

Low Power Pre-Distorter Design For 5G Radio Using Machine Learning

DI WANG

SUMEETH DIDDIGI KULKARNI

MASTER'S THESIS

DEPARTMENT OF ELECTRICAL AND INFORMATION TECHNOLOGY

FACULTY OF ENGINEERING | LTH | LUND UNIVERSITY



Low Power Pre-Distorter Design For 5G Radio Using Machine Learning

Di Wang
di4156wa-s@student.lu.se
Sumeeth Diddigi Kulkarni
su1825di-s@student.lu.se

Department of Electrical and Information Technology
Lund University

Academic Supervisor: Liang Liu
Sidra Muneer

Supervisor: Asad Jafri
Naeem Abbas

Examiner: Erik Larsson

September 11, 2020



LUND
UNIVERSITY

§



Abstract

A Power Amplifier (PA) is an essential electronic component in all microwave and millimeter-wave applications and, more specifically, in any transmitting system where the level of input power signal needs amplification to the desired level. Linearity and high efficiency are of utmost importance in PAs. However, high-efficiency PAs tend to be non-linear, and PAs working in the linear region might have low efficiency. Hence, there is always a trade-off between efficiency and linearity while designing a PA.

For an efficient system design, the efficiency of the PA gets prioritized by the designers, and for linearity, an additional linearization technique can be deployed.. Designers have been considering many linearization methods. Among those, digital predistorter tends to be the most popular one as it can provide a right amalgamation between linearity performance and implementation complexity.

However, the computation process used to obtain an inverse PA behavior inside a digital predistorter consumes significant power. In this thesis, the main target is to find a power-efficient way to enhance the current algorithm for the digital predistorter (look-up-table based) and evaluate the power results along with Adjacent Channel Power Ratio (ACPR).

Popular Science Summary

The power amplifier is an electronic device designed to increase the magnitude of a given input signal. The signal with an enhanced magnitude drives the devices like speakers, headphones, and RF transmitters. Power amplifiers are one of the essential elements within the wireless communication area since base stations utilize them to broadcast and transmit wireless signals to the users. Moreover, with increased power levels, higher data transfer rates, and long-range transmission is possible. From a computation perspective, an ideal power amplifier multiplies the input signal with the desired gain. However, the power amplifiers are a non-linear source for a communication system and different from the ideal scenario. The non-linearity is introduced as output power increases and reaches near to its maximum threshold, which can lead to in-band distortion within the system. For this reason, the linearization of power amplifiers is an essential topic and widely researched in the digital communication field.

The most common method for linearizing a power amplifier's behavior is the digital predistortion. This method is very power efficient as well as cost-saving. Ideally, with the predistortion, the characteristics of a power amplifier are inversed in order to compensate for the non-linearities. The predistorter unit, along with the system's amplifier inside a digital communication system, corrects any possible gain and phase non-linearities introduced to produce a distortion free signal. The need for a bigger, less efficient, expensive amplifier is avoided by using predistortion to provide the gain stability at the output of the amplifiers. Although the predistortion technique is widely implemented and successful, the current techniques of employing the predistortion are very power-consuming due to their complexity and the amount of computational power they require. For that reason, new approaches are under the scope.

Earth, our 'home' has limited resources, which are continually diminishing. With technological improvements, the goal is to use resources in an efficient way for a sustainable world. Without any doubt, increasing the efficiency of mobile communication systems is crucial. The base station is the most power-consuming block in mobile communications, out of which power amplifiers consume a significant portion of the overall power budget. In 2010, Europe's telecoms used the power equivalent of 21.4 TWh, which is expected to rise to 35.8 TWh by 2020 [1]. Thus, increasing improving the efficiency of power amplifiers is vital for sustainable base stations.

This research assesses the influence of LUT numbers on power consumption and ACPR rating. The main goal is to reduce the power consumption of current DPD design by proposing a power-efficient algorithm to decrease the number of hardware resources that meets the required neighbouring channel leakage rating for 3GPP specification.

Table of Contents

1	Introduction	1
1.1	Background and Motivation	1
1.2	Outline of the thesis	2
2	Power Amplifier Theory	3
2.1	Nonlinear behavior of PA	3
2.2	Linearity and efficiency	4
2.3	Nonlinear power amplifier models	10
3	Digital Predistorter Theory	13
3.1	Introduction	13
3.2	Digital Predistortion Identification Architectures	15
3.3	DPD Techniques for Linearizing a Power Amplifier	16
3.4	Look-up Table Based Predistorter	17
4	Design theory and Implementation	19
4.1	ACPR requirements in this project	19
4.2	Power issues	21
4.3	Machine learning algorithm	25
4.4	The predistortion module	26
4.5	Machine learning steps	28
4.6	Hardware implementation	29
5	Results	31
5.1	Area report	31
6	Conclusion and Future Work	39
6.1	Conclusion	39
6.2	Future work	39
	References	41

List of Figures

1.1	A simplified system with PA and predistorter	2
2.1	AM/AM distortion	4
2.2	AM/PM distortion	5
2.3	Harmonic distortion and Intermodulation distortion[8]	7
2.4	1dB compression point[9]	8
2.5	Input back-off and output back-off[10]	8
2.6	Third-order intercept point[11]	9
2.7	Adjacent channel[27]	9
2.8	Memoryless behavioral model[13]	10
2.9	Memory effect behavioral model[15]	11
3.1	Basic Principle of Digital Pre Distortion [21]	14
3.2	System Block Diagram of Digital Predistortion	14
3.3	Direct Learning Architecture	15
3.4	Indirect Learning Architecture	16
3.5	Volterra Series Structure	17
3.6	Look-up Table Based Predistorter	18
4.1	ACPR with X LUTs predistorter	19
4.2	ACPR with X-1 LUTs predistorter without modification	20
4.3	ACPR with X-1 LUTs predistorter after modification	20
4.4	Dynamic power–switching power and short-circuit power	22
4.5	Clock gating theory block[32]	24
4.6	Clock buffer schematic	25
4.7	Block diagram of predistortion	26
4.8	Address generator	27
4.9	Machine learning training steps	28
4.10	Top level block diagram	29
5.1	Area of Different Blocks inside conventional DPD design (μm)	32
5.2	Area of Scheduler Design (μm)	33
5.3	Area of Different Blocks inside One DPD Model for Scheduler and Clock Gated Scheduler Design (μm)	34

5.4	Power of Different Blocks inside One DPD Model for Scheduler Design (W)	37
5.5	Power Results for Scheduler Design (W)	37
5.6	Power of Different Blocks inside One DPD Model for Clock Gated Scheduler Design (W)	38
5.7	Power Results for Clock Gated Scheduler Design (W)	38

List of Tables

4.1	Power variation according to technology[29]	21
5.1	Area of Different Blocks inside conventional DPD design (μm)	32
5.2	Area of Scheduler Design (μm)	33
5.3	Area of Different Blocks One DPD Model for Scheduler and Clock Gated Scheduler Design (μm)	34
5.4	Conventional Mode Power report for Scheduler Design and Clock Gated Design (mW)	36
5.5	Open loop Mode Power report for Scheduler Design and Clock Gated Design(mW)	36
5.6	Optimized Mode Power report for Scheduler Design (mW)	36
5.7	Optimized Mode Power report for Clock Gated Design (mW)	36

List of Abbreviations

ACLR	Adjacent Channel Leakage Ratio
ACPR	Adjacent Channel Power Ratio
A/D	Analog-to-Digital
CMOS	Complementary metal-oxide-semiconductor
D/A	Digital-to-Analog
DLA	Direct Learning Architecture
DPD	digital predistortion
HD	Harmonic Distortion
ILA	Indirect Learning Architecture
IMD	Inter-Modulation Distortion
IP3	Third-order intercept point
LUT	Look-up table
MOSFET	Metal Oxide Semiconductor Field Effect Transistor
NNs	neural networks
PA	Power Amplifier
PTPX	Prime-Time PX
RF	Radio Frequency
SoC	Silicon on chip

Introduction

The increasing demands of users for higher data rates and transmission volume has vastly promoted the vigorous development of wireless communication, it has also ensured that there is an immense focus on new innovative technologies to meet all these needs[2]. PA is a key component of communication system, but they are inherently nonlinear. The non-linearity generates spectral re-growth, which leads to adjacent channel interference and violations of the out-of-band emissions standards mandated by regulatory bodies. With the development of communication technology, a variety of different linearization technologies have been proposed. Amongst them, digital predistortion (DPD) has attracted wide attention, this thesis work is on implementation of dual stage learning model for DPD using Look-up table (LUT), to reduce the power consumption.

1.1 Background and Motivation

PAs are crucial components in wireless communication systems; high efficiency and linearity are the most basic and significant requirements for them. However, there is a trade-off between PA efficiency and linearity.

Efficiency is the ability of a PA to provide most of the power taken from supply to load, and linearity is the capability of a PA to provide a linear output at high input power, more in-sight on efficiency and linearity in chapter 2. PA can be run efficiently by operating it at lower power (that is, “backed off”) so that it works within the linear portion of its operating curve. Still, due to the newer transmission formats like 3GPP, etc., the power amplifier needs to be backed off well below its maximum saturated output power, which results in the very low-efficiency area.

DPD is one of the most popular techniques. It features an excellent linearization capability, the ability to preserve overall efficiency, along with low implementation complexity. There has been continuous research on DPD, and researchers have proposed many methods to implement it. Ideally, the main objective is to replicate the inverse of power amplifiers nonlinear behavior in the predistortion stage, to achieve a linear output of the PA. Due to the computation process to get the inverse response of PAs, power consumption becomes challenging to optimize.

In the below figure 1.1, the block X stands for the PA and the block before it is a digital predistorter. PA is a non-linear component, and ideally, the behavior of the DPD block (placed before the PA) should be the inverse of the PA. Thus the

distortion from both of them will be canceled, resulting in a linear input-output curve. Currently, several algorithms exist to model and estimate PA behavior and to compensate for the linearity. These algorithms are sometimes complex to implement in hardware and provide a complexity accuracy trade-offs. In practice, these algorithms are resource-intensive, resulting in high power consumption.

The current DPD design itself consumes a significant portion of overall chip power (30%-60%), making the design infeasible. The motivation of this thesis work is to reduce the power consumption by modifying the current single stage learning model (based on Volterra-Series) to dual stage learning with the aid of a scheduler, which trains the system to work with reduced LUTs. The main parameter, ACPR, is measured to evaluate the performance of the system as compared to the current conventional DPD model.

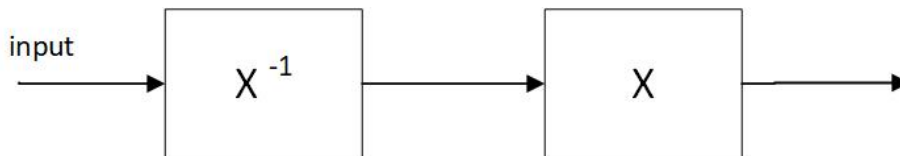


Figure 1.1: A simplified system with PA and predistorter

1.2 Outline of the thesis

The thesis is organized as following:

- Chapter 1 gives a brief introduction and background of the thesis.
- Chapter 2 introduces the Power amplifier and all concepts related to it in detail.
- Chapter 3 first gives a brief about Digital Predistortion and then reviews the identification architectures along with the DPD techniques used to linearize a PA.
- Chapter 4 instigates in-depth detail about the hardware design and optimization techniques used in this thesis.
- Chapter 5 includes the details about the verification of the design, power, and area analysis.
- Chapter 6 presents the main conclusion points along with the suggestions about future work to improve the current design.

Power Amplifier Theory

2.1 Nonlinear behavior of PA

Ideally, a power amplifier can be described by a linear expression, where the output is scalar multiple of the input. Here, as in the formula below, A is the gain of the PA, therefore, the behavior could be described as [3],

$$y_{out}(t) = Ax_{in}(t) \quad (2.1)$$

However, in reality, when the input power of the PA increased, the nonlinear behavior cannot be ignored. The output is expressed with a nonlinear function of the input. When the signal goes through the nonlinear component, the impact on the spectrum that can be observed at the output, is called distortion. Assume that there is a memoryless PA (memory effects are quite important to PA as well, it will be introduced in the coming section), the output can be represented by Taylor Series [4], as listed below in formula 2.2.

$$y_{out} = \sum_{k=1}^{\infty} a_k x_{in}^k = a_1 x_{in} + a_2 x_{in}^2 + a_3 x_{in}^3 + a_4 x_{in}^4 + a_5 x_{in}^5 + \dots \quad (2.2)$$

In the formula above, x_{in} and y_{out} are the input and output, which are varying according to time, a_k are the gains of the corresponding terms. In fact, one coefficient will only work for a specific frequency component. The first term $a_1 x_{in}$ of the expression 2.2 stands for the linear part, which is the desired output of the PA, as shown in equation 2.1.

The even order terms $a_2 x_{in}^2$, $a_4 x_{in}^4$, $a_6 x_{in}^6$,... and $a_{in} x_{in}^{2k}$, will introduce extra frequency components which are multiple times of the carrier frequency, usually these are out of band and can be filter out. This kind of distortion is called Harmonic Distortion (HD).

However, the odd order terms above $a_3 x_{in}^3$, $a_5 x_{in}^5$, $a_7 x_{in}^7$,... and $a_{in} x_{in}^{2k+1}$, introduce some frequency components which fall close to the carrier. These components, comparing with HD, are more difficult to be filtered out, are called Inter-Modulation Distortion (IMD). Inter modulation products cause both in-band and out of band distortion.

2.2 Linearity and efficiency

The non-linear performance of the power amplifier (PA) is a serious problem in wireless communication systems. Its main requirement is the desire to meet high linear power amplification while keeping the higher power efficiency [5]. For the first requirement, the PA needs to work in linear region. But operating in linear region means that the maximum output power of PA must be decreased, which leads to lower efficiency. However, in order to obtain greater output power, power amplifier should work near the saturation point, but at the same time, distortion will also be greatly increased. This will reduce the linearity. In general, linearity and efficiency is a pair of trade-off parameters.

2.2.1 Single-tone test

Based on formula 2.2, assume the modulated input signal contains information not only in amplitude, but also in phase. Thus, the input signal will be expressed like formula 2.3 , where $A(t) \geq 0$ and $\varphi(t)$ are the instantaneous amplitude and phase respectively. ω_c stands for the carrier frequency.

$$x_{in}(t) = A(t)\cos(\omega_c t + \varphi(t)) \quad (2.3)$$

After going through the amplifier, the distortion will appear in both amplitude and phase. In order to observe the effects of nonlinear behavior of a PA, another two terms will be introduced here, AM/AM and AM/PM characterization of distortion.

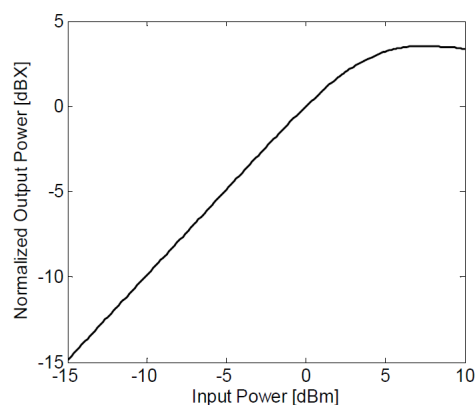


Figure 2.1: AM/AM distortion

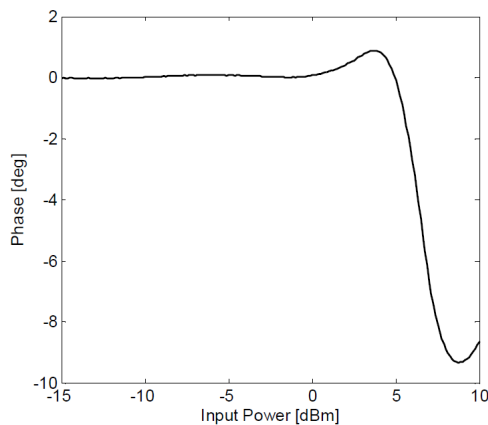


Figure 2.2: AM/PM distortion

AM/AM describes the conversion between the amplitude from input signal and the amplified amplitude present on output signal. It shows the nonlinear relationship between the amplitude of output signal and $A(t)$. AM/PM represents the relationship between the undesired phase deviation (PM) that caused by the amplitude modulation of input signal.

According to 2.3, the output signal $y_{out}(t)$ can be expressed like:

$$y_{out}(t) = F[A(t)]\cos(\omega_c t + \varphi(t) + \Phi(A(t))) \quad (2.4)$$

where $F[A(t)]$ stands for the AM/AM conversion expression, from which the change of amplitude between input and output could be seen. $\Phi(A(t))$ is the AM/PM conversion characteristic, illustrates the phase deviation according to the amplitude of input signal.

Usually, to deduce information about F and Φ , a single-tone test as introduced above is performed: the input signal $x_{in}(t)$ is assumed as a pure sine wave. However, the single-tone test is just an idealized simulation as in real life the input signal cannot be pure sine wave. It can not represent a PA at normal working conditions, where memory effects may take place (memory effects will be introduced in the coming section). Under this situation, two-tone tests are performed.

2.2.2 Two-tone test

For two-tone test[7], different from the formula 2.3, the input signal will be substituted by two input signals with the same amplitude, placing close to each other and make the frequency difference between them as Δf . (Amplitude could be different, but to simplify the following equations, assume that they have equal amplitude.)

$$x_{in}(t) = A[\cos(\omega_1 t) + \cos(\omega_2 t)] \quad (2.5)$$

$$\omega_1, \omega_2 = 2\pi(f_c \pm \frac{\Delta f}{2}) \quad (2.6)$$

So, the output will be [7],

$$\begin{aligned} y_{out}(t) = & a_1 A [\cos(\omega_1 t) + \cos(\omega_2 t)] \\ & + a_2 A^2 [\cos(\omega_1 t) + \cos(\omega_2 t)]^2 \\ & + a_3 A^3 [\cos(\omega_1 t) + \cos(\omega_2 t)]^3 \\ & + a_4 A^4 [\cos(\omega_1 t) + \cos(\omega_2 t)]^4 \\ & + \dots \end{aligned} \quad (2.7)$$

If all the polynomials are expanded so that there are no higher powers, then the following result will be obtained,

$$\begin{aligned} y_{out}(t) = & a_2 A^2 \\ & + (a_1 A + \frac{9a_3 A^3}{4}) [\cos(\omega_1 t) + \cos(\omega_2 t)] \\ & + \frac{a_2 A^2}{2} [\cos(2\omega_1 t) + \cos(2\omega_2 t)] \\ & + a_2 A^2 [\cos(\omega_1 - \omega_2)t + \cos(\omega_1 + \omega_2)t] \\ & + \frac{a_3 A^3}{4} [\cos(3\omega_1 t) + \cos(3\omega_2 t)] \\ & + \frac{3a_3 A^3}{4} [\cos(2\omega_1 + \omega_2)t + \cos(2\omega_2 + \omega_1)t] \\ & + \frac{3a_3 A^3}{4} [\cos(2\omega_1 - \omega_2)t + \cos(2\omega_2 - \omega_1)t] \\ & + \dots \end{aligned} \quad (2.8)$$

According to equation 2.8, all the polynomials could be observed clearly. During the expansion process, some extra components are generated, which are unexpected, and lead to the output signal lying on much higher frequencies. There are two categories of the generated components, which are harmonic distortion (HD) and intermodulation distortion (IMD) products.

As shown in figure 2.3, assume that $\omega_2 > \omega_1$, HD in formula 2.8 can be classified into 2nd order harmonic distortion (at frequency $2\omega_1$ and $2\omega_2$) and 3rd order harmonic distortion (at frequency $3\omega_1$ and $3\omega_2$), occurs at multiple times of the fundamental zone, which are far from the carrier and could be filtered away easily. Except the harmonic products above, other distortion products will also appear, for example at $\omega_2 - \omega_1$ and $\omega_2 + \omega_1$.

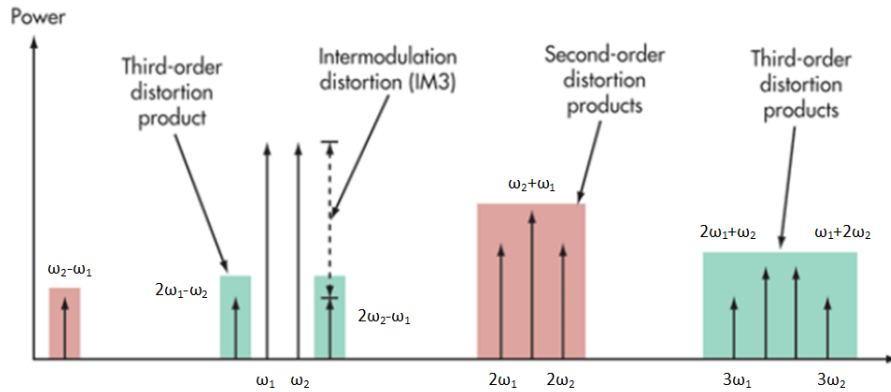


Figure 2.3: Harmonic distortion and Intermodulation distortion[8]

Thus, IMD components are the crucial distortion that need to be carefully considered, as they occur much closer to the fundamental tones. The 3rd order Intermodulation distortion falls at $2\omega_1 \pm \omega_2$ and $2\omega_2 \pm \omega_1$.

The IMD products also have components generated during the expansion process from higher orders such as $3\omega_1 - 2\omega_2$ and $3\omega_2 - 2\omega_1$. As these frequencies are even closer to the carrier frequency and will cause distortion, some additional techniques need to be implemented to improve the linearity, which are called linearization techniques. The use of linearizer is necessary because it is a good choice to ensure that the unwanted frequency components appearing near the desired signal are minimized. The linearization techniques will be further discussed in the coming chapter.

To evaluate the PA nonlinear performance, terms are commonly used, such as 1dB compression point and saturation point, input and output back-off.

Figure 2.4 shows the 1 dB compression point. When the input power is not enough, PA operates in the linear region, the gain is a constant value. But as the input power increases, the gain starts to decrease and the output power begin to compress. 1 dB compression point is the point where the difference between ideal output power and the real output power is 1 dB. It is important because after this point the output power compresses a lot more, and finally reach the saturation point.

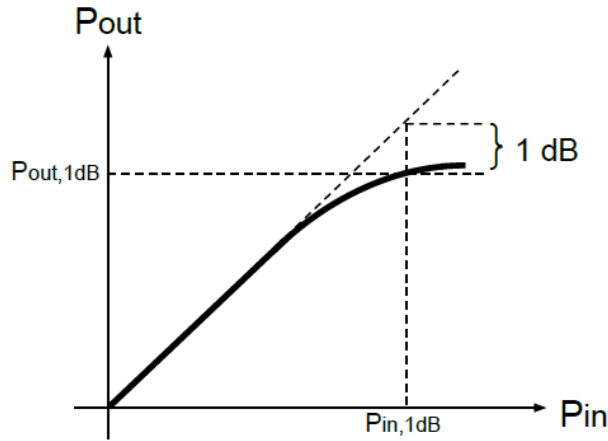


Figure 2.4: 1dB compression point[9]

Figure 2.5 illustrates the saturation point and back-off area. If the PA needs to operate properly, then it should work far away from the saturation point, which means back-off area, otherwise the output power cannot be amplified. Here, $P_{in_{SAT}}$ and $P_{out_{SAT}}$ stand for the input and output power at saturation point and $P_{in_{av}}$ and $P_{out_{av}}$ refer to the average power of input and output.

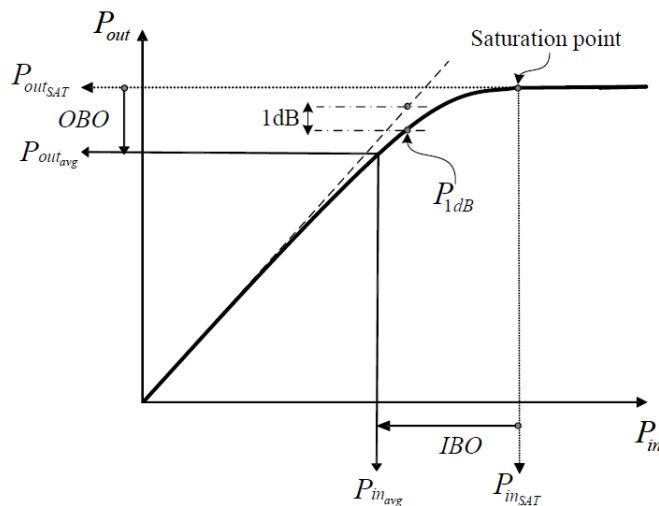


Figure 2.5: Input back-off and output back-off[10]

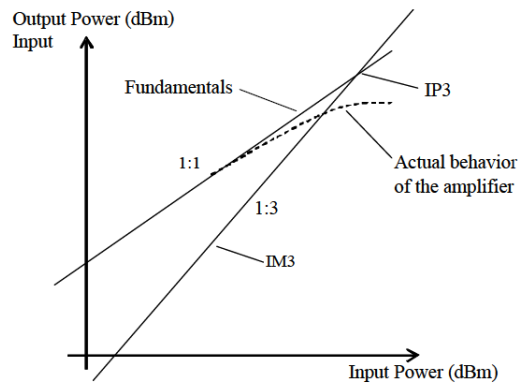


Figure 2.6: Third-order intercept point[11]

Besides the terms mentioned before, Third-order intercept point (IP3) is also worth to be discussed. As shown in the figure 2.6, IP3 is found by the ideal behavior curve of fundamental product with slope = 1, and the third order intermodulation product, whose slope is 3. IP3 is a theoretical point, that will never get in real life. It shows that amplitude of IM3 signal is the same as the amplitude of the fundamental one or input one. IP3 is used to evaluate the linearity of a PA.

2.2.3 ACPR requirements

Besides the terms introduced above, another important measure to characterize the nonlinear behavior of PA is ACLR (Adjacent Channel Power Ratio). It measures the interference of the power in adjacent frequency channels[26]. Usually it refers to the ratio of the average power in the adjacent channel to the average power in the transmitted signal channel. ACPR is also called Adjacent Channel Leakage Ratio (ACLR). Figure 2.7 describes how the signal band and upper and lower adjacent channels placed[26].

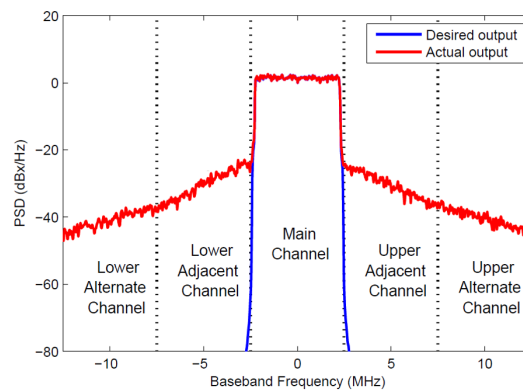


Figure 2.7: Adjacent channel[27]

$$ACPR_{dB} = 10 \times \log \frac{P_{adj}}{P_{ref}} \quad (2.9)$$

In equation 2.9, P_{adj} is the average power of the adjacent channels(also called out-of-band power), while P_{ref} is the average power of the amplified signal. This parameter can evaluate the spectral regrowth. Ideally, we want to transmit all the power on carrier and no power in the adjacent channels. However, in reality, because of nonlinear distortion introduced by components such as PA, the only possible method is to make P_{adj} as small as possible, and P_{ref} as big as possible, thus, ACPR can be smaller.

2.3 Nonlinear power amplifier models

Before designing the predistorter, the fundamental step is to find a proper description of the PA behavior. In general, PA could be divided into two main categories, PA models without memory and PA models with memory [6]. The term in the title "memory", indicates that the output of a PA at a certain point of time, depends on not only the current input but also the inputs in the past. The memory effect is quite important if it is wide band wireless system, however, when it is narrow band system, the nonlinearity of PA usually is treated as memoryless, as the output of PA mostly depends on instantaneous input[12].

2.3.1 PA models without memory effects

To model PA, there are two types, physical models and black-box models. In order to make a physical model, all electronic components that make up the PA need to be known and their relationship needs to be clear. Then use the theory and their relationship to describe it. However, the black-box models only focus on the input and output, which used to characterise the behavior of PA [13].

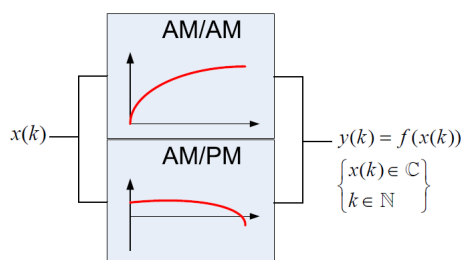


Figure 2.8: Memoryless behavioral model[13]

Memoryless Polynomial Model

Memoryless Polynomial Model is widely used to describe the nonlinear behavior of PA.

$$y_{out}(k) = \sum_{i=0}^i \epsilon_i \times x(k) \times |x(k)|^i \quad (2.10)$$

where $y(k)$ and $x(k)$ are the input and output of a PA at the k_{th} sample. ϵ_i are the coefficients that can characterize the memoryless behavior while i is the order [14].

Except the memoryless polynomial model, there are also few other models which express the output as a Fourier series expansion of the input signal. They are called Fourier series model, Bessel-Fourier model and so on [15].

2.3.2 PA models with memory effects

The memoryless PA will have impact on the amplitude, but if there exists a phase distortion, then the memory effects need to be considered. Generally, the memory effects can be set into two categories, electrical memory effects and electrothermal memory effects[16].

The electrical memory effects are caused by variable impedance which mainly come from the transistors in the bias network. The changeable impedance at DC, fundamental and harmonic band will bring unexpected signals with the same frequencies as the intermodulation distortion products. The electrothermal memory effects caused by variable properties of the transistors under different temperatures, which will also produce intermodulation distortion products [17].

In order to achieve PA models with memory effects, dynamic measurement systems should be used to characterize PA. Among all the nonlinear dynamic models, Volterra series, neural networks (NNs) and their modified versions are the most commonly used. In this thesis, the focus is on the Volterra series.

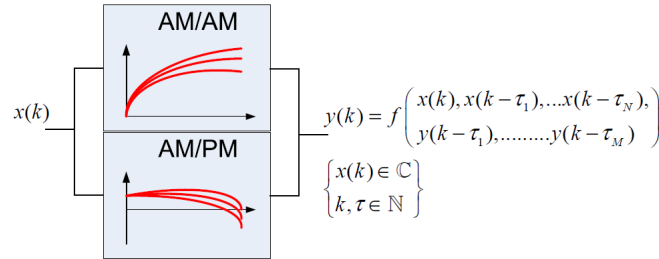


Figure 2.9: Memory effect behavioral model[15]

Volterra series was first found by an Italian mathematician Vito Volterra, considered as an extension of Taylor series. The Volterra series in continuous time can be expressed as following[18]:

$$y_{out}(t) = h_0 + \sum_{n=1}^N \int_a^b \dots \int_a^b h_n(\gamma_1, \dots, \gamma_n) \prod_{j=1}^n x(t - \gamma_j) d\gamma_j \quad (2.11)$$

where x and y are the input and output, and $h_n(\gamma_1, \dots, \gamma_n)$ is called the kernels of the Volterra series, which are the coefficients that define the system. In digital system, discrete time Volterra series is needed.

$$y_{out}(k) = \sum_{n=1}^P \sum_{\gamma_1=1}^{N-1} \dots \sum_{\gamma_n=1}^{N-1} h_p(\gamma_1, \dots, \gamma_p) \prod_{i=1}^p x(k - \gamma_i) \quad (2.12)$$

where P is the order of the polynomial, N is the depth of the memory, $(\gamma_1, \dots, \gamma_p)$ is the delays in discrete time and h_p is the coefficients, called Volterra series kernels [19]. To get more accuracy, P and N can be increased. However, the complexity of calculation will increase dramatically, as parameters grow exponentially.

Digital Predistorter Theory

3.1 Introduction

As mentioned in the introduction, PAs are key component of communication system, but they are inherently non-linear, and they are more efficient in terms of performance when operating in these conditions. Nevertheless, the peak points of the output of PA gets clipped in the compression region resulting in deterioration of the output frequency spectrum. To avoid this, estimate the inverse of the PA and place it before the PA, so that it compensates the clipping effect [20]. However, while applying this concept for high bandwidth systems, consideration of memory effect is a requisite, if not it would result in mediocre performance, as the memory depth of the PA increase with the increase in the bandwidth.

DPD is one of the well-known techniques because of its good linearization performance. To implement DPD, the initial step would be to mine out the PAs performance which is achieved by observing the output of the PA to different inputs. The following step is to study and evaluate the memory effects through amplitude-to-amplitude modulation (AM/AM) and the amplitude-to-phase modulation (AM/PM) plots. Later, the inverse equivalent for the input signal should be built after the memory effects are evaluated to eliminate the distortions, figure 3.1 describes the predistortion process. The result of the DPD is evaluated through ACPR.

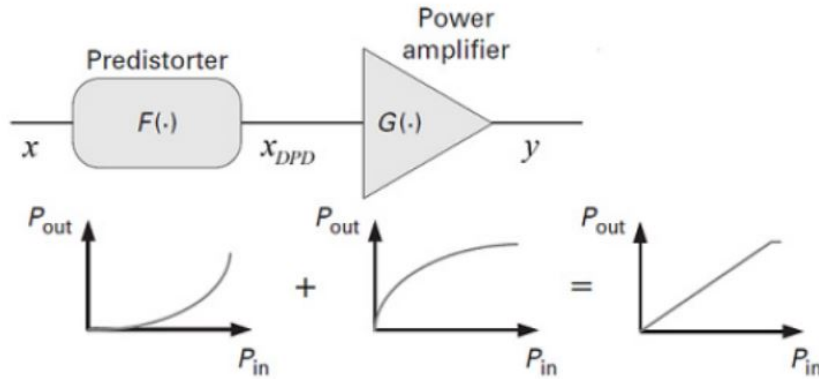


Figure 3.1: Basic Principle of Digital Pre Distortion [21]

As illustrated in the figure 3.2, the implementation of DPD technique will be in digital domain, where the digital signal processing techniques is used to make the computation simpler. The DPD system has digital, analog and Radio Frequency (RF) parts. The predistorter in the digital part takes the input signal from base-band and feeds it to Digital-to-Analog (D/A) block found in analog part. The D/A converts the digital signal to analog, and then the signal enters the PA in RF part before it goes back to the digital part through Analog-to-Digital (A/D) block where it will enter the correction algorithm. The correction algorithm helps to construct the inverse behavioral model of the PA by assessing coefficients and the distortions [22].

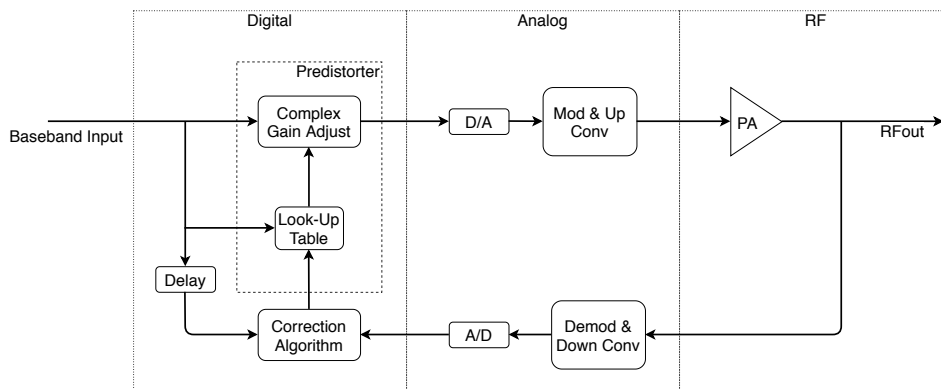


Figure 3.2: System Block Diagram of Digital Predistortion

3.2 Digital Predistortion Identification Architectures

3.2.1 Direct Learning Architecture

Direct Learning Architecture (DLA) is a well-known technique commonly used to identify the kernels or parameters, also known as coefficients, of a predistorter, illustrated in figure 3.3. DLA is used to directly lessen the inaccuracy between the anticipated output signal $y_d(n)$ and the actual output from the amplifier $y(n)$, i.e.,

$$e(n) = y_d(n) - y(n) \quad (3.1)$$

It utilizes complex algorithms to calculate the predistorter coefficients, and unfortunately, these algorithms are resource-intensive and complex in structure [24].

There are two steps involved in DLA. The first one is to identify the forward model of the PA. Later, there is an evaluation of the predistorter coefficients with the help of a non-linear algorithm to lessen the error between the expected output and the PA model output. In the next step, DLA uses the extracted coefficients to build a predistorted signal which is applied to linearize the PA. This process takes place in the predistorter block until the algorithm identifies the finest workable solution.

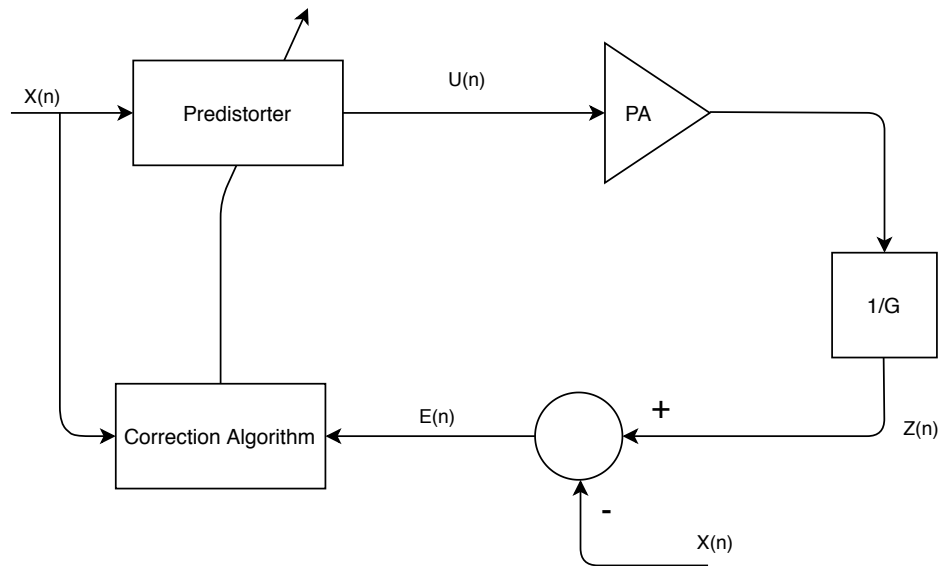


Figure 3.3: Direct Learning Architecture

3.2.2 Indirect Learning Architecture

Indirect Learning Architecture (ILA) makes use of an inverse modeling approach, where it identifies the post-inverse of the PA by using its output signal to model the PAs input. After identifying the post-inverse of PA (also known as predistorter), it exports the coefficients to an identical model and use it as the predistorter. As

illustrated in figure 3.4, the PA's output is now acting as input and the PA's input as output. ILA now extracts the coefficients by comparing these two signals, and the cycle repeats [25].

This iterative process develops a precise predistorter block. This research uses ILA, as it eliminates the need for model assumptions and coefficients estimation of the PA, which is a significant advantage when juxtaposed to DLA.

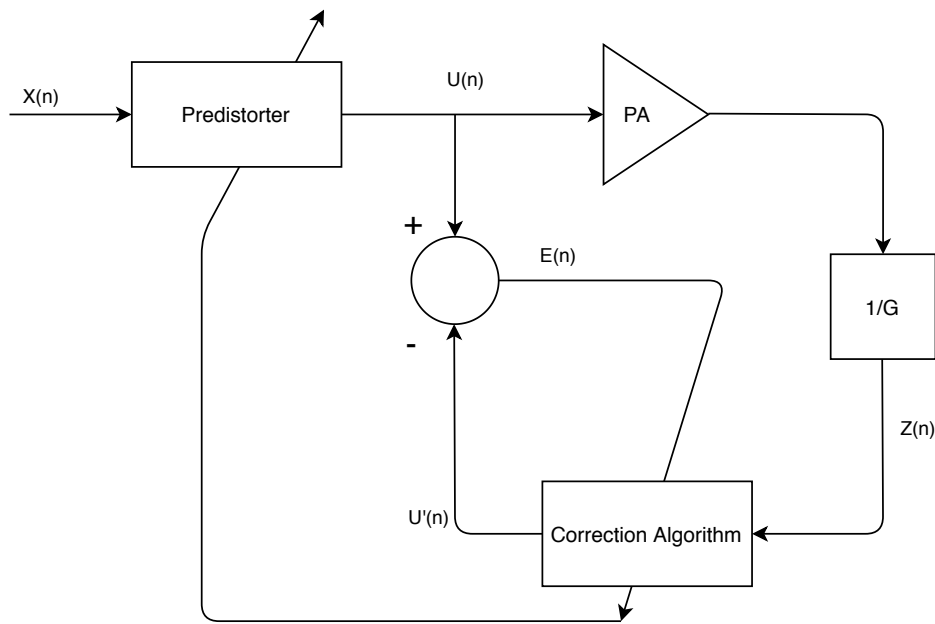


Figure 3.4: Indirect Learning Architecture

3.3 DPD Techniques for Linearizing a Power Amplifier

The PA tends to be non-linear as input increases and reaches the compression zone. These non-linearities can introduce inter modulation distortion, which can create a leakage into the adjacent channel. Hence, the linearization of the PA is very important.

This section aims to provide a basic understanding of some linearization algorithms and implementation techniques. As introduced in chapter 2, there are two types of behavioral models for PA - memoryless and memory models. Some of the well known memoryless algorithms are the Saleh model, Rapp model, etc. In memory models, the most popular algorithms are the Volterra series, Memory Polynomial, LUT based predistorter, and so on. Throughout this thesis, we used one of the most common methods known as LUT based predistorter. The LUT-predistorter technique makes use of LUTs to form the predistorter and built utilizing the memory polynomial model.

3.3.1 Memory Polynomial Model

A memory polynomial model is a simplified form of the Volterra series based model. The Volterra series, as detailed in chapter 2, is a famous tool used to represent the input-output relationship of non-linear systems with memory effects. Figure 3.5 illustrates the Volterra series structure. Nevertheless, the Volterra series introduces a significant number of coefficients resulting in higher evaluation time plus massive resource utilization. Hence, the researchers proposed reduced versions of the Volterra series to model the memory effects and non-linearities of the PA using fewer elements. The memory polynomial is known as one of the most effective reduced versions of the Volterra series. The equation 3.2 expresses the memory polynomial model.

$$y(n) = \sum_{p=0}^P \sum_{m=0}^M h_{p,m} |x(n-m)|^{p-1} x(n-m) \quad (3.2)$$

Where, x and y are the input and output, h is the coefficients, P and N are the order of the polynomial and the memory depth, respectively [23].

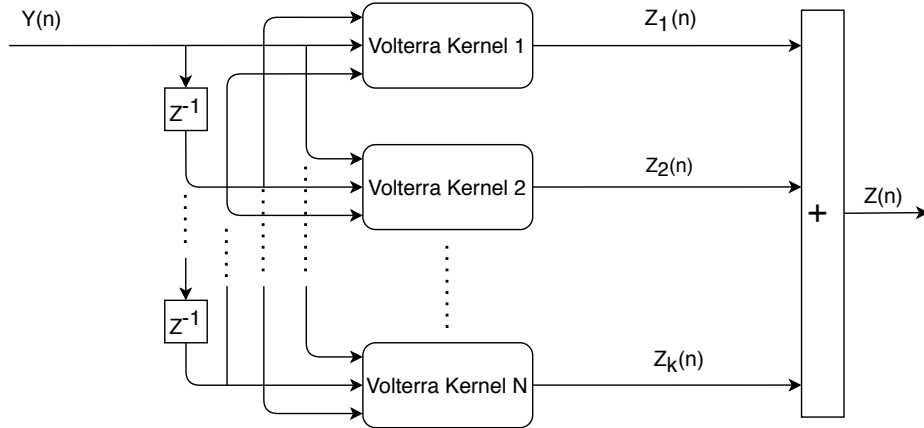


Figure 3.5: Volterra Series Structure

3.4 Look-up Table Based Predistorter

The LUT based predistorter uses look-up tables to form the predistorter. Based on the input signal's amplitude, the input signal will get multiplied with the complex-valued coefficients stored in the LUTs.

The LUTs in this thesis overall, has four columns. The first one corresponds to the bin address of the LUT. The second and third column corresponds to the real and imaginary value of the input signal, and finally, the fourth column corresponds to the LUT instance. There are different ways of indexing the bins of the LUTs, such as squared(power), logarithmic, etc. In this thesis, input signals magnitude is used to calculate the index, i.e., using $I^2 + Q^2$ to calculate the index.

Figure 3.6 illustrates the structure of the LUT based predistorter. The power computation block takes the input signal and feeds the computed power to address generation block. The generated address indexes the bins of the LUT, which feed the coefficients to the complex multiplier. Complex multiplier takes the input signal and multiplies it with the coefficients provided by the LUTs. And in the end, the output signal is the sum of all the memory tap products. Chapter 4 gives an in-detailed explanation of the conventional implementation of LUT based predistorter, the problems with the conventional approach, and the modifications done in this thesis to overcome those problems.

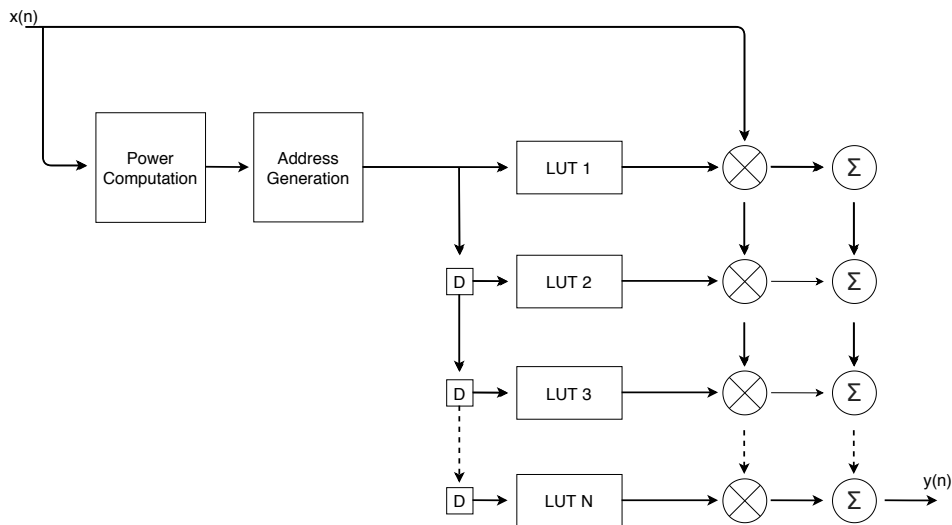


Figure 3.6: Look-up Table Based Predistorter

Design theory and Implementation

4.1 ACPR requirements in this project

As discussed above, ACPR is an important parameter to be considered in the design of communication devices. In this thesis, the design target is -35dB, which is set according to the published 3GPP specification about ACPR[28].

Before implementation, evaluation needs to be done to check if the method that's gonna be realized in hardware can meet the requirement.

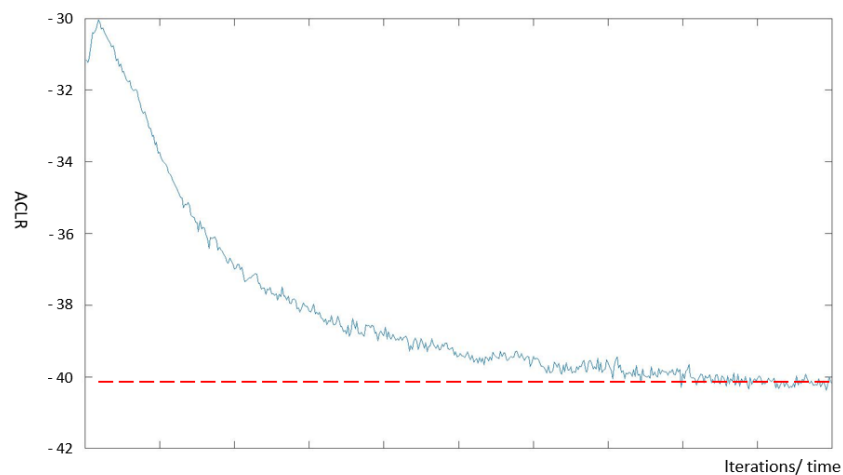


Figure 4.1: ACPR with X LUTs predistorter

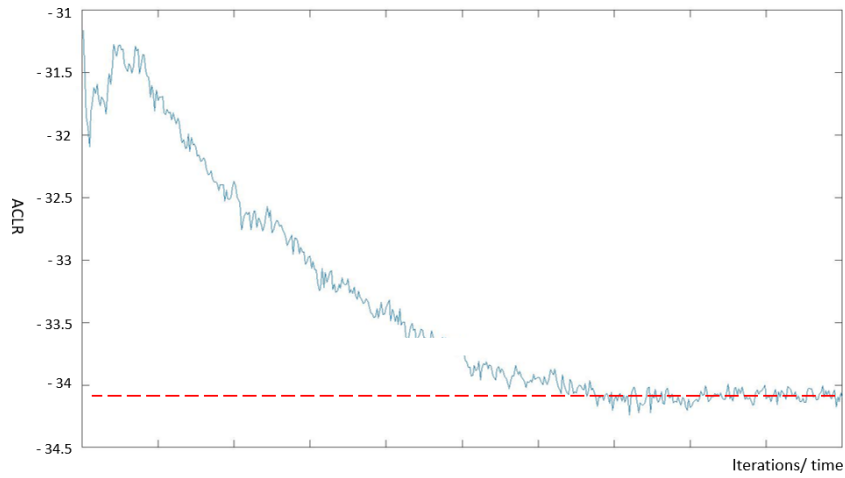


Figure 4.2: ACPR with X-1 LUTs predistorter without modification

As shown in figure 4.1 and figure 4.2, they present the simulation results for X LUTs predistorter and X-1 LUTs predistorter. In the two figures, the X-axis represents the simulation time or the number of iterations, and the y-axis is the ACPR value. In figure 4.1, it shows the ACLR result of X LUTs predistorter simulation, it is obvious that the ACPR will be stable around -40dB, which meets the requirement (red dash line). In figure 4.2, shows the result after removing one LUT directly without any modification. The red dash line shows that the final ACPR will be stable around -34dB, which doesn't meet the requirement.

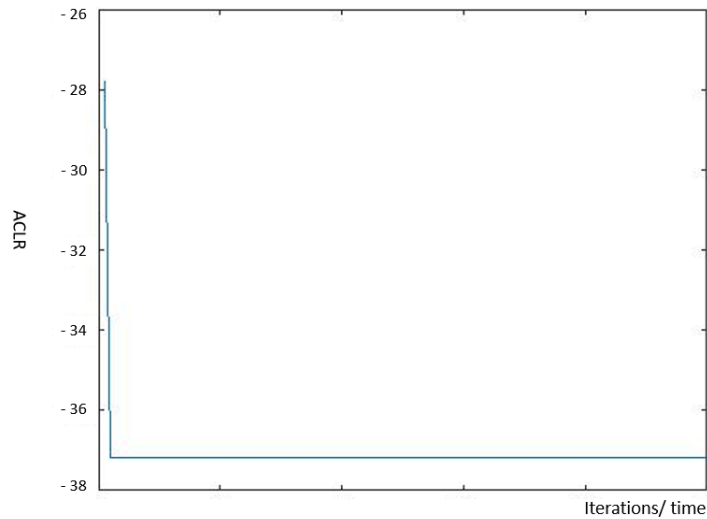


Figure 4.3: ACPR with X-1 LUTs predistorter after modification

Figure 4.3 shows the simulation result of X-1 LUTs predistorter after modifica-

tion. The final ACPR is around -37dB, which meets the requirement. At the same time it is running with fewer LUTs, in return, the total power consumption and area is less than the original one. According to the result present in figure 4.1, figure 4.2 and figure 4.3, the modification method can meet the ACPR requirement. The coming section will introduce the method in more detail.

4.2 Power issues

When talking about chip design, many factors need to be paid attention to, such as cost, area, timing issue and power consumption and so on.

With the continuous upgrading of Silicon on chip (SoC) technology, as shown in Table 4.1, the overall power consumption of the chip has also greatly increased according to the higher level of technology (not only multiplied but also exponentially), so the power consumption budget has gradually become one of the most important design goals[29].

Before proposing methods to optimize power consumption, the first thing to know is how the power consumption is generated, so that corresponding methods could be proposed based on different causes and parameters.

Table 4.1: Power variation according to technology[29]

Node	90nm	65nm	45nm
Dynamic Power per cm^2	1x	1.4x	2x
Static Power per cm^2	1x	2.5x	6.6x
Total Power per cm^2	1x	2x	4x

4.2.1 Static power and dynamic power

The total power consumption in a device is composed of two different types : dynamic power and static power. Dynamic power refers to the power consumption when components are in active state, that is, signals pass through and change values. Static power is when circuit is powered on but no signals are changing[30].

$$\begin{aligned}
 Power_{total} &= Power_{static} + Power_{dynamic} \\
 &= Power_{static} + Power_{switching} + Power_{short-circuit}
 \end{aligned}
 \tag{4.1}$$

Static power

Static power, in Complementary metal-oxide-semiconductor (CMOS) devices, also called off-state leakage or leakage power. It is consumed when all the transistors are off, but still have current going through. There are many causes for static power consumption, three main reasons are listed below [30].

- When Metal Oxide Semiconductor Field Effect Transistor (MOSFET) works in weak inversion region, there will be sub-threshold leakage from drain to source.
- Gate leakage due to gate oxide tunneling.
- Drain junction leakage currents through the reverse-biased drain junctions.

Concluding from the reasons for static power consumption listed above, it is highly correlated with the properties of the device itself. There are many technical methods to reduce static power consumption, such as using some more ideal switching components and applying some high-threshold cells by increasing the thickness of oxide, which can reduce both sub-threshold leakage and tunneling current at the same time [31]. In this thesis, static power consumption is not the main focus, as it highly related to the characteristics of components.

Dynamic power

As illustrates in equation 4.1, dynamic power is made up of two parts, one is switching power and the other one is short-circuit power.

Switching power, shown in figure 4.4(a), takes place during which signal passes CMOS circuits. And it changes logic state of the circuit, charge and discharge the internal capacitance and output node capacitance.

The reason for short-circuit power consumption, presented in figure 4.4(b), is when switching states of the gates, V_{dd} (supply voltage) and ground will be short-circuit connected simultaneously, which means both NMOS and PMOS are on at the same time[32].

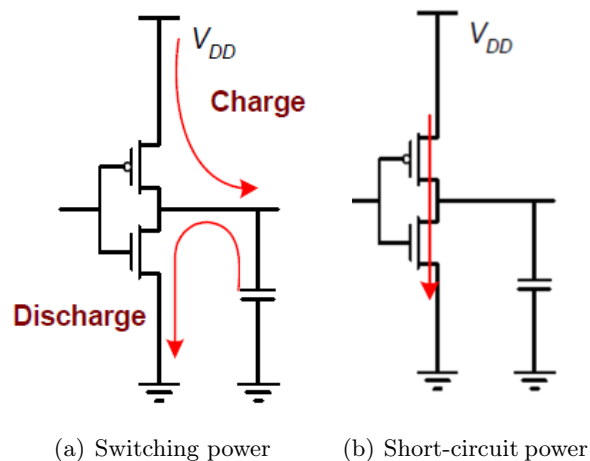


Figure 4.4: Dynamic power—switching power and short-circuit power

In dynamic power, switching power plays a leading role. According to figure

4.4(a), energy used by every transition can be expressed by[29],

$$Energy/transition = C_L \times V_{dd}^2 \quad (4.2)$$

Where C_L stands for the load capacitance and V_{dd} is the supply voltage. Thus, the dynamic power can be described as,

$$\begin{aligned} Power_{dynamic} &= Energy/transition \times f \\ &= C_L \times V_{dd}^2 \times P_{transition} \times f_{clk} \end{aligned} \quad (4.3)$$

Here, in the first line of equation 4.3 f is the frequency of all the transitions. In the second line of equation 4.3, $P_{transition}$ is the possibility of a transition happen and use f_{clk} stands for the system clock. If assume the following equation,

$$C_{eff} = C_L \times P_{transition} \quad (4.4)$$

Then the dynamic power could be written like,

$$Power_{dynamic} = C_{eff} \times V_{dd}^2 \times f_{clk} \quad (4.5)$$

From equation 4.5, it is obvious that the dynamic power highly depends on transitions, or in other words, switching activities and load capacitance.

Except for the switching power introduced above, another part of the dynamic power cannot be ignored as well, that is the internal power consumption as presented in figure 4.4(b).

$$Power_{short-circuit} = T_{short-circuit} \times V_{dd} \times f_{clk} \times I_{peak} \quad (4.6)$$

$$P_{dynamic} = C_{eff} \times V_{dd}^2 \times f_{clk} + T_{short-circuit} \times V_{dd} \times f_{clk} \times I_{peak} \quad (4.7)$$

Where $T_{short-circuit}$ is the total time that short-circuit current lasts and I_{peak} represents the sum of the short-circuit current and the current that charges internal capacitors. Since the short-circuit current lasts for a very short time, the overall dynamic power consumption is basically provided by switching power. Thus, formula 4.7 can be simplified as,

$$P_{dynamic} = C_{eff} \times V_{dd}^2 \times f_{clk} \quad (4.8)$$

4.2.2 Clock gating

Most of the time, for a specific design, data is not input to registers from outside all the time, however, the clock signal does flip in every clock cycle. Clock buffers can generate up to 50% or even more of the dynamic power[30]. The main purpose of clock gating is to prevent providing unnecessary clock signals to the circuit when it is sure that there is no change in the input.

Generally speaking, clock gating is to have an *enable* signal acting as a switch, controlling the *clock* signal on or off. Figure 4.5 shows a conceptual block diagram

block diagram. There is an AND gate added in front of the *clock* pin of a register. When *enable* is set to "1", then the *gatedclk* signal will be "on", otherwise the *gatedclk* signal will not work. When the cell is clock gated, of course it will not consume any dynamic power, but the leakage power is still there.

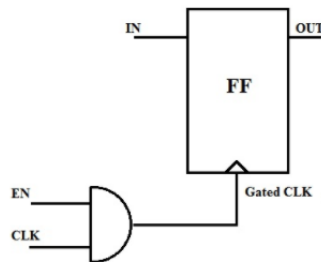


Figure 4.5: Clock gating theory block[32]

However, if the output of AND gate is used to feed the clock path, there will be glitches generated[32]. As there is a certain delay when the signal passes through the routing and the logic unit inside the device. The size of the delay is related to the length of the wire and the number of logic cells, and is also affected by the properties of the device, such as the manufacturing process, operating voltage, and temperature. The signal's high and low level conversion also requires a certain transition time.

Due to these two factors, when the input value of the multi-channel signals change, at the moment the signal changes, the output of the combinational logic is sequential, the results are not generated at the same time. In this process, some incorrect spikes will appear, which are called "glitches".

Basically, as long as the input signals change at the same time, the combinational logic (via internal routing) will inevitably produce glitches. The design method of connecting their outputs directly to the clock pin can lead to serious consequences. Therefore, it must be ensured that the clock input pin which connects to the clock gating signal does not contain any glitches[33].

For this reason, in this thesis, device-specific clock components are used to achieve the desired final circuit implementation and results. This device-specific component is usually included in some specific libraries. Before instantiating the component, the simulator needs to reference this library describing the function of the component to ensure that the simulation runs normally and correctly. The following snippet shows the library declaration.

```
1 library UNISIM;
2 use UNISIM.Vcomponents.all;
```

Inside this library, there are multiple components pre-defined, and a clock buffer called BUFGCE is chosen, which means a global clock buffer with enable. The post-synthesis schematic is shown in figure 4.6.

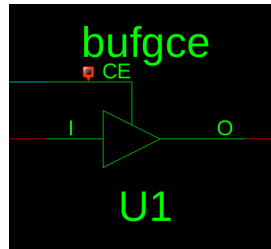


Figure 4.6: Clock buffer schematic

This clock buffer, shown in figure 4.6, is only used for synthesis to FPGA platform. However, when it comes to synthesis to ASIC, the functional library UNISIM cannot be compiled. In this way, the design was adjusted to make sure that the clock enable signal will not come from any combinational logic to avoid glitches as introduced above. And at the same time it can also get the same result as the clock buffer does.

4.3 Machine learning algorithm

In this thesis, gradient descent algorithm is used to obtain the least number of iterations. Gradient, if in a function of a single variable, is actually differentiation of the function, representing the slope of the tangent of the function at a given point.

However, if comes to a function with multiple variables, the gradient is a vector, and the vector has a direction. The direction of the gradient indicates the fastest rising direction of the function at a given point [34].

Generally speaking, gradient descent algorithm is used to find the minimum value of a function. And in this work, it is used to find the minimum number of iterations. In real life, the fastest way to descend a mountain is to find the steepest direction at current location, and then walk down towards that direction. If corresponds this method to mathematical function, the task is to find the gradient of a point at which the function value changes fastest, and then move towards the opposite direction of the gradient. Therefore, if using this method to find the gradient repeatedly, finally it can reach the minimum point[36].

$$\Theta^1 = \Theta^0 - \alpha \nabla J(\Theta) \quad (4.9)$$

In this formula 4.9, J is a function of Θ , the current position is Θ^0 , while the target is to reach the minimum point of the function J from Θ^0 , or can also be called start point. First determine the direction of the function, which is the reverse of the gradient, then move a step, which is the α in formula 4.9. After finishing this step, the function J will reach the point Θ^1 , repeat the above process at Θ^1 then next result can be achieved, which is Θ^2 . After repeating the above process multiple times, it will eventually reach the minimum.

α in equation 4.9 is called the learning rate or step length in the gradient descent algorithm. α cannot be too large or too small. If it is too small, it may

lead to delays in reaching the lowest point. If it is too large, it will miss the minimum value.

4.4 The predistortion module

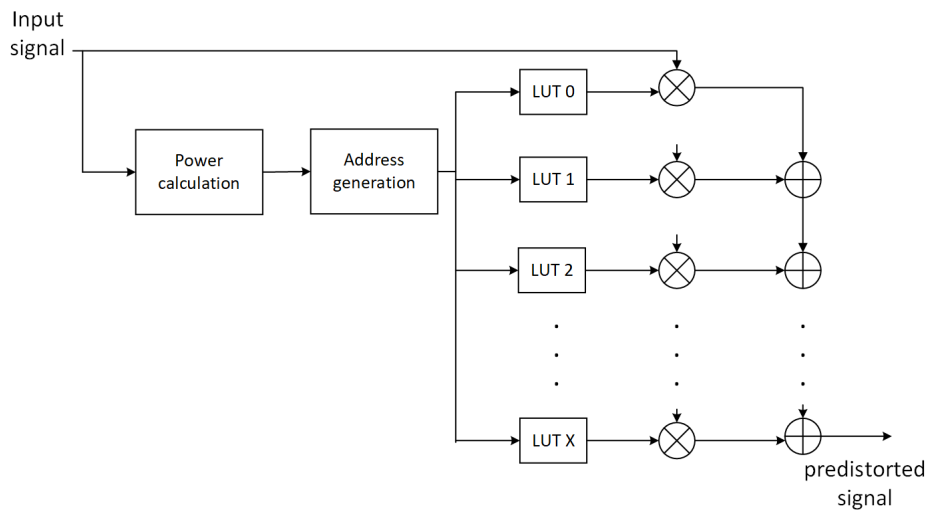


Figure 4.7: Block diagram of predistortion

In this work it is required to implement a modification to predistortion module and then evaluate the power consumption. Block diagram of the predistortion module is shown in figure 4.7. If looks inside the predistorter, there are some LUTs, multipliers and adders and other blocks. Every LUT is a dual port memory, if down scaling LUT bins, the size of the memory can be decreased, in return, the total area and power consumption can be less comparing with the original one.

To get less power, another possible way is to have less of LUTs. As presented in 4.7, each LUT followed by one complex multiplier. Even though multiplication is one of the most commonly used operations in arithmetic, in hardware applications, a series of optimizations are often performed because of power consumption and area occupied by the multiplier itself. Therefore, under the premise of meeting the design requirements, using as few multipliers as possible will lower total power consumption to a certain extent[35].

The input of this predistortion module is a complex signal and output of the module is a predistorted signal to be amplified by PA, that is placed after predistorter in transmitting chain. When the input signal goes inside the predistorter, the very first block is a power calculation. $I^2 + Q^2$ is used as an index of the LUT address. The calculated power value will be the input of next block, the address generation. The main function of address generation block is to determine which LUT bin is going to be chosen, details shown in figure 4.8.

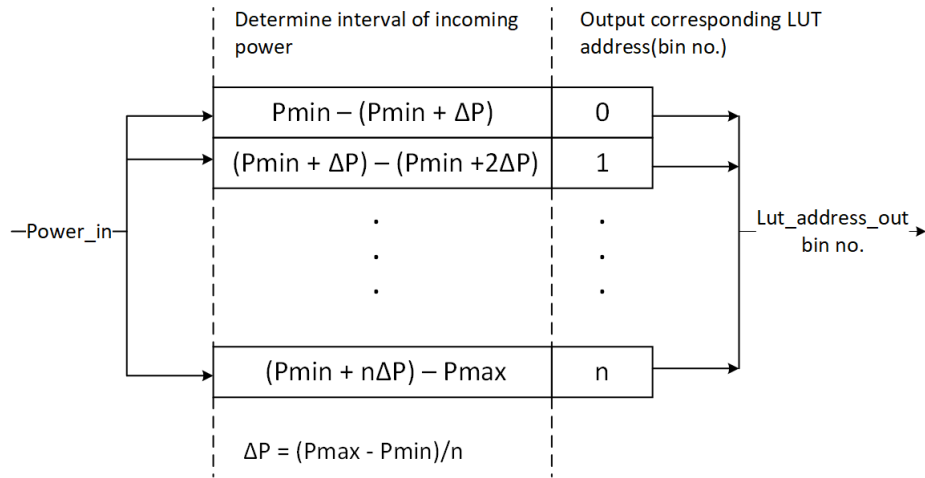


Figure 4.8: Address generator

In address generating block, figure 4.8, the input is $power_{in}$ ($I^2 + Q^2$), output of this block is the number of the bin, or can also be called the LUT address range. Assume there are in total n bins or entries in one LUT. During the initialization phase of this block, the system will detect P_{min} and P_{max} according to the range of the input signal. After determining the input power range (from min to max), divide it by n (the total number of bins inside one LUT), the result represents the power level of each LUT bin. When a specific power value goes in, it will automatically map with the power level and output the LUT bin number.

Focus back to the block diagram of predistorter, figure 4.7. After getting the LUTs coefficients, feed both the coefficients and input signal to the complex multipliers, add then add all the results coming from the multipliers together, the final result is the pre-distorted signal. Previously, the input power range from P_{min} to P_{max} was mentioned, if the instantaneous power value $power_{in}$ is too small, even smaller than P_{min} then the input signal value will ignore the calculation process just mentioned, go directly to the output, and output directly without processing.

4.5 Machine learning steps

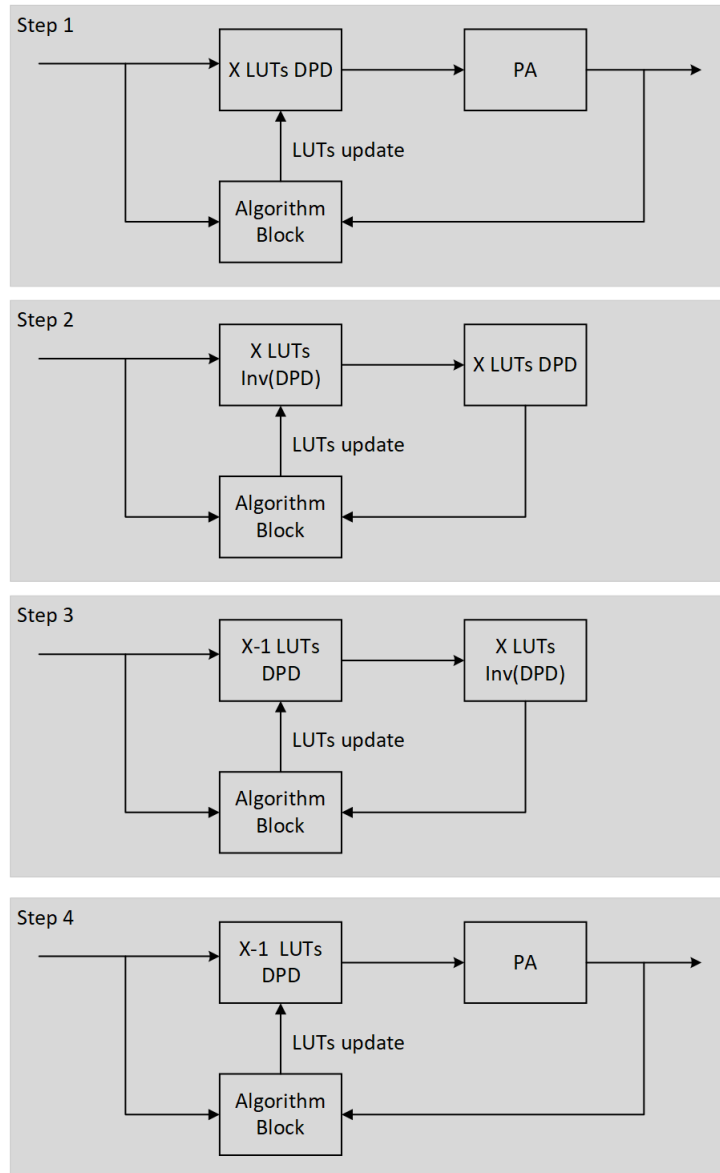


Figure 4.9: Machine learning training steps

As presented in the figure 4.9, there are four stages.

- The first stage is to have X LUT predistorter to start with. PA is only shown in this figure to make the idea clear. The feedback path from the first step is the training algorithm. All the related LUT coefficients can be

generated from the algorithm block.

- After running long enough time, it will come to the second step. In second step, using X LUTs predistorter getting from last step to train the inverse of the X LUTs predistorter.
- For step 3, using the inverse of the X LUTs predistorter which achieved before to train the X-1 LUTs predistorter.
- Step 4 in figure 4.9 shows that an X-1 LUTs predistorter is working and placed in front of a PA, and the PA is same as the one in first stage. At the same time, one less LUT can decrease the power consumption comparing with the original one in step 1, while can also meet the ACLR requirement.

4.6 Hardware implementation

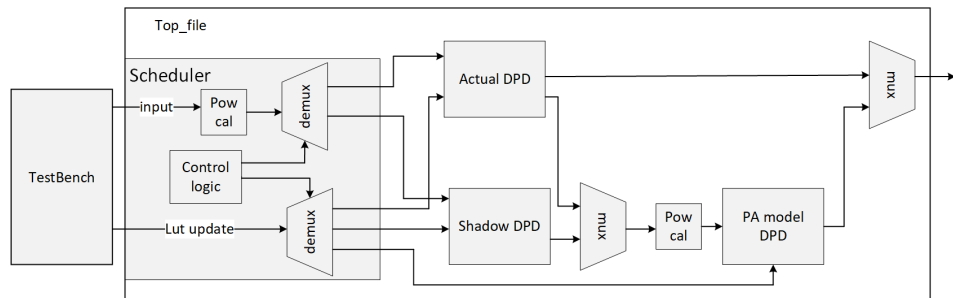


Figure 4.10: Top level block diagram

The figure above shows the top-level block diagram. All the input data and LUT coefficients will come from the testbench, then they will go into the very first block, scheduler. Inside the scheduler, there will be two de-multiplexers (also known as demux). According to the control logic, the demuxes will send the input values and coefficients of a certain period to those relevant modules. One demux is to deliver input to different modules, and the other demux is to serve LUTs updating.

These three predistortion blocks in figure 4.10 are the instantiation of the predistortion module shown in figure 4.7.

As mentioned before, in order to determine the LUT coefficients, the power of the input signal is needed. Because of this, before demux, a power calculation block is necessary. When designing the power calculation module, consideration is given to the number of bits of the input signal and the number of bits of the power pin of the predistortion module to ensure that the $power_{in}$ is within the address generator power range and will not be bypassed.

The control logic module is a state machine, designed according to the machine learning steps mentioned previously, in figure 4.9, and it is a one-way FSM, which automatically stops after reaching the last stage. The trigger condition of each stage is different counters. When a certain counter reaches the threshold, it will automatically jump to the next stage.

The thresholds of each counter is determined according to the gradient descent method mentioned above. The minimum number of iterations that each stage needs to update LUT coefficients so that the power consumption is smallest under the premise of ensuring that the design requirements are met.

As mentioned before, the LUT coefficients are read in from the testbench. A very important condition for the system to operate normally according to the theoretical design is to ensure that the counter accumulation is consistent with the updating speed of reading new data from the testbench.

According to the design, every time when the LUTs complete one round of updating, a trigger signal will be sent out, and this signal will take one clock cycle. The LUTs update process is continuous until the threshold is reached. In this case, when the counters inside the state machine count, these extra clock cycles must be taken into account to ensure that the two parts are synchronized.

5.1 Area report

The approximate area results were obtained by synthesizing the design in 16 nm technology using the Design Vision tool by Synopsys.

Table 5.1 shows the total area of conventional design consisting of 1 DPD. It is similar to Table 5.3, which represents an area of 1 DPD out of 3 DPD's in scheduler design, and only one DPD's area is represented here for scheduler design as all the DPD's in it have similar areas. It can be observed from Table 5.2 that the scheduler design is almost three times more than the conventional design. The overhead area in this design, when compared to the conventional model, is caused by the addition of a top-level block (refer figure 4.10), which contains two more DPD's (PA model DPD and Shadow DPD) apart from the actual DPD, and a scheduler module. However, there are several transmitters(Tx's) in a system, and each Tx gets trained by the same top-level block. In the beginning, the block trains first Tx, upon completion of the training, it switches to train another Tx, and the process repeats till all the Tx's get trained. Therefore, the additional area is negligible as it divides between all the Tx's.

Figure 5.1 shows the pictorial area representation of different blocks for conventional design. It shows that LUTs consume the majority of the area, followed by complex multipliers, which indirectly states that reducing the number of LUTs will save significant area for the design. Figure 5.3 and figure 5.3 represent the area consumption of blocks inside the scheduler design and the blocks inside one of the DPD, respectively.

Table 5.1: Area of Different Blocks inside conventional DPD design (μm)

Blocks	Area (μm)
DPD	16770 μm
→ LUTs	10935 μm
→ Multipliers	3883 μm
→ Adders	57 μm
→ Registers	2.6e-05 μm

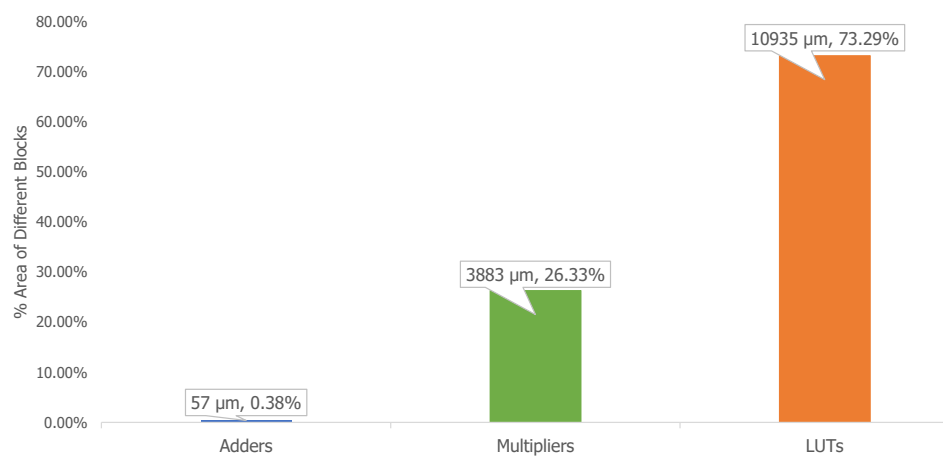


Figure 5.1: Area of Different Blocks inside conventional DPD design (μm)

Table 5.2: Area of Scheduler Design (μm)

Blocks	Area (μm)
DPD Scheduler	52180 μm
→ Actual DPD	16863 μm
→ Shadow DPD	16863 μm
→ PA Model DPD	16863 μm
→ Scheduler	882 μm
→ Power Calculation	677 μm

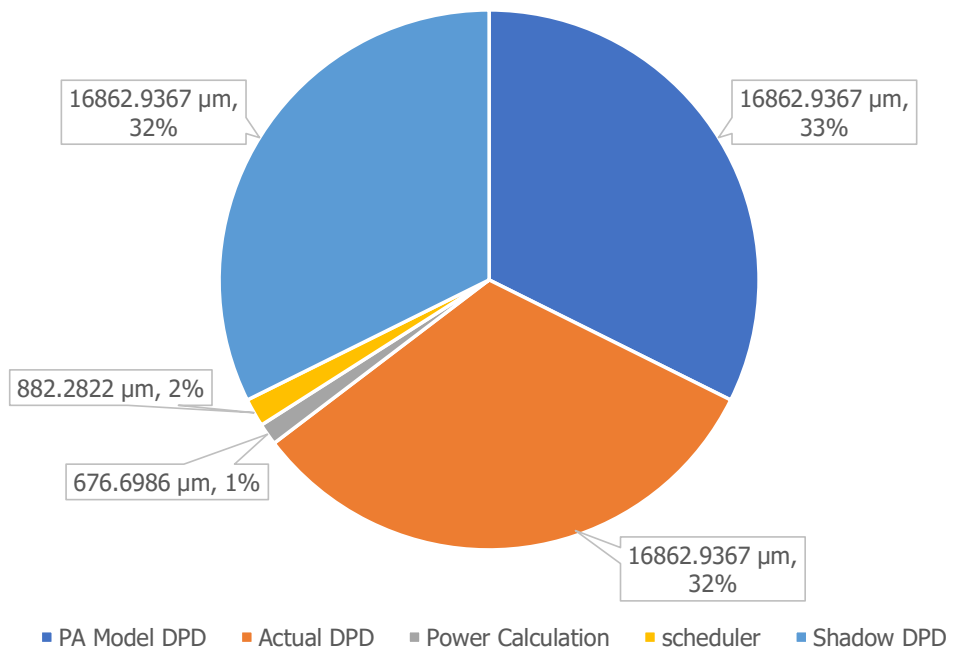
**Figure 5.2:** Area of Scheduler Design (μm)

Table 5.3: Area of Different Blocks One DPD Model for Scheduler and Clock Gated Scheduler Design (μm)

Blocks	Area (μm)
DPD	16863 μm
→ LUTs	10976 μm
→ Multipliers	3883 μm
→ Adders	117 μm
→ Registers	2.6e-05 μm

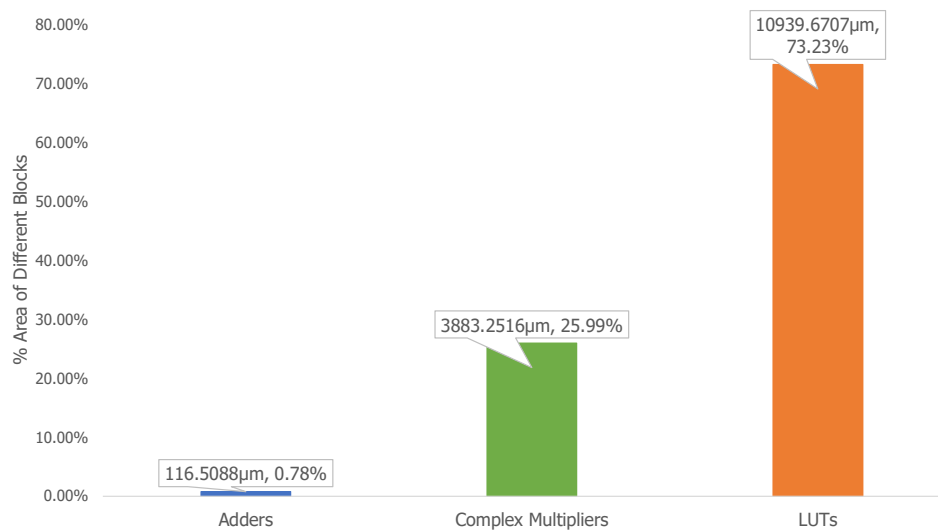


Figure 5.3: Area of Different Blocks inside One DPD Model for Scheduler and Clock Gated Scheduler Design (μm)

5.1.1 Power analysis

Initially, to calculate the power, Ericsson's in-house synthesis and power calculation tool was tried. The tool needs a predefined block to link a design successfully to it. As the tool had a lot of interconnections, creating a new block for it was quite tricky. Hence, this technique was not suitable for farther approaches. Later, energy-based and area-based power calculation methods gave the required power results but were not accurate, and finally, the Prime-Time PX (PTPX) tool gave more precise power results.

In PTPX, at first, the power results were extracted using some basic PTPX scripts, and later an automated process was developed to calculate the power using PTPX. Additionally, this research conducted power calculations using different libraries and compared their results. In conclusion, the 16 nm library with 0.9 voltage gave the best results for this thesis design.

There are three modes in this design, conventional mode, open loop mode, and optimized mode. The conventional mode is similar to the base design that works with 1 DPD X LUTs, and in open loop mode, 3 DPD's are active. Finally, the optimized mode has 1 DPD active with X-1 LUTs.

Table 5.4 shows the power consumption of different blocks in conventional mode. It states that LUTs consume the highest power, followed by multipliers. Hence, this thesis tried to reduce the LUT numbers in a way that it does not affect the current ACPR. In table 5.5, which represents the open loop mode, shows that the total power is more than three times (from figure 5.4 and figure 5.6, 3.3 times) the conventional as all 3 DPD's are active in this mode.

Table 5.6 and Table 5.7 depicts the optimized mode's power consumption of scheduler design and clock gated scheduler design, respectively. The conventional design's power consumption is the base for the comparison. The total power consumption in table 5.6 and table 5.7 is 28.12% and 34.5% less than the conventional mode, respectively (refer figure 5.4 and figure 5.6).

Figure 5.5 and figure 5.7 depicts a pictorial representation of the power events occurring in different modes for the scheduler and clock gated designs, respectively. It can be noticed that optimized mode runs for a longer time than the conventional and open loop mode. Hence, when the average power of the entire system is calculated, the results are less than the base design or the current design.

Table 5.4: Conventional Mode Power report for Scheduler Design and Clock Gated Design (mW)

Blocks	Dynamic Power (mW)	Leak Power(mW)	Total Power(mW)
DPD	5	0.051	5.05
→ LUTs	3.2	0.033	3.25
→ Multipliers	1.01	0.011	1.0224
→ Adders	0.018	0.28e-03	0.018

Table 5.5: Open loop Mode Power report for Scheduler Design and Clock Gated Design(mW)

Blocks	Dynamic Power (mW)	Leak Power(mW)	Total Power(mW)
DPD Scheduler	16.52	0.160	16.7
→ Actual DPD	4.53	0.051	4.58
→ Shadow DPD	4.76	0.051	4.82
→ PA Model DPD	4.43	0.051	4.48
→ Scheduler	0.505	0.004	0.51

Table 5.6: Optimized Mode Power report for Scheduler Design (mW)

Blocks	Dynamic Power (mW)	Leak Power(mW)	Total Power(mW)
DPD	3.579	0.051	3.63
→ LUTs	2.02	0.022	2.04
→ Multipliers	0.83	0.008	0.838
→ Adders	0.014	0.217e-03	0.014

Table 5.7: Optimized Mode Power report for Clock Gated Design (mW)

Blocks	Dynamic Power (mW)	Leak Power(mW)	Total Power(mW)
DPD	3.26	0.051	3.31
→ LUTs	1.86	0.022	1.88
→ Multipliers	0.672	0.008	0.68
→ Adders	0.014	0.217e-03	0.014

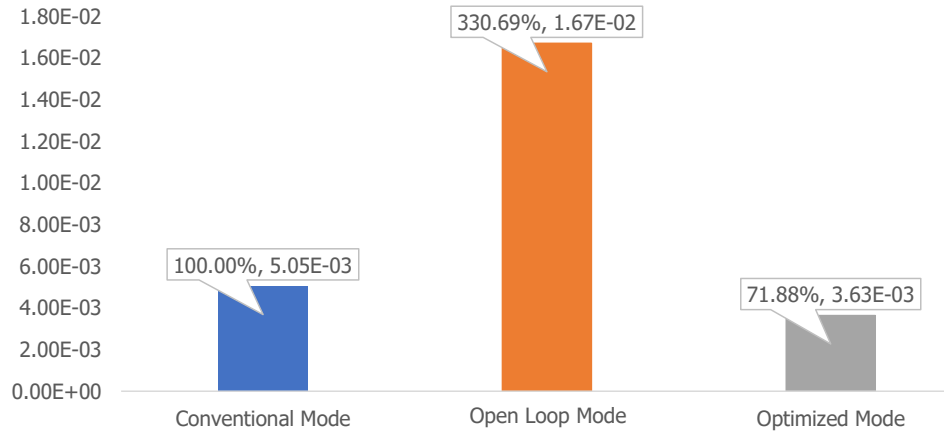


Figure 5.4: Power of Different Blocks inside One DPD Model for Scheduler Design (W)

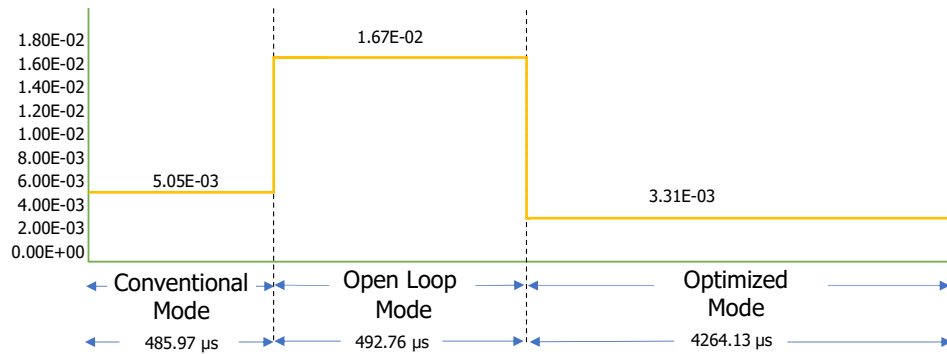


Figure 5.5: Power Results for Scheduler Design (W)

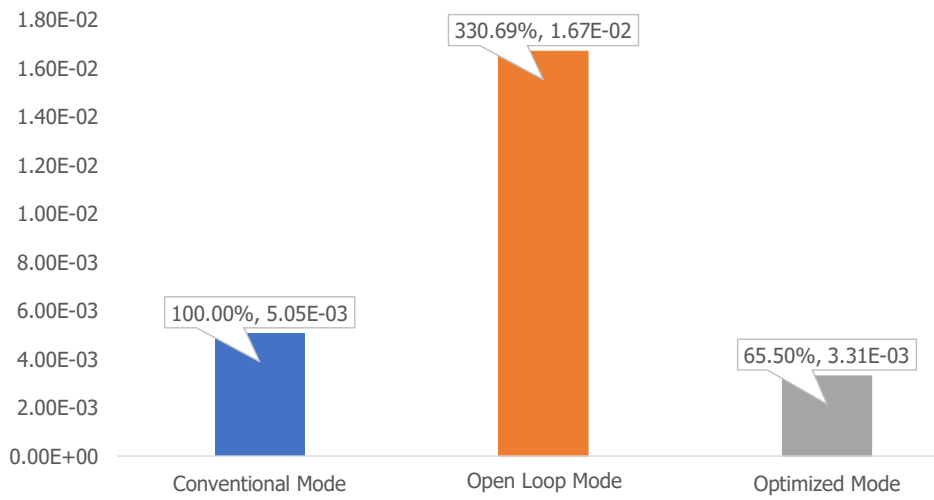


Figure 5.6: Power of Different Blocks inside One DPD Model for Clock Gated Scheduler Design (W)

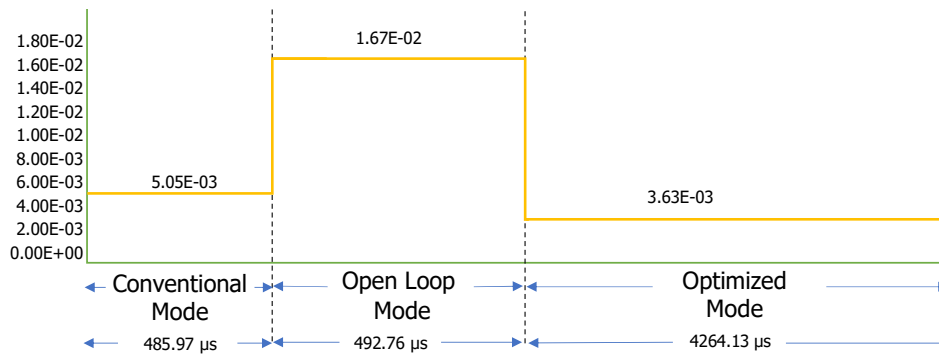


Figure 5.7: Power Results for Clock Gated Scheduler Design (W)

Conclusion and Future Work

6.1 Conclusion

The primary purpose of this thesis was to reduce the power consumption of the LUT based predistorter without affecting the ACPR. The results were compared against a base design with 1 DPD, 3 LUTs. In the very beginning, the study of base design revealed that the LUTs are the significant area and power contributors. Hence, this research aimed to reduce the number of LUTs to lower the area and power consumption. However, the direct downscaling of LUTs affects ACPR. Thus, to tackle this, a scheduler was implemented to train the predistorter to work only with X-1 LUTs, and it also made sure that ACPR was not affected. According to the results shown in the previous chapter, this research saved around 35% and 28% of the power for clock gated design and scheduler design, respectively, while still achieving the same ACPR as X LUTs predistorter.

Therefore, it proves that the design met the primary purpose and expectations. The reduction in the number of LUTs will use less area and consume less power, but at the same time, it usually means reduced accuracy. Nevertheless, the design also met the ACPR requirements(-35dB), which means that although there are distortion and intermodulation products during signal transmission, these effects on the final predistorted signal will not have a high impact on accuracy.

6.2 Future work

- In a real product, the actual DPD will be the one placed in the transmitting chain, the shadow and PA model PDP, shown in figure 4.10, they should be placed inside the adaption block to continuously update and train the LUTs to finally get the step 4 in machine learning steps (figure 4.9).
- For this thesis, simulation about the power consumption based on the design theory has been done, but in real products, there should be more practical issues that need to be considered. For example, there should be some logic in front of the scheduler to switch between training mode and normal operating mode.

- Prototyping the design in actual hardware, currently, the results are based on simulations.
- The scheduler can be clock gated after it trains all the Tx's.

References

- [1] European Commission, *Energy-efficient networks: green by design*, <https://ec.europa.eu/digital-single-market/en/news/energy-efficient-networks-green-design#:~:text=With%20telecommunications%20and%20internet%20use,in%202020%20if%20nothing%20changes>
- [2] John Wood *Digital Pre-Distortion of RF Power Amplifiers: Progress to Date and Future Challenges* Maxim Integrated Inc. , San Jose, CA USA, 978-1-4799-8275-2/15/2015 IEEE to be corrected
- [3] Tony R. Kuphaldt, *Lessons In Electric Circuits, Volume I – DC, Fifth Edition*, <https://www.ibiblio.org/kuphaldt/electricCircuits/DC/index.html>
- [4] I. Sarkas, D. Mavridis, M. Papamichail and G. Papadopoulos, *Large and small signal distortion analysis using modified Volterra series*, *Analog Integr Circ Sig Process* (2008) 54:133–142 DOI : 10.1007/s10470-007-9110-4
- [5] Walter Ciccognuni, Pa010 Colantonio, Franco Giannini, Ernest0 Limifi and Massimiliano Rossi *AM/AM and AM/PM power amplifier characterisation technique* Conference Paper · June 2004, DOI: 10.1109/MIKON.2004.1357126 · Source: IEEE Xplore
- [6] Emanuele Tolomei, *Design and implementation of a software predistorter for amplifier linearization in OFDM-based SDR systems* Universita Di Pisa, 2016
- [7] Mahmoud Alizadeh and Daniel Rönnowa, *A two-tone test for characterizing nonlinear dynamic effects of radio frequency amplifiers in different amplitude regions* *Measurement* Volume 89, July 2016, Pages 273-279 <https://doi.org/10.1016/j.measurement.2016.04.027>
- [8] David Hall, *Understanding Intermodulation Distortion Measurements* *ElectronicDesign* OCT 09, 2013 <https://www.electronicdesign.com/technologies/communications/article/21798494/understanding-intermodulation-distortion-measurements>
- [9] Tony J. Roupheal *Transceiver System Analysis and Design Parameters* RF and Digital Signal Processing for Software-Defined Radio, 2009, Pages 161-198 <https://doi.org/10.1016/B978-0-7506-8210-7.00006-0>

- [10] Ziad El-Khatib, Leonard MacEachern and Samy A. Mahmoud, *Modulation Schemes Effect on RF Power Amplifier Nonlinearity and RFPA Linearization Techniques* chapter2 ISBN:978-1-4614-0271-8
- [11] Hank Zumbahlen, *Linear Circuit Design Handbook* Figure 1-60 <https://doi.org/10.1016/B978-0-7506-8703-4.X0001-6>
- [12] Raviv Raich and G. Tong Zhou, *On the modeling of memory nonlinear effects of power amplifiers for communication applications* Proceedings of 2002 IEEE 10th Digital Signal Processing Workshop and the 2nd Signal Processing Education Workshop, pages 7–10, Oct. 2002.
- [13] Karam G, Sari H, *Analysis of pre-distortion, equalization and ISI cancellation techniques in digital radio systems with nonlinear transmit amplifiers* IEEE Transactions on Communication, 1989, 37(12): 1245-1253.
- [14] Jinghua Yin, Bo Su and Dongxing Wang, *The Nonlinearities of Memoryless Power Amplifier and Model of Predistorter* Advanced Materials Research (Volume 981) pp40-45, July 2014, DOI: <https://doi.org/10.4028/www.scientific.net/AMR.981.40>
- [15] Schreurs, M. O'Droma, A.A. Goacher, and M. Gadringer. *RF Power Amplifier Behavioral Modeling* Cambridge University Press, New York, NY, USA, 1st edition, 2008. ISBN 0521881730, 9780521881739. Cited on pages 19, 20 and 24.
- [16] Ghannouchi F.M. and Hammi O, *Behavioral modeling and predistortion* IEEE Microwave Magazine, 10(7) :52–64, 2009.
- [17] Jeonghyeon Cha, Ildu Kim et al, *Memory Effect Minimization and Wide Instantaneous Bandwidth Operation of a Base Station Power Amplifier* Microwave Journal, January 2007
- [18] Vito Volterra *Theory of Functionals and of Integrals and Integro-Differential Equations*. Madrid 1927 (Spanish), translated version reprinted New York: Dover Publications, 1959.
- [19] Erik Andersson and Christian Olsson, *Linearization of Power Amplifier using Digital Predistortion, Implementation on FPGA* 2014 Linköpings universitet
- [20] Ibrahim Can Sezgin, *Different Digital Predistortion Techniques for Power Amplifier Linearization*. Department of Electrical and Information Technology, Faculty of Engineering, LTH, Lund University, 2016.
- [21] Md Zahidul Islam Shahin and Himanshu Gaur, *Efficient DPD Coefficient Extraction For Compensating Antenna Crosstalk And Mismatch Effects In Advanced Antenna System*. Department of Electrical and Information Technology, Faculty of Engineering, LTH, Lund University, 2018.
- [22] Takao Inoue, *Digital Predistortion (DPD) Design-to-Prototype Framework for PA's*. National Instruments, AWR Corporation, 2013.
- [23] L. Ding, G. T. Zhou, D. R. Morgan, Z. Ma, J. S. Kenney, J. Kim, and C. R. Giardina, *A Robust Digital Baseband Predistorter Constructed Using Memory Polynomials*, IEEE Trans. Commun., Vol. 52 No 1., 2014

- [24] Jessica Chani-Cahuana, *Digital Predistortion for the Linearization of Power Amplifiers*. Department of Signals and Systems, Chalmers University of Technology, Göteborg, Sweden, 2015.
- [25] Michail Isaakidis, *Self-organizing Maps for Digital Pre-distortion*. Department of Electrical and Information Technology, Faculty of Engineering, LTH, Lund University, 2020.
- [26] Engineer Ambitiously, *Optimizing IP3 and ACPR Measurements* http://www.ni.com/pdf/en/Optimizing_IP3_and_ACPR_Measurements_With_the_PXIe_5668R.pdf
- [27] *What degrades ACPR performance to greater extent: OIP3 or P1dB?* Habib Ur Rahman Mohammed, Apr 2, 2013, https://e2e.ti.com/blogs_/b/analogwire/archive/2013/04/02/what-degrades-acpr-performance-to-greater-extent-oip3-or-p1db
- [28] 3rd Generation Partnership Project (3GPP); Technical Specification Group (TSG) RAN WG4; UTRA (BS) TDD; Radio transmission and Reception <https://www.3gpp.org/>
- [29] Michael Keating, David Flynn, Robert Aitken, Robert Aitken, Kaijian Shi, *Low Power Methodology Manual For System-on-Chip Design*, Library of Congress Control Number: 2007928355, ISBN 978-0-387-71818-7
- [30] Nikolic Rabaey, Chandrakasan, *Digital Integrated Circuits: A Design Perspective* PEARSON INDIA; 2nd edition (2016), ISBN-10: 9332573921
- [31] Amit Agarwal, Saibal Mukhopadhyay, Arijit Raychowdhury, Kaushik Roy, Chris H. Kim, *Leakage power analysis and reduction for nanoscale circuits*, IEEE Micro, Volume: 26 , Issue: 2 , March-April 2006, Pages: 68 - 80
- [32] N. Srinivasan, N.S.Prakash, S. D.Sivaranjani, D.SwethaSri and B.T.Sundari *Power Reduction by Clock Gating Technique* Procedia Technology Volume 21, 2015, Pages 631-635
- [33] *Need for clock gating checks - need for glitchless clock propagation* <https://vlsiuniverse.blogspot.com/search/label/glitch%20free%20clock\protect\penalty\z%20gating>
- [34] Parul Pandey, *Understanding the Mathematics behind Gradient Descent* Mar 18, 2019 <https://towardsdatascience.com/understanding-the-mathematics-behind-gradient-descent-dde5dc9be06e>
- [35] Mohan Shoba and Rangaswamy Nakkeeran *Energy and area efficient hierarchy multiplier architecture based on Vedic mathematics and GDI logic* Engineering Science and Technology, an International Journal, Volume 20, Issue 1, February 2017, Pages 321-331
- [36] Qian Ning *On the momentum term in gradient descent learning algorithms* Neural Networks. 12 (1): 145–151. CiteSeerX 10.1.1.57.5612. doi:10.1016/S0893-6080(98)00116-6, 8 May 2014.



LUND
UNIVERSITY

Series of Master's theses
Department of Electrical and Information Technology
LU/LTH-EIT 2020-788
<http://www.eit.lth.se>