements the algorithm and calculates the new LUT value in polar form. Now these values are written in CORDIC by Nios Processor and converted to Cartesian form.
2. Read the following remaining polar form data from the output of the CORDIC block to be used this cycle.
   A. PA input phase
   B. PA output magnitude
   C. PA output phase
   D. LUT index.
3. Compute the error in magnitude and phase between PA output and PA input and pass through the DPD algorithm to compute the new LUT value around the given index.
4. Update the NIOS private polar form LUT with new LUT value.
5. Compares the gain of CORDIC by multiplying the magnitude component of each new LUT entry.

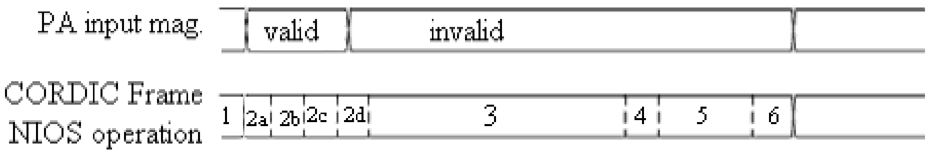


Figure 5.7: CORDIC frame and Nios operation

CHAPTER SIX

Design Walkthrough

6.1 Introduction:

This chapter depicts all the design steps involves to implement the Digital Predistortion Design. The design walkthrough involves the following steps,

A. Parameterize the design
B. Auto-generate Verilog HDL simulation files from the MATLAB.
C. Build the SOPC builder project.
D. Simulate the design.
E. Synthesize the design

6.2 Parameterize the Design:

To parameterize the design or examine the Digital Predistortion Design's performance, perform the following steps:

A. Start the MATLAB
B. Type the following command: dpd_gui

1. The output message is still output to the MATLAB command window. The Digital Predistortion GUI opens,

A. Upon starting MATLAB and whenever the design configuration turned on, the algorithm configuration section has all appropriate setting for the default design configuration.
B. If turned off, then it can only work in MATLAB domain; the RTL will not match the MATLAB model.
C. In the algorithm drop down box, choose algorithm type. For all algorithms enter LUT size 1 and learn factor.

2. Another important thing, the bit width must be specified for the entire signal used in the design. The bit widths description for the following signals is given bellow;

A. PA IQ signals: The number of bits in PA input and output signals
B. LUT IQ signals: The number of bits in LUT values
C. Phase signals: The number of bits for phase

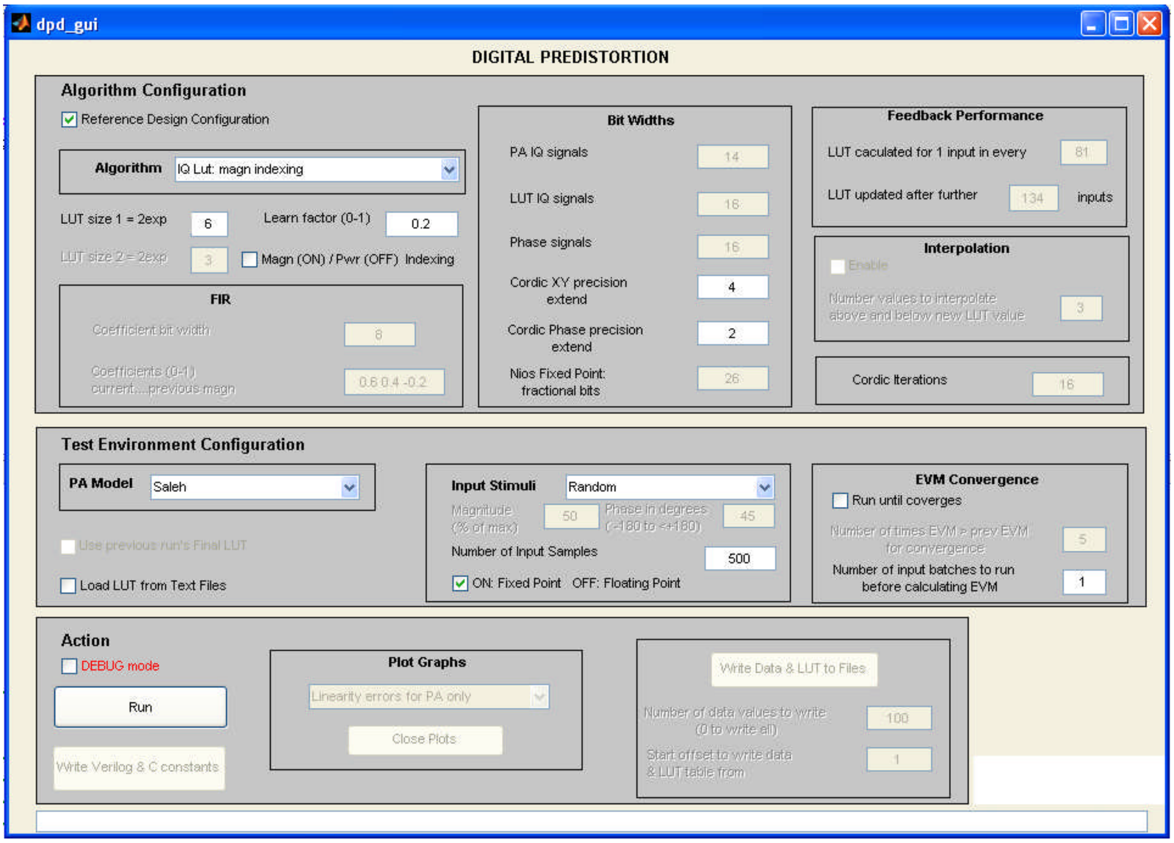


Figure 6.1: MATLAB Graphical Used Interface DPD

D. CORDIC XY precision extend: The number of bits to extend XY vectors used in CORDIC block, which improves the accuracy of results Polar or Cartesian value calculation

E. CORDIC phase precision extend: The number of bits to extend phase signal used in CORDIC block, which improves the accuracy of the result of Cartesian

F. Nios fixed point fractional bits: The Nios processor uses 32-bits fixed point format. This field determines the number of fractional bits.

3. In the feedback performance section enter the following values to determine the modeling performance of feed back path; input signals are applied every clock cycle, yet it takes a

finite number of clock cycle for the feed back path to process one set of values to calculate a LUT value, before being able to process the next set of values.

A. Enter the delay for the Nios to calculate the next LUT value in terms of how many input samples are surpass.
B. Also there is a delay between Nios calculating the new LUT values and the LUT being updated with those values.

4. In the test environment configuration part of the Digital Predistortion GUI, perform the following steps:

A. In the Power amplifier model drop down box, choose the PA model. For this design the Saleh model is used.
B. To load the LUT from text file turn on **Load from text file.**
C. Click **Run**, to run the simulation. And after selecting the different type of graph from drop down box, different plot are visible.

6.3 Auto-generate Design configuration and Test data for Verilog HDL simulation from MATLAB:

Once MATLAB testing reaches a conclusion with the configuration, now the time to auto generate Test Data for Verilog simulation from MATLAB. This involves the following steps:

A. Number of data values which is to write is entering first.
B. A offset value must be enter, if the file is going to be use file for RTL simulation 0 must be entered.
C. Click *Write Data & LUT to Files*.
D. Click *Write Verilog & C Constant.*

6.4 Build the SOPC builder project:

To build the SOPC builder project involves the following steps:

A. Run the Perl Script
B. Open Quartus || project
C. Launch SOPC builder

D. Generate the system

Running the Perl script, first creates a directory and after that copies the required files in to that directory. To run the Perl script the following steps are involves:

A. Open the command prompt

B. Use cd command to change directory

C. Type the *perl gen_dpd_frame.pl –rel*, **press** enter key.

As it runs, the Perl script informs that files are being copied. Now the Quartus || software will be launch. After launching this *Open project* option must be choose, then choose a ***.quartus*** file from the directory made by the Perl script. After following these steps a top level input output schematic will be appear.

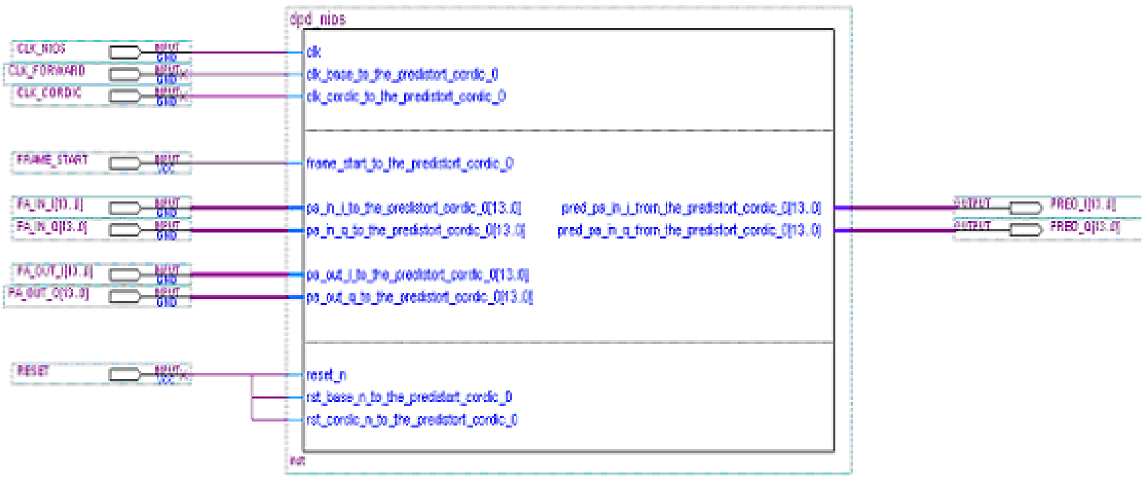


Figure 6.2: Digital Predistortion design top level diagram

6.5 Launch SOPC Builder:

SOPC builder can be launched by double click on the **dpd_nios** symbol with in the top level schematic. The SOPC builder system content page for the design will appear. Now to simulate the design,

A. Click the *System Generation* tab in SOPC builder

B. Click *Generate*

C. Click *Run ModelSim*, to run the ModelSim software.

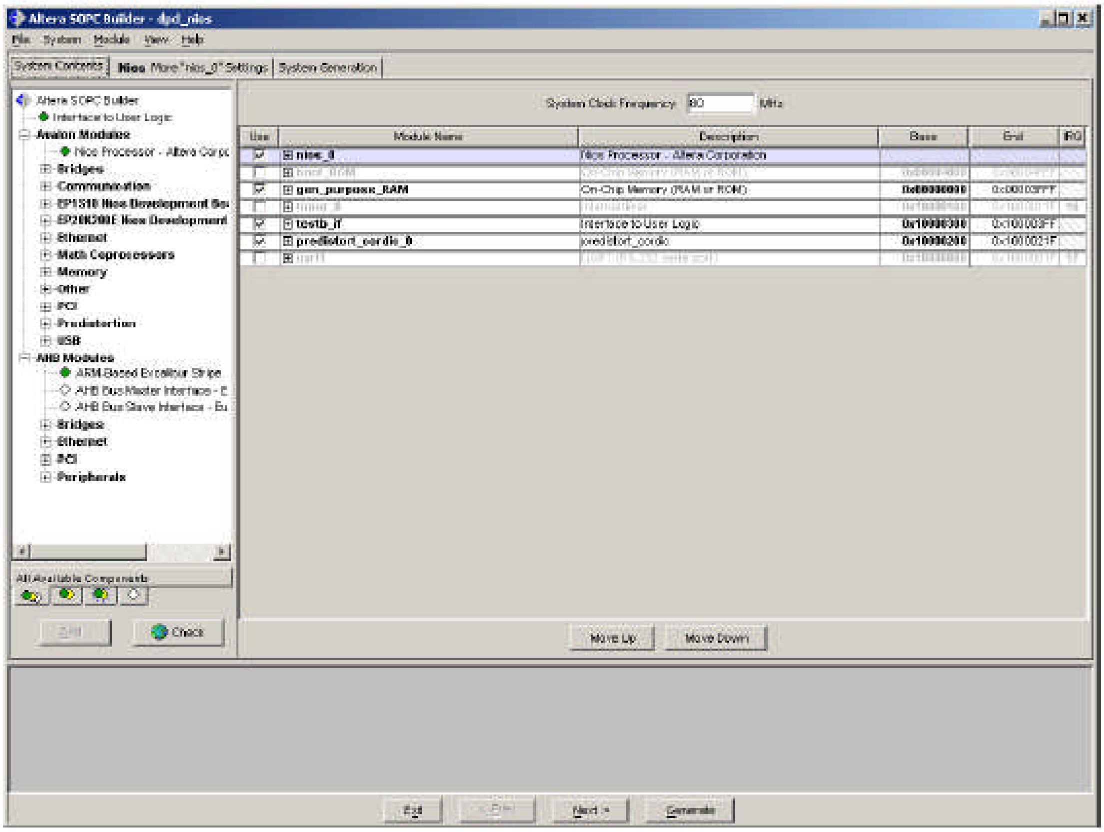


Figure 6.3: SOPC builder system content page

There must be providing a correct path to ModelSim software choosing *SOPC builder setup* option. In this chapter, all the processing steps are provided by having an objective to use the design successfully.

CHAPTER SEVEN

Results and Discussion

7.1 Results:

1. **Error Vector Magnitude:**

    - EVM for non-predistortion system = 108.7263%,
    - EVM for Predistortion system EVM = 102.6331%

2. **Average Error Vector Magnitude:**

    - Average EVM for non-predistortion system = 108.7263%,
    - Average EVM for predistortion system EVM = 102.6331%
    - EVM reduction over non_predistortion = 5.604%

3. **Mean values over last 500 input samples: Non-Predistorted system:**

    - Absolute Phase Error = 3421.21545
    - Absolute Magnitude Error = 1049.64671

4. **Mean values over last 500 input samples: Predistorted system:**

    - Absolute Phase Error = 3077.30933
    - Absolute Magnitude Error = 1007.00127

5. **Predistorted reduction compared to non-predistortion:**

    - Mean Absolute Phase Error reduction = 10.052%
    - Mean Absolute Magnitude Error reduction = 4.063%

6. Predistortion: 3rd order:

- Average sideband magnitude = -55.360dB,
- Max sideband magnitude = -28.364dB

7. Predistortion: 5th order:

- Average sideband magnitude = -70.594dB,
- Max sideband magnitude = -47.468dB

8. No Predistortion 3rd order:

- Average sideband magnitude = -57.124dB,
- Max sideband magnitude = -32.873dB

9. **No Predistortion: 5th order**:

- Average sideband magnitude = -84.975dB,
- Max sideband magnitude = -55.535dB

7.2 Simulation Results:

7.2.1 Linearity Error for PA:

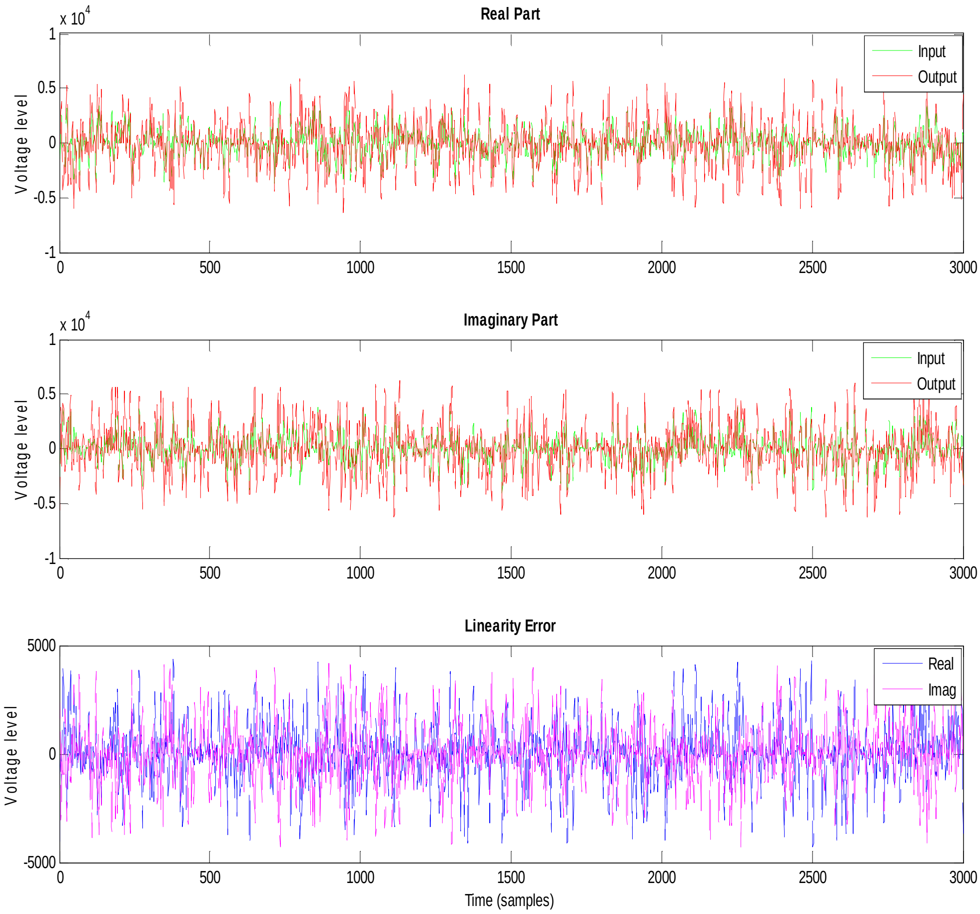


Figure 7.1: linearity error for PA only (no predistortion)

7.2.2 Normalized Linearity error for PA:

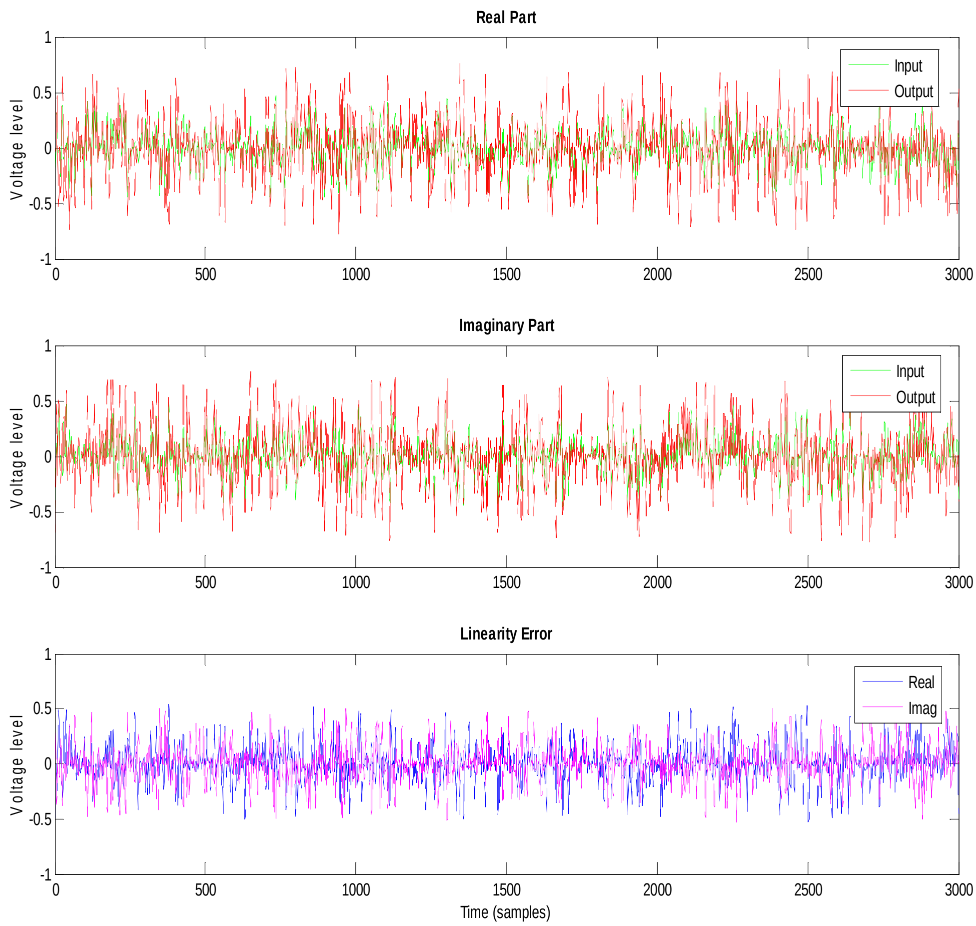


Figure 7.2: Normalized linearity error for PA only (no predistortion)

7.2.3. Linearity error for DPD system:

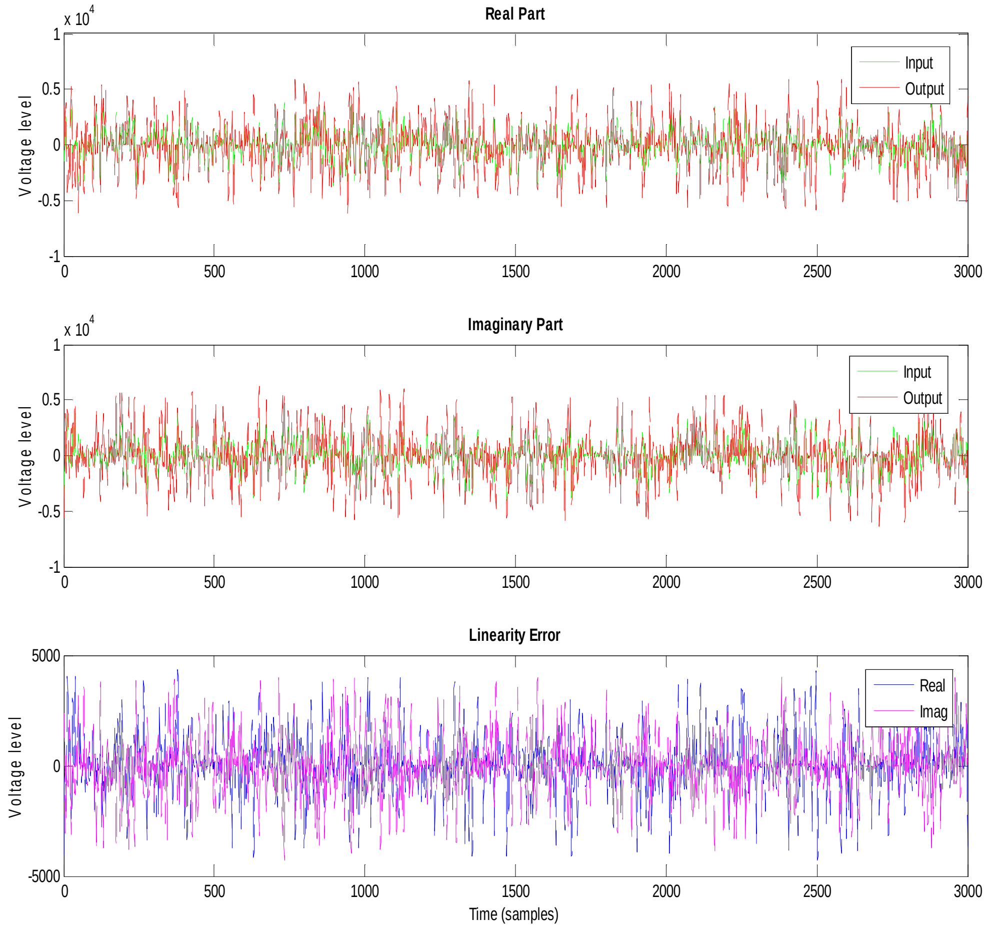


Figure 7.3: linearity error for DPD system (Predistorted PA behavior)

7.2.4. Normalized linearity error for DPD system:

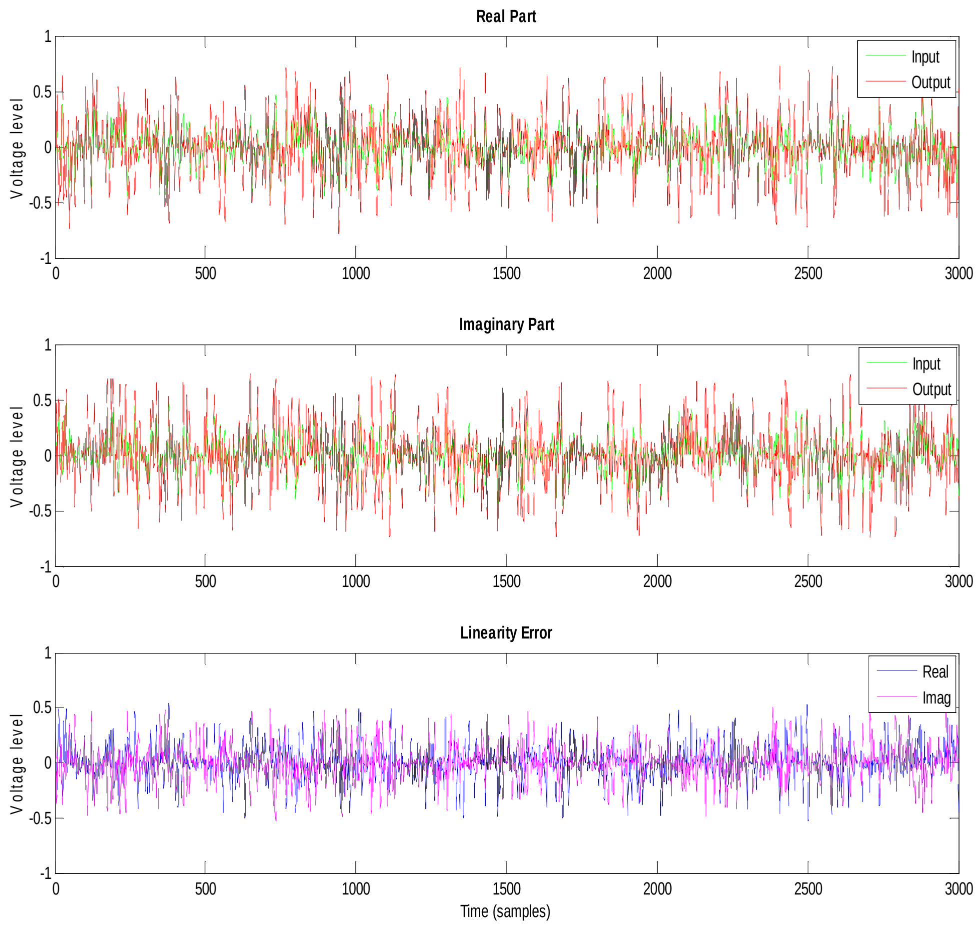


Figure 7.4: Normalized linearity error for DPD system (Predistorted PA behavior)

7.2.5. Error comparison between Pridistorted and non Predistorted system:

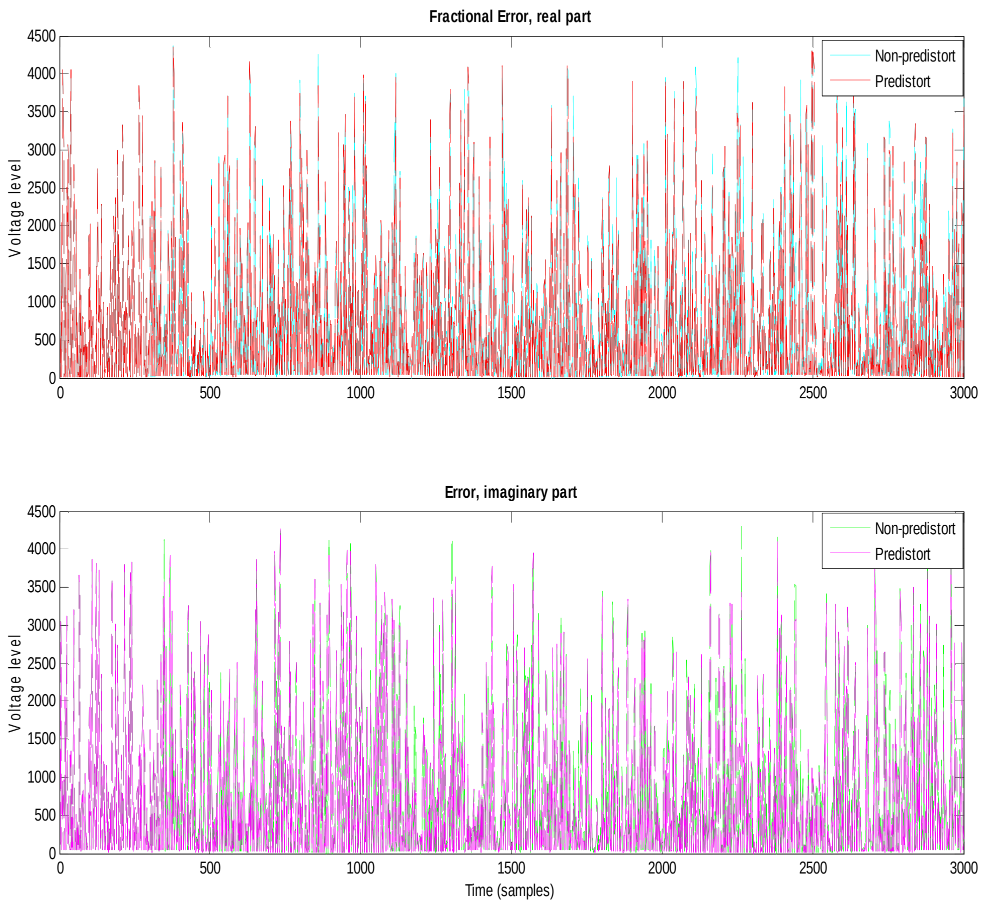


Figure 7.5: Comparison of errors with predistorted and non predistorted PA system

7.2.6 Error comparison between Pridistorted and non predistorted system (Normalised):

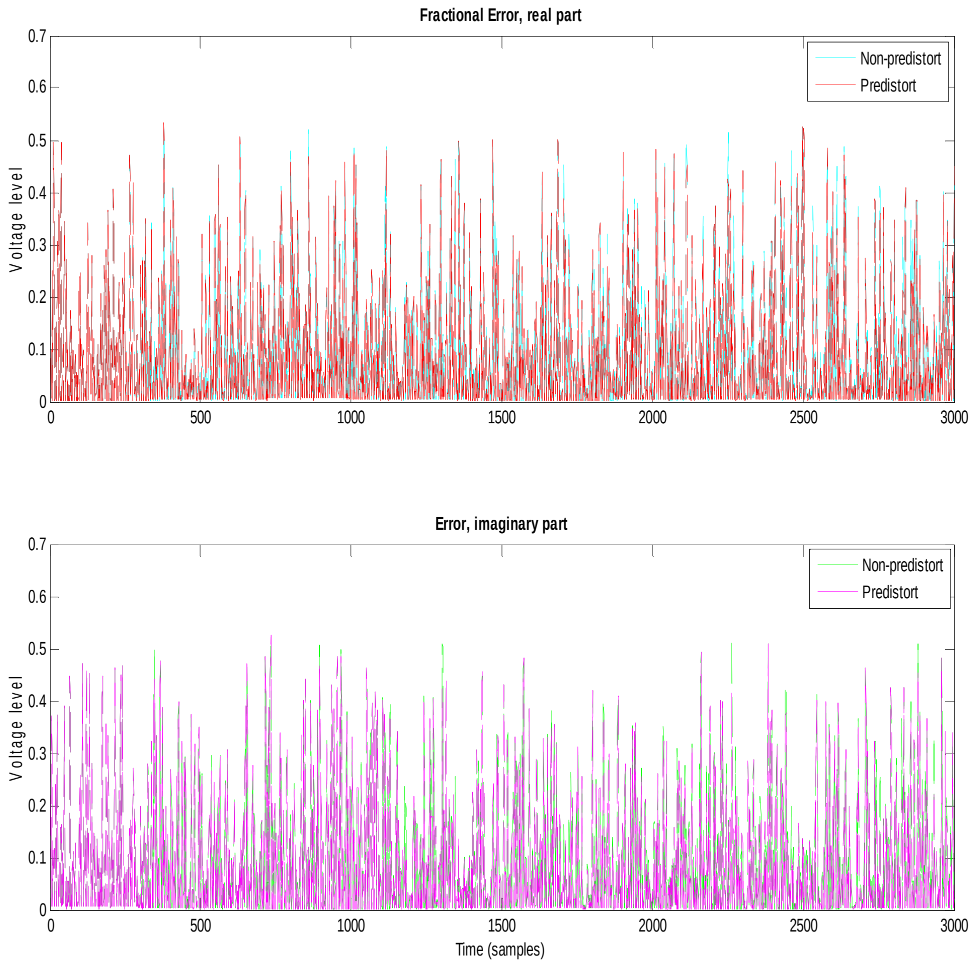


Figure 7.6: Normalized comparison of errors with predistorted and non predistorted PA system

7.2.7 Error magnitude and phase for DPD system:

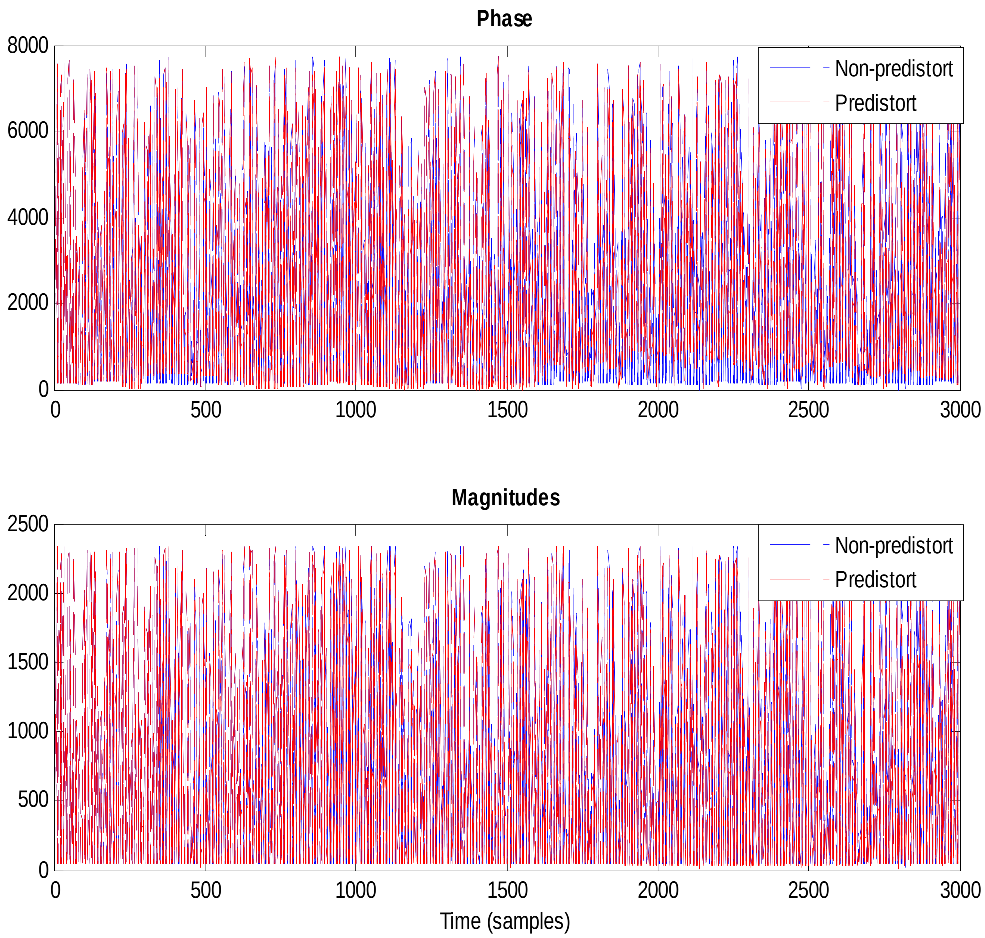


Figure 7.7: Error magnitude and phase for DPD system

7.2.8 Different signal magnitude for DPD system:

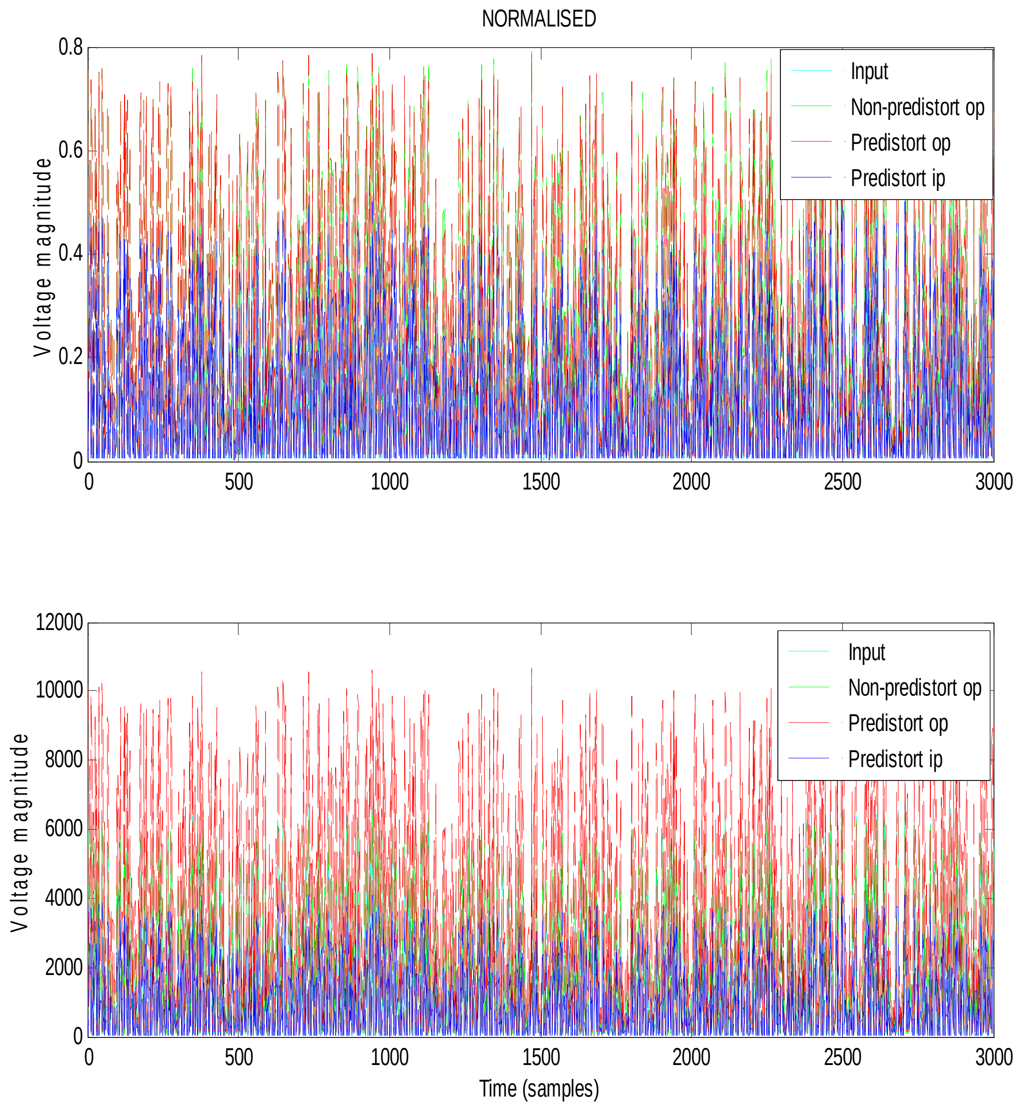


Figure 7.8: Different Signal magnitude for DPD system:

7.2.9 Frequency plot:

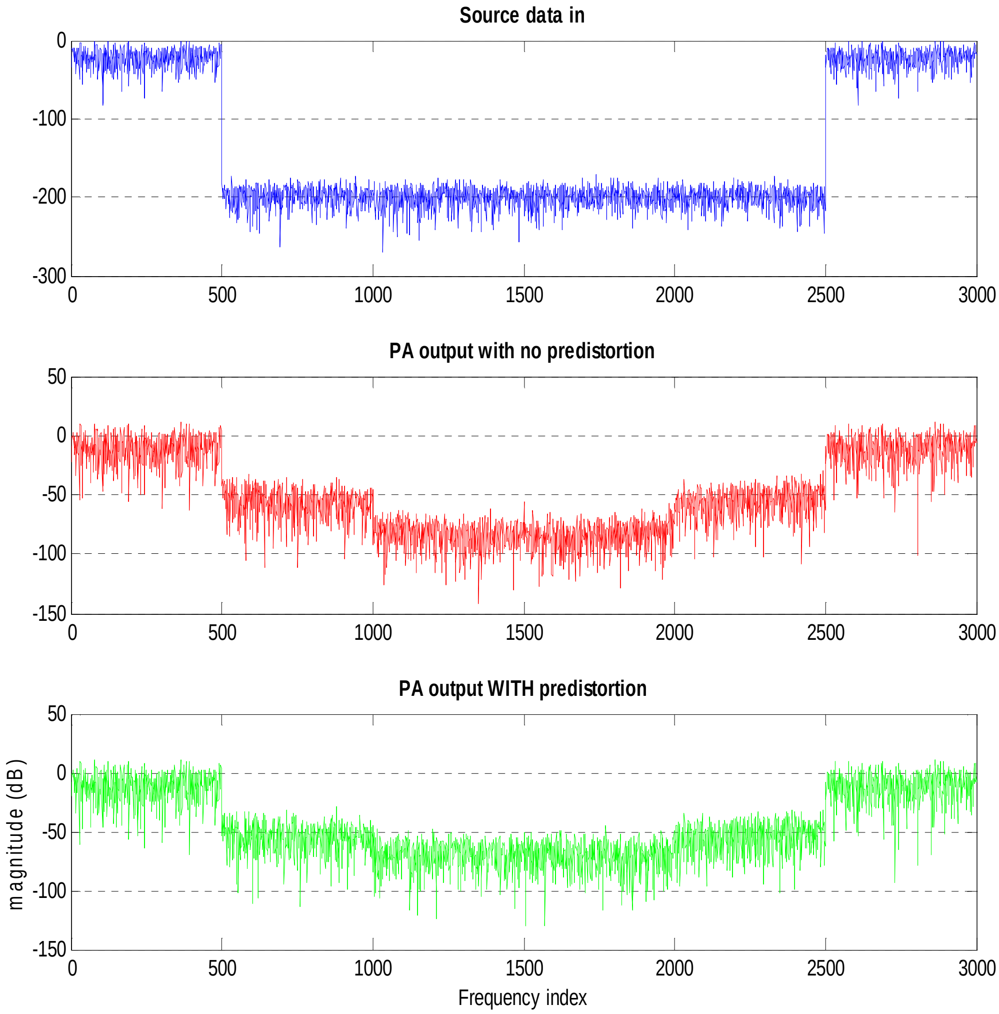


Figure 7.9: frequency plot for DPD system

7.2.10 Power Amplifier (no Predistortion) Unclipped input:

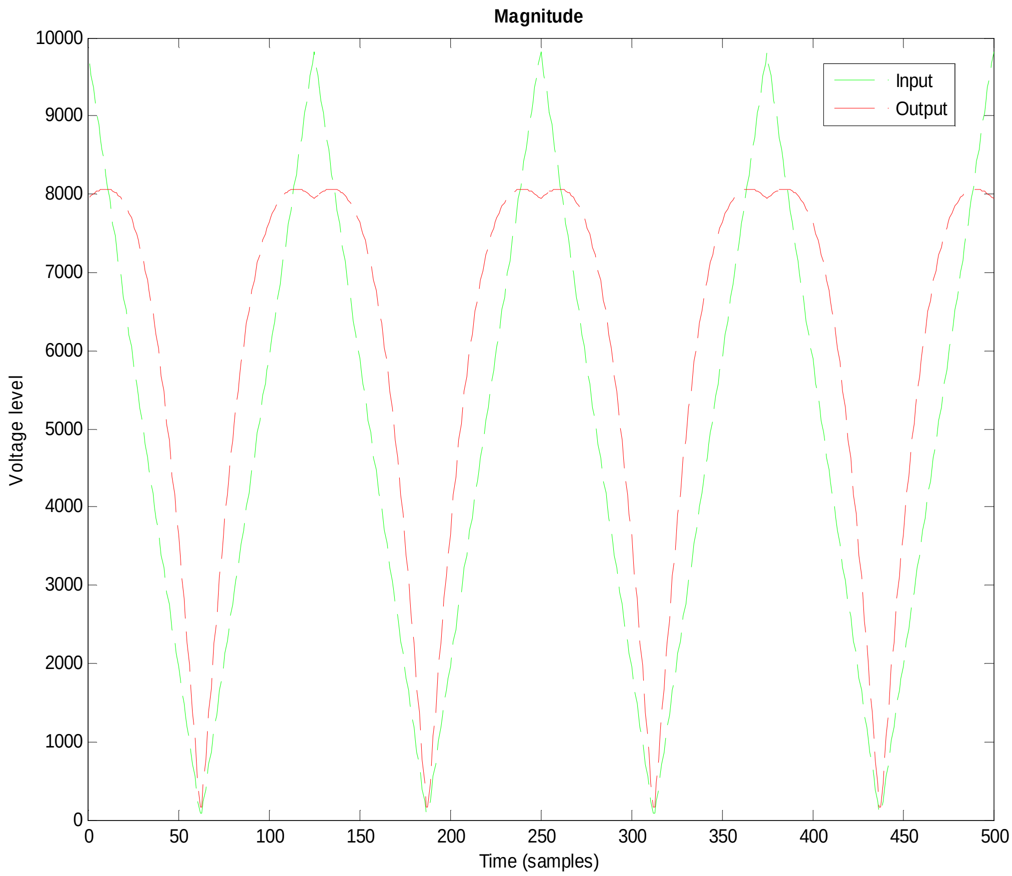


Figure 7.10: Power Amplifier (no Predistortion) Unclipped input

CHAPTER EIGHT

Conclusion and Challenges

This dissertation considered the design of digital predistortion systems to linearize power amplifiers with memory effects. By adding a digital predistorter in the base band, the power amplifier is allowed to operate into its nonlinear region, thereby significantly increasing its efficiency. The efficiency gain translates into electricity and cooling cost savings for service providers and longer battery life for mobile terminal users. The challenge here is to address the memory effects exhibited by the higher power amplifiers or the power amplifiers for wideband signals. In addition, analog components in the transmitter have imperfections that need to be compensated as well.

A multi-LUT digital adaptive predistorter capable of supporting different modulation formats at the time that assures high efficient linear amplification by feed backing the BER information at the receiver has been presented. The design process and basic principles have been reported, as well as simulation results that show its good performance. Among different possibilities for model parameters identification those showing less computational effort have been here considered, aiming to set up the predistorter algorithm inside the already existent DSP infrastructure in modern communication systems, thus minimizing perturbation to other DSP functionalities.

The main advantage of this multi-LUT predistorter is that even when the dynamic range is adjusted at the best back-off level, there is no penalization in the amplification gain, since it always operates as close to saturation as the specified BER at reception permits.

8.1 Design Solutions:

- The ADS digital predistorter design example demonstrates the performance that can be achieved with linearization.
- 2. System level simulation provides a solid starting point for building an implementation quickly.
- 3. Designed components can be integrated into a system to witness the impact on overall performance.

8.2 Contributions:

Primary contributions of this dissertation are summarized here:

- Designed novel predistorters and their parameter extraction algorithms, which include the Hammerstein predistorter, the memory polynomial predistorter, and the combined predistorter.
- Explained the benefits of including even-order terms in power amplifier modeling and predistorter design.
- Designed compensation techniques for analog imperfections in the transmitter, which include the linear frequency distortion and frequency-dependent gain/phase imbalance.
- Integrated a wideband predistortion testbed. In addition, we implemented the memory polynomial predistorter training algorithm on a Texas Instruments C6711 Starter Kit and evaluated the real-time performance of the

8.3 Suggestions for Future Research:

This dissertation can be extended in a number of directions, including:

- Designing a fast adaptive memory polynomial predistorter based on the orthogonal polynomial theory.
- Performing tests on different types of power amplifiers and establishing connections between the memory behavior of the power amplifier and the kernels of the Volterra series.
- Combining predistortion with peak-to-average ratio reduction techniques to further improve the efficiency of the power amplifier.

8.4 Implementation Challenges:

One of the most critical sub-systems in a 3G base station is the base band processing card. This card takes digitized base band radio signals and base station control and signaling as input, and it produces multiple simultaneous user data channels, including voice and data streams. The base band processing can be broadly divided into symbol-rate processing and chip-rate processing. The symbol-rate processing for voice users is somewhat different from that of data users. Symbol-rate processing for voice users encompasses the Viterbi decoder, de-interleaver, and rate matching. Symbol-rate processing for data additionally includes turbo encoding and decoding. Chip-rate processing involves the processing of the input data at chip-rate or a multiple of the chip-rate. In 3GPP, chip-rate processing requires significant parallel processing and, therefore, is ideally suited for Xilinx FPGA implementation. The inherent flexibility of the FPGA provides a platform that can be easily migrated to support multiple cell sizes, deployment scenarios, and multiple standards. 3GPP Rel'5 has generated tremendous interest in the 3G base station industry for the wide ranging economic and deployment benefits it offers. Below are some of the crucial requirements encountered by the base band processing designers that ultimately drive the choice of the hardware platform.

8.4.1 Processing Speed:

Several advanced signal processing techniques such as FIR filters, FFT/IFFT, and turbo convolution coding/decoding are being used in base band signal processing. These are very computationally intensive and require several billion multiply and accumulate (MAC) operations per second.

8.4.2 Flexibility:

With the continuing changes in the 3GPP specification, a flexible base band processing solution is extremely important to enable equipment manufacturers to upgrade their base station to include new feature sets from future 3GPP specifications. Flexible solutions allow equipment manufacturers to add their own IP to differentiate their product and to scale their design based on cell size, latency, number of antennas and sectors, input sample bandwidth, and detection methods.

8.4.3 Low Power/Cost:

To reduce cost, the design should be low power to allow the use of smaller and lower cost cooling systems and less expensive battery backup systems. Because current base station designs require significant space; the size of the base band design directly impacts the deployment and rental costs for the space used by the base station.

8.4.4 Time to Market:

Because 3GPP is relatively new, the availability of software, boards, IP, and reference designs enables the designer to shorten the development cycle. This has a direct impact on the time to market of the product and the early success in gaining market share.

REFERENCES

1. Navid Lashkarian, Chris Dick FPGA implementation of digital predistortion linearizers for wideband power Amplifiers; Signal Processing Division, Xilinx Inc., San Jose, USA,.2007.
2. Lei Ding, Digital Predistortion of Power Amplifiers for Wireless Applications; 2004.
3. Cripps, S.C.: RF power 2005. Microwave Journal 48 (2005), 22-36.
4. Euns, C.; Powers, E.J.: A new Volterra predistorter based on the indirect learning architecture. IEEE Tran. On Signal Processing 45 (1997), 223-227.
5. Kim, J.; Konstantinou, K.: Digital predistortion of wideband signals based on power amplifier model with memory. Electronics Letters 37 (2001), 1417-1418.
6. Nazim Ceylan Linearization of power amplifiers by means of digital predistortion– Erlangen 2005.
7. García, P.; Ortega, A.; de Mingo, J.; Valdovinos, A.: Nonlinear distortion cancellation sings LINC transmitters in OFDM systems. IEEE Trans. On Broadcasting 51 (2005), 84-93.
8. Altera , Digital Predistortion Reference Design, Altera corporation, 2006
9. Shawn P. Stapleton, Agilent EEs of EDA Digital Predistortion of Power Amplifiers in cooperation with: Agilent Technologies, Inc, 2005.
10. Ding, L.; Raich, R.; Zhou, G.T.: A Hammerstein predistortion linearization design based on the indirect learning. IEEE Int. Conf. on Acoustics, Speech, Signal Proc. **3** (2002), 2689-2692.
11. Jung, W.J.; Kim, W.R.; Kim, K.M.; Lee, K.B.: Digital predistorter using multiple lookup tables. IEEE Electronics Letters 39 (2003), 1386-1388.
12. Baudoin, G.; Jardin, P.: Adaptive polynomial pre-distortion for linearization of power amplifiers in wireless communications and WLAN. Int. Conf. on Trends in Comm. EUROCON 2001 1, 157-160.
13. Marsalek, R.; Jardin, P.; Baudoin, G.: From post-distortion to pre-distortion for power amplifiers linearization. IEEE Electronics Letters 7 (2003), 308-310.
14. Ma, Z., Zierdt, M., Dunkleberger, L., and Pastalan, J., “Memoryless power Amplifier characterization for digital baseband predistortion (ii).” unpublished work, Jan. 2001.

15. Look-Up Table Techniques for Adaptive Digital Predistortion: A Development and Comparison Kathleen J. Muhonen, Mohsen Kavehrad, *Fellow, IEEE*, and Rajeev Krishnamoorthy, IEEE transactions on vehicular technology, vol. 49, no. 5, september 2000.

16. An Approximation of Volterra Series Using Delay Envelopes, Applied to Digital Predistortion of RF Power Amplifiers With Memory Effects, Nima Safari, *Student Member, IEEE*, Terje Røste, *Member*, IEEE, Pavlo Fedorenko, *Student Member, IEEE*, and J. Stevenson Kenney, Senior Member, IEEE, VOL. 18, NO. 2, FEBRUARY 2008

17. Ma, Z., Zierdt, M., and Pastalan, J., "Characterization of power amplifier memory effect for digital baseband predistortion." unpublished work, Jan. 2001.

18. Ding, L., Zhou, G. T., Morgan, D. R., Ma, Z., Kenney, J. S., Kim, J., and Giardina, C. R., "A robust predistorter constructed using memory polynomials,"IEEE Trans. Commun., vol. 2, pp. 159–165, Jan. 2004.

19. Graboski, J. and Davis, R. C., "An experimental M-QAM modem using amplifier linearization and baseband equalization techniques," in Proc. IEEE Nat.Telecommun. Conf., pp. E3.2.1–E3.2.6, Nov. 1982.

20. Eskinat, E., Johnson, S. H., and Luyben, W. L., "Use of Hammerstein models in identification of nonlinear systems," AICHE J., vol. 37, pp. 255–267, Feb. 1991.

21. B. **S.** Virdee , B. Banyamin Digital Pre-distortion Linearizer, London Metropolitan University, Dept. of Computing, Communications Technology & Mathematics.

22. Zhu, A.; Brazil, T.J.: An adaptive Volterra predistorter for the linearization of RF high power amplifiers. MTT-IMS 2002 1 (2002), 461-464.

23. Kenney, J. S., Woo, W., Ding, L., Raich, R., Ku, H., and Zhou, G. T., "The impact of memory effects on predistortion linearization of RF power amplifiers," in Proc. Int. Symp. Microwave Optical Technol., pp. 189–193, June 2001.

24. Kim, J. and Konstantinou, K., "Digital predistortion of wideband signals based on power amplifier model with memory," Electron. Lett., vol. 37, pp. 1417–1418, Nov. 2001.

25. Introduction to CPLD and FPGA Design By Bob Zeidman, from Prentice-Hall

26. Ma, Z., Zierdt, M., Dunkleberger, L., and Pastalan, J., "Memoryless power amplifier characterization for digital baseband predistortion (ii)." unpublished work, Jan. 2001.

27. Digital Predistortion User Guide, Agilent Technologies, 2005.

28. Ma, Z., Zierdt, M., and Pastalan, J., "Charaterization of power amplifier memory effect for digital baseband predistortion." unpublished work, Jan. 2001.

29. Maas, S. A., Nonlinear Microwave Circuits. Piscataway, NJ: IEEE Press, 1997.

30. Moon, T. K. and Stirling, W. C., Mathematical Methods and Algorithms for Signal Processing. Englewood Cliffs, NJ: Prentice Hall, 1999.

31. Morgan, D. R., Ma, Z., and Ding, L., "Reducing measurement noise effects in digital predistortion of RF power amplifiers," in Proc. IEEE Int. Conf. Commun., pp. 2436–2439, May 2003.

32. Nagata, Y., "Linear amplification technique for digital mobile communications," in Proc. IEEE Veh. Technol. Conf., vol. 1, pp. 159–164, May 1989.

33. Oppenheim, A. V. and Schafer, R. W., Discrete-Time Signal Processing. Englewood Cliffs, NJ: Prentice-Hall, 1999.

34. Raich, R., Qian, H., and Zhou, G. T., "Digital baseband predistortion of nonlinear power amplifiers using orthogonal polynomials," in Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing, pp. 689–692, Apr. 2003.

35. Raich, R. and Zhou, G. T., "On the modeling of memory nonlinear effects of power amplifiers for communication applications," in Proc. IEEE Digital Signal Processing Workshop, Oct. 2002.

36. Saleh, A. A. M., "Frequency-independent and frequency-dependent nolinear models of TWT amplifiers," IEEE Trans. Commun., vol. 29, pp. 1715–1720, Nov. 1981.

37. Saleh, A. A. M. and Salz, J., "Adaptive linearization of power amplifiers in digital radio systems," Bell Syst. Technical J., vol. 62, pp. 1019–1033, Apr. 1983.

38. Stapleton, S. P. and Cavers, J. K., "A new technique for adaptation of linearizing predistorters," in Proc. IEEE Veh. Technol. Conf., pp. 753–758, May 1991.

39. Stapleton, S. P. and Costescu, F. C., "An adaptive predistortion for a power amplifier based on adjacent channel emissions," IEEE Trans. Veh. Technol., vol. 41, pp. 49–56, Feb. 1992.

40. Sundstr¨om, L., Digital RF Power Amplifier Linearisers. PhD thesis, Lund Univ., Lund, Sweden, 1995. 102

41. Giardina, C. R., Kim, J., and Konstantinou, K., "System and method for predistorting a signal using current and past signal samples," July 2001. U.S. Patent Application, Serial No. 09/915042.

42. Graboski, J. and Davis, R. C., "An experimental M-QAM modem using amplifier linearization and baseband equalization techniques," in Proc. IEEE Nat. Telecommun. Conf., pp. E3.2.1–E3.2.6, Nov. 1982.

43. Kang, H. W., Cho, Y. S., and Youn, D. H., "On compensating nolinear distortions of an OFDM system using efficient adaptive predistorter," IEEE Trans. Commun., vol. 47, pp. 522–526, Apr. 1999.

44. Karam, G. and Sari, H., "Data predistortion techniques using intersymbol interpolation," IEEE Trans. Commun., vol. 38, pp. 1716–1723, Oct. 1990.

45. Karam, G. and Sari, H., "A data predistortion technique with memory for QAM radio systems," IEEE Trans. Commun., vol. 39, pp. 336–344, Feb. 1991.

46. Kenington, P. B., High-Linearity RF Amplifier Design. Boston, MA: Artech House, 2000.

47. Tuthill, J. and Cantoni, A., "Optimum precompensation filters for iq modulation systems," IEEE Trans. Commun., vol. 47, pp. 1466–1468, Oct. 1999.

48. Valkama, M., Renfors, M., and Koivunen, V., "Compensation of frequencyselective i/q imbalances in wideband receivers: models and algorithms," in Proc. IEEE Workshop Signal Processing Advances in Wireless Commun., pp. 42–45, Mar. 2001.

49. Vuolevi, J. H. K., Rahkonen, T., and Manninen, J. P. A., "Measurement technique for characterizing memory effects in RF power amplifiers," IEEE Trans. Mi- crowave Theory Tech., vol. 49, pp. 1383–1388, Aug. 2001.

50. Watkins, D. S., Fundamentals of Matrix Computations. New York, NY: John Wiley & Sons, 2 ed., 2002.

51. Westesson, E. and Sundstrom, L., "A complex polynomial predistorter chip in CMOS for baseband or IF linearization of RF power amplifiers," in Proc. IEEE Int. Symp. Circuits and Systems, pp. 206 –209, May 1999.