TKK Dissertations 136
Espoo 2008

DIGITAL PREDISTORTION LINEARIZATION METHODS FOR RF POWER AMPLIFIERS

Doctoral Dissertation

Ilari Teikari

Helsinki University of Technology
Faculty of Electronics, Communications and Automation
Department of Micro and Nanosciences

TKK Dissertations 136
Espoo 2008

DIGITAL PREDISTORTION LINEARIZATION METHODS FOR RF POWER AMPLIFIERS

Doctoral Dissertation

Ilari Teikari

Dissertation for the degree of Doctor of Science in Technology to be presented with due permission of the Faculty of Electronics, Communications and Automation for public examination and debate in Auditorium S4 at Helsinki University of Technology (Espoo, Finland) on the 26th of September, 2008, at 12 noon.

Helsinki University of Technology
Faculty of Electronics, Communications and Automation
Department of Micro and Nanosciences

Teknillinen korkeakoulu
Elektroniikan, tietoliikenteen ja automaation tiedekunta
Mikro- ja nanotekniikan laitos

Distribution:
Helsinki University of Technology
Faculty of Electronics, Communications and Automation
Department of Micro and Nanosciences
P.O. Box 3000
FI - 02015 TKK
FINLAND
URL: http://www.ecdl.tkk.fi/
Tel. +358-9-451 2271
Fax +358-9-451 2269

ISBN 978-951-22-9545-6
ISBN 978-951-22-9546-3 (PDF)
ISSN 1795-2239
ISSN 1795-4584 (PDF)
URL: http://lib.tkk.fi/Diss/2008/isbn9789512295463/

TKK-DISS-2504

Multiprint Oy
Espoo 2008

ABSTRACT OF DOCTORAL DISSERTATION	HELSINKI UNIVERSITY OF TECHNOLOGY P.O. BOX 1000, FI-02015 TKK http://www.tkk.fi

Author Ilari Teikari

Name of the dissertation
Digital predistortion linearization methods for RF power amplifiers

Manuscript submitted 29.03.2008	Manuscript revised 24.08.2008

Date of the defence 26.09.2008

☒ Monograph	□ Article dissertation (summary + original articles)

Faculty	Faculty of Electronics, Communications and Automation
Department	Department of Micro and Nanosciences
Field of research	Electronic circuit design
Opponent(s)	Prof. Timo Rahkonen
Supervisor	Prof. Kari Halonen
Instructor	

Abstract

Even though high linearity is crucial in modern mobile communications, it is not desirable to use the most linear power amplifier types due to their poor efficiency. Predistortion is a commonly used, fairly simple and robust method for improving linearity of power amplifiers (PA). This thesis will investigate digital RF and baseband PA predistortion methods. A digital RF predistortion system uses an analog predistortion element prior to the power amplifier that is controlled by digital circuitry to compensate for the PA nonlinearity. One problem with RF predistorter is its sensitivity to delays between the control signals generated by the digital circuitry and the RF signal. This thesis presents a delay compensation method that can be implemented with digital circuitry, thus the delay being much smaller than the previously used analog methods. Another implementation issue that affects the performance of the RF predistorter, are analog envelope detectors that are required for generating the control signals for the digital circuitry. Three commonly used detection methods, power, linear diode and logarithmic detector, are compared. The linear diode detector was shown to be the most versatile. By using a lookup table, the power and logarithmic detectors can be linearized so that their performance comes close to the linear diode, but the biasing is easier.

Design of an RF predistorter and the measurement results are presented. The designed RF predistorter was implemented to linearize a class AB PA with 22 dB gain and a 18 kHz 16QAM signal at 420 MHz. The digital algorithm is implemented with an FPGA. The predistorter was able to achieve 10 dB improvement in the ACP.

The thesis also investigates the design of a baseband predistorter, that is implemented using the complex gain predistortion method. The effects of nonlinear quadrature modulator errors on the predistortion are discussed. Simulation and measurement results of the predistorter are presented. The designed baseband predistorter was implemented to linearize a class AB PA with 50 dB gain and an 18 kHz 16QAM signal at 400 MHz. The digital algorithm is implemented with an FPGA. The predistortion improves the ACP by 15 dB.

Finally, predistortion function generation methods applicable to both RF and baseband predistorter implementations are discussed. Some improvements to these methods are suggested and simulations with and without the suggested improvements are presented. The simulations show that the suggested improvements are able to improve the ACP and reduce the time required for convergence.

Keywords Digital predistortion, Power amplifiers, Linearization, FPGA, Radio frequency

ISBN (printed)	978-951-22-9545-6	ISSN (printed)	1795-2239
ISBN (pdf)	978-951-22-9546-3	ISSN (pdf)	1795-4584
Language	English	Number of pages	209

Publisher Helsinki University of Technology, Department of Micro and Nanosciences

Print distribution Helsinki University of Technology, Department of Micro and Nanosciences

☒ The dissertation can be read at http://lib.tkk.fi/Diss/2008/isbn9789512295463/

VÄITÖSKIRJAN TIIVISTELMÄ	TEKNILLINEN KORKEAKOULU PL 1000, 02015 TKK http://www.tkk.fi

Tekijä Ilari Teikari

Väitöskirjan nimi
Radiotaajuustehovahvistimien digitaaliset esisärötysmenetelmät

Käsikirjoituksen päivämäärä 29.03.2008	Korjatun käsikirjoituksen päivämäärä 24.08.2008

Väitöstilaisuuden ajankohta 26.09.2008

☒ Monografia	□ Yhdistelmäväitöskirja (yhteenveto + erillisartikkelit)

Tiedekunta	Elektroniikan, tietoliikenteen ja automaation tiedekunta
Laitos	Mikro- ja nanotekniikan laitos
Tutkimusala	Piiritekniikka
Vastaväittäjä(t)	Prof. Timo Rahkonen
Työn valvoja	Prof. Kari Halonen
Työn ohjaaja	

Tiivistelmä

Nykyaikaiset langattomat tietoliikennejärjestelmät vaativat lähettimeltä korkeaa lineaarisuutta. Tarvittavilla lineaarisilla tehovahvistimilla on kuitenkin huono hyötysuhde. Esisärötys on yleinen menetelmä tehokkaampien epälineaaristen vahvistimien linearisoimiseksi. Tämä työ perehtyy digitaalisiin RF- ja kantataajuusesisärötysmenetelmiin.

Digitaalinen RF-esisärötysjärjestelmä käyttää tehovahvistimen linearisointiin vahvistinta edeltävää digitaalisesti ohjattua analogista kytkentää, joka kompensoi tehovahvistimen epälineaarisuuden. Merkittävä ongelma RF-esisärötysmenetelmissä on esisärötyskytkennän kantataajuisten ohjaussignaalien ja vahvistettavan RF-signaalin viive-ero. Tässä työssä esitetään viiveen kompensointiin menetelmä, joka voidaan toteuttaa kokonaan digitaalisesti. Tämä toteutus on huomattavasti kompaktimpi kuin aiemmin käytetyt analogiset menetelmät.

Merkittävä RF-esisärötyksen tehokkuuteen vaikuttava tekijä ovat analogiset verhokäyränilmaisimet, joita tarvitaan esisärötyksen ohjaussignaalien luomiseen. Työssä vertaillaan kolmea yleisesti käytettyä ilmaisintyyppiä. Lineaarinen ilmaisin osoittautui yleiskäyttöisimmäksi tyypiksi. Muidenkin ilmaisinten toiminta on parannettavissa DSP-linearisointipiirillä.

Työssä selostetaan RF-esisäröttimen suunnitelu ja esitetään toteutetun järjestelmän mittaustuloksia. Toteutettu esisärötin suunniteltiin linearisoimaan AB-luokan tehovahvistin, jonka vahvistus oli 22 dB. Vahvistettavan signaalin kantoaaltotaajuus oli 420 MHz ja kaistanleveys 18 kHz. Digitaalinen esisärötysalgoritmi toteutettiin FPGA-piirillä. Mittauksissa saavutettiin 10 dB parannus viereisen kaistan häiriötehossa.

Työssä tutkitaan myös digitaalisen kantataajuusesisärötysjärjestelmän suunnittelua. Esisärötys toteutettiin käyttäen kompleksisen vahvistuksen menetelmää. Työssä perehdytään myös kvadratuurimodulaattorin epälineaarisuuden vaikutukseen kantataajuusesisärötykseen. Kantataajuusesisärötysjärjestelmä suunniteltiin linearisoimaan AB-luokan tehovahvistin, jonka vahvistus oli 50 dB. Vahvistettavan signaalin kantoaaltotaajuus oli 400 MHz ja kaistanleveys 18 kHz. Digitaalinen esisärötysalgoritmi toteutettiin FPGA-piirillä. Mittauksissa saavutettiin 15 dB parannus viereisen kaistan häiriötehossa.

Työn lopussa perehdytään menetelmiin esisärötysfunktion luomiseksi. Nämä menetelmät soveltuvat sekä kantataajuus- että RF-esisärötyksessä käytettäviksi. Näihin menetelmiin esitetään parannusehdotuksia, jotka varmennetaan simulaatioin. Simulaatiot osoittivat parannusten kohentavan saavutettavaa lineaarisuutta sekä lyhentävän adaptiivisen esisärötysalgoritmin asettumisaikaa.

Asiasanat Digitaalinen esisärötys, Tehovahvistimet, Linearisointi, FPGA, Radiotaajuus

ISBN (painettu)	978-951-22-9545-6	ISSN (painettu)	1795-2239
ISBN (pdf)	978-951-22-9546-3	ISSN (pdf)	1795-4584
Kieli	Englanti	Sivumäärä	209

Julkaisija	Teknillinen korkeakoulu, Mikro- ja nanotekniikan laitos

Painetun väitöskirjan jakelu Teknillinen korkeakoulu, Mikro- ja nanotekniikan laitos

☒ Luettavissa verkossa osoitteessa http://lib.tkk.fi/Diss/2008/isbn9789512295463/

Preface

The research for this thesis was done in the Electronic Circuit Design Laboratory, Helsinki University of Technology. The study started as a project funded by TEKES and Satel Oy in 2003 and was taken to conclusion with the help of funding from Graduate School in Electronics, Telecommunications and Automation (GETA). Without the graduate school position in GETA it would have been very hard to finish the thesis work, so I express my gratitude to GETA, its former Director Prof. Iiro Hartimo and to GETA coordinator Marja Leppäharju for their support. I also want to thank Finnish Foundation for Technology Promotion and Nokia Foundation for additional funding for the research. The thesis is part of the research done at the Centre of Exellence in Smart Radios and Wireless Research - SMARAD in Helsinki University of Technology.

I'm grateful to Professor Kari Halonen for offering me the possibility to carry out my research at the Electronic Circuit Design Laboratory and for his guidance and support. I also want to thank my instructors Dr. Jouko Vankka and Prof. Hannu Tenhunen for the guidance they have given me, as well as the pre-examiners of this thesis, Professors Lars Sundström and Máirtín O'Droma for their comments and effort.

Thanks to Jaakko Ajanki and Miika Koivisto, especially for their help with the RF hardware implementation. I am also grateful my room mates at the Electronic Circuit Design Laboratory, Lasse Aaltonen, Yang Fan, Jaakko Ketola, Johan Sommarek and Olli Väänänen, for their help and for ensuring that the work never got boring. I also want to thank Sanna Heikkinen, Artturi Kaila, Marko Kosunen, Jonne Lindeberg, Mika Länsirinne, Ville Saari, Mikko Talonen, Helena Yllö and everybody else I bothered during my thesis work in Electronic Circuit Design Laboratory.

Finally, I want to thank my parents and my sister for their important support that helped me to get to this point and Taru for helping with the finishing touches of the thesis and just being there.

Ilari Teikari

Helsinki, September 2008

Symbols and Abbreviations

$(\cdot)$	any function
$(\cdot)^*$	complex conjugate
μ	adaptive filter convergence coefficient
ω	angular frequency
ω_{rf}	carrier angular frequency
ω_{env}	maximum angular frequency component of the baseband envelope
a_n	nth polynomial distortion coefficient of the power amplifier
a_{PAn}	nth polynomial distortion coefficient of the power amplifier
a_{PDn}	nth polynomial distortion coefficient of the predistorter
A_{PA}	power amplifier gain function
A_{pd}	predistorter gain function
AC	alternating current
ACP	adjacent channel power (in dBc)
A/D	analog to digital
AM-AM	amplitude to amplitude distortion
AM-PM	amplitude to phase distortion
BAW	bulk acoustic wave
BER	bit error rate
CDMA	code division multiple access
D	the prediction amount at a digital predistorter

D/A	digital to analog
dBc	decibels relative to carrier power
DC	direct current
DQPSK	differential quadrature phase shift keying
DSP	digital signal processing
EDGE	Enhanced Data rates for GSM Evolution
EE&R	Envelope elimination and restoration
EVM	Error vector magnitude
FET	field effect transistor
FIR	finite impulse response
f_{clk}	clock frequency
f_{env}	maximum frequency component of the baseband envelope
f_{rf}	carrier frequency
GSM	Groupe Special Mobile
hd3	third order harmonic distortion
H-N	Heinonen-Neuvo
$\Im(x)$	imaginary part of x
I	in-phase
IF	intermediate frequency
IIP3	third intermodulation intercept point
IIR	infinite impulse response
LAN	local area network
LINC	linear amplification using nonlinear components
LMS	least mean squares
LO	local oscillator
LUT	look up table

MSE	mean square error
Niter	number of iterations
NLMS	normalized least mean squares
N_{LUT}	Number of LUT address bits, LUT size = $2^{N_{LUT}}$
N-M	Nelder-Mead
NMT	nordic mobile telephone
OFDM	orthogonal frequency division multiplexing
OPAMP	operational amplifier
OSR	oversampling ratio
PA	power amplifier
PA1	power amplifier model with nonlinearities mainly at small amplitudes
PA2	power amplifier model with nonlinearities at both high and low amplitudes
PA3	power amplifier model with nonlinearities mainly at large amplitudes
PAE	power added efficiency
PAR	peak to average ratio
pchip	piecewise cubic Hermite polynomial
PD	predistorter
PDF	probability density function
Φ_{PD}	predistorter phase distortion function
Φ_{PA}	power amplifier phase distortion function
PM-AM	phase to amplitude distortion
PM-PM	phase to phase distortion
Q	quadrature
QAM	quadrature amplitude modulation
*N*QAM	*N* level QAM
QDM	quadrature demodulator

QM	quadrature modulator
QPSK	quadrature phase shift keying
$\Re(x)$	real part of x
RAM	random access memory
RF	radio frequency
RLS	recursive least squares
RMS	root mean square
ROM	read only memory
RRC	root raised cosine
SB	signal band
SNR	signal to noise ratio
SAW	surface acoustic wave
TETRA	terrestrial trunked radio
THD	total harmonic distortion
TV	television
TWT	traveling wave tube
t_{env}	envelope period, $t_{env} = \frac{1}{f_{env}}$
WLAN	wireless local area network
v_{in}	predistorter input voltage
v_{out}	power amplifier output voltage
v_{pd}	predistorter output voltage/PA input voltage

Contents

Chapter 1

Introduction

1.1 RF transmitter

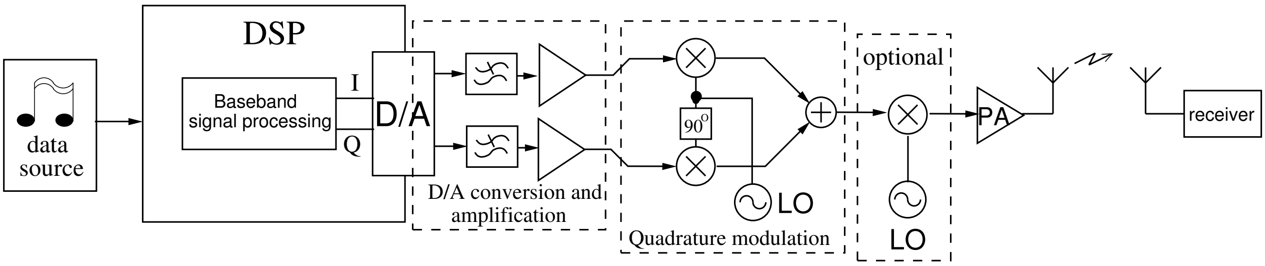


Figure 1.1: RF transmitter

An RF transmitter is an important building block of a communications system. It converts the baseband signals containing the data to be transferred through the transport medium to the receiver. Figure 1.1 shows a block diagram of an RF transmitter. Nowadays the baseband signal processing is usually performed digitally, and thus first the baseband signals have to be digital-to-analog (D/A) converted before feeding to the transmitter. The conversion also requires reconstruction filters to remove unwanted frequency components after the conversion. After that, also baseband amplification may be required. Often the baseband signals are in quadrature form, which allows generation phase, frequency and amplitude modulated signals. To combine these signals into a single analog signal for transmission, they are combined and converted into intermediate frequency (IF) or directly to RF using a quadrature modulator. This conversion can also be accomplished using frequency and amplitude modulators, depending on the modulation method used. If IF is used, the signal has still to be upconverted to RF. After this, the signal is amplified with a power amplifier (PA) for transmission and then fed to the transport medium.

1.2 The need for linear transmitters in modern communications systems

During the last few decades, the amount of data transferred between data terminals has increased significantly due to the popularity of Internet and other computer networks. In recent years, ever-larger amounts of this data are transferred over wireless data transfer channels, such as WLAN or digital mobile networks. This is due to the capability of mobile data terminals to generate high-resolution photographs and video and to send these from one terminal to another through wireless networks. Also mobile connections to the Internet are becoming more common.

This ever-increasing wireless data traffic makes the limited wireless frequency spectrum more and more congested. Previously it was common to use frequency modulation in wireless communications (e.g. radio, NMT and GSM) due to the ability of the modulation method to withstand noise, interference and nonlinearities [1]. However, due to their nonlinearity, they require wide bandwidth [2] and the spectral efficiency (the amount of data transferred in the used bandwidth) is poor.

Recently the variable amplitude modulation methods have again become more common in telecommunications systems and standards due to their better spectral efficiency compared to the frequency and phase modulation methods [3]. The variable amplitude methods usually use both the phase and amplitude to carry the message and are often implemented using baseband quadrature signals. This increases the data carrying ability per signal bandwidth.

However, since the information is stored in the signal amplitude, variable amplitude modulation methods are very sensitive to disturbances that affect the amplitude of the signal, such as nonlinear amplification. Nonlinearities cause errors in the data, but also spread the spectrum of the signal, thus disturbing the adjacent channels. This is very undesirable when the frequency spectrum is tightly packed with signals from different terminals.

Linear amplification can be achieved by proper design of the PA [4]. However, the most linear PA types are also the most inefficient. The energy efficiency, on the other hand, is a very important factor in mobile data terminals, which have to use a battery as a power source. Furthermore, the PA is one of the most power-consuming parts in communications devices. Thus one has to make a choice between linearity and power consumption.

This thesis will discuss digital predistortion, a method that can be used to reduce the nonlinearity of some more inefficient PA types, thus reducing the trade off between linearity and efficiency.

1.3 Variable amplitude modulation methods

As discussed in the previous section, the variable amplitude modulation methods, despite their sensitivity to nonlinearities, are common in modern communications systems. Examples of these are N-QAM, used, for example, in digital TV broadcasting, satellite communications and mobile data transfer, CDMA used in cell phones and OFDM used in wireless LAN devices and digital TV broadcasting. In addition to these modulations, also constant amplitude modulation schemes can become variable amplitude signals when filtered to limit the bandwidth. An example of these is a DQPSK modulated signal filtered with a root raised cosine filter. These methods do not suffer from the data corruption due to distortion, but nevertheless suffer from spectral spreading, which may interfere with the signals on the adjacent channels.

When upmixed to RF, all these modulations can be described with formula [2]

$$x(t) = A(t)sin(2\pi ft + \phi(t)), \tag{1.1}$$

where $A(t)$ is the part of the modulating function transmitted by the amplitude, $\phi(t)$ is the part of the modulating function transmitted by the phase and f is the carrier frequency. This can be further transformed into orthogonal signal presentation [2]

$$\begin{aligned} x(t) &= A(t)cos(\phi(t))sin(2\pi ft) + A(t)sin(\phi(t))cos(2\pi ft) \\ &= I(t)sin(2\pi ft) + Q(t)cos(2\pi ft) \qquad (1.2) \end{aligned}$$

The baseband signals $I(t)$ and $Q(t)$ contain the modulated data to be transmitted and are multiplied with the carrier wave and its $\pi/2$ phase shifted version.

The modulation data points create a constellation of amplitude values and at the receiver the received data samples are referred to this constellation to determine the most probable transmitted data values. Figure 1.2 shows the transmitted constellation for 16QAM modulation, X's are the data points and the decision regions are marked with dashed lines. If the distortion causes a data point to fall into other decision region, it is misinterpreted at the receiver [2].

Equations 1.1 and 1.2 are quite demanding for numerical calculations and simulations due to the fact that the carrier frequency is usually much higher than the baseband frequency. This means that the sampling frequency used in the simulations has to be very high in order to accommodate the carrier signal without aliasing but the actual data signal changes very slowly, thus requiring numerous iterations to get enough data to draw any conclusions [5]. It is mostly the case that one is actually interested in the effects that affect only the vicinity of the carrier frequency and the rest of the spectrum is actually filtered away in the transmitter and the receiver [5]. It is therefore usually enough to inspect the baseband equivalent of the RF signal and thus (1.2) can be written in complex form

$$x(t) = I(t) + jQ(t)$$

where $I(t)$ and $Q(t)$ are the baseband modulated signals and the imaginary unit j represents the orthogonal function [5]. Using this simplification, it is possible to significantly reduce the complexity of the numerical calculations. This complex baseband format will be used throughout this thesis.

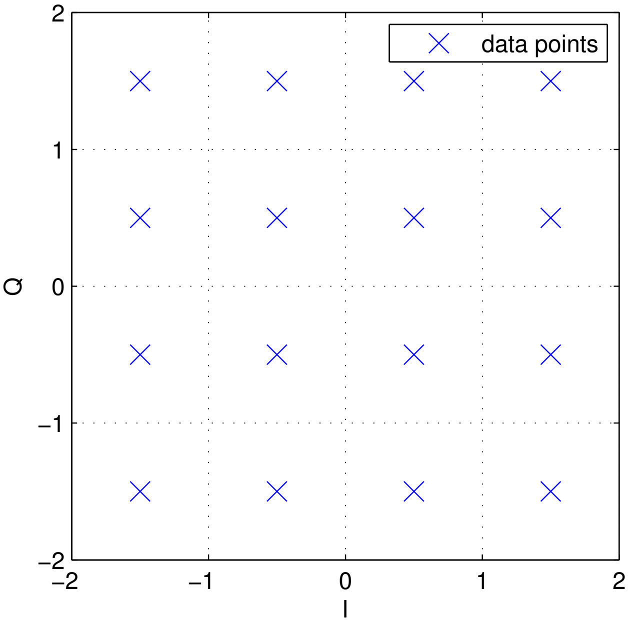


Figure 1.2: 16QAM constellation

1.4 Outline of the thesis

This thesis can be divided into four parts.

The first part consists of Chapters 1-3 and 10, which respectively give an introduction to the subject and conclude the thesis. Chapter 2 deals with the PAs and their effect on variable amplitude signals. Common PA types will be presented and, at the end of the chapter, the effect of memory in the PAs on distortion is discussed. Chapter 3 gives an introduction to PA linearization methods and discusses different predistortion methods in more detail. A survey of published predistorters is presented at the end of the chapter.

The second part consists of Chapters 4-7 that concentrate on RF-predistortion. Chapter 4 discusses the basics of RF-predistortion and presents commonly used RF-predistortion structures. Chapter 5 presents results of studies of the effect of control signal delays on RF-predistortion systems and presents a novel, fully digital method to reduce the effect of the delays. Chapter 6 discusses the effect of filtering and envelope detection on the RF-predistortion systems, neither of which have been studied previously. Chapter 7 presents the RF-predistortion system implemented for this thesis. The system is the first to use adaptive phase and amplitude predistortion function that is updated on the basis of the time-domain measurements of the PA output signal. Simulation and measurement results of the system are presented and improvements to the adaptation algorithm is suggested according to the results.

The third part consists of Chapter 8 and deals with baseband predistortion systems. This chapter gives a description of different baseband predistortion system implementations and describes the design of a complex gain predistorter. It also describes the linear quadrature errors and methods to compensate them and compares adaptive methods to find the correct values for the compensation functions. The chapter also investigates the effect of nonlinear quadrature modulator errors on the baseband predistorter. Finally simulation and measurement results of the implemented predistorter are presented.

The fourth and final part consists of Chapter 9, which discusses predistortion function generation methods applicable to both RF and baseband predistorter implementations. This chapter discusses the effect of lookup-table (LUT) size quantization and indexing on digital predistorters. Different methods to fill the LUT are investigated as well as methods to update the LUT. Finally some improvements to these methods are suggested and simulations using genetic algorithm and the Nelder-Mead algorithm with and without the suggested improvements are presented.

1.5 Specifications of the transmitters and predistorters investigated in the thesis

To measure the hardware implementations of the designed predistorters, two transmitter (PA) chains had to be implemented, one for RF predistortion and one for baseband predistortion. Separate transmitter implementations for the baseband and RF predistorters were required, since the RF predistorter requires only the PA chain and analog predistortion elements to be implemented whereas the baseband predistorter additionally requires quadrature modulators and demodulators and for transforming the orthogonal baseband signals to RF and back to baseband.

The carrier frequencies of the systems were decided to be on the 380 MHz to 430 MHz frequency band that is used by the TETRA digital mobile radio [6] and also for satellite communications [7]. The actual carrier frequencies used in the predistortion systems were decided based on the optimal frequency ranges of the parts easily available for the implementation of the predistortion systems. This led to selecting the carrier frequency of the RF predistorter to be 420 MHz and the carrier frequency of the baseband predistorter to be 400 MHz. However since the operational frequency range of the RF predistorter components reached also 400 MHz, the carrier frequency in the RF predistorter was later changed to this to match to the baseband predistorter. The different transmitter paths was also the reason that the PA chains used in the RF and baseband predistorters had different gains.

Both TETRA and the satellite communications have very limited bandwidth allowance [6] and thus they require very linear transmitters to adhere to the strict adjacent channel interference specifications.

The signal bandwidth mainly used in the simulations and measurements was selected according to the TETRA specification to be 18 kHz. However, since the operation of the predistorter is also dependent on the bandwidth of the signal, also narrower and wider bandwidths were used to test the operation of the predistorter in these situations.

The TETRA standard uses $\frac{\pi}{4}$DQPSK signal modulation to transmit the data. This modulation transfers 2 bits per data symbol. We however chose to use QAM modulations for our implementations. The QAM modulated signals have more amplitude variation and more closely spaced quantization levels than the DQPSK modulation, thus being more sensitive to nonlinearities and more challenging modulation to linearized, the possibility to use QAM modulations would also increase the spectral efficiency of the system. In the simulations, in the first measurements of the RF-predistortion system and in the measurements 16QAM modulation was used. To make the effect of the implemented predictor circuit more visible in the measured spectrum, $\frac{\pi}{4}$DQPSK modulation was used in these measurements. Finally the modulation method used in the RF predistortion system was changed to 32QAM to further increase the spectral efficiency.

1.6 Research goals

One of the reasons that started the study that led to this thesis was to investigate if a simple predistorter could be used to replace the more complicated linearization methods used in TETRA systems. The RF predistortion seemed to have the most potential to reduce the complexity of the linearization of the transmitter so it was chosen as the main predistortion method to be studied.

The RF predistortion also offers a possibility to be used as an universal linearization module that requires very little information on the workings of the transmitter chain to be linearized. An universal predistortion module could enable production of efficient and linear power amplifiers that would contain the predistorter but look just like a normal power amplifier from the hardware designers point of view or a simple predistortion module that could very simply be added to a design to improve the linearity and efficiency of a power amplifier. As a result one of the goals of this thesis was chosen to be to study the RF predistortion and examine limitations and problems that restricted the usability and efficiency of the RF predistorter and if possible to develop solutions to these problems. Chapters 4 to 7 concentrate on the development of the RF predistortion system.

The baseband predistorter was studied as an possible alternative to the RF predistortion to be used in the linearization of the TETRA system. However as the development potential of the baseband predistorter in memoryless predistorter implementations was evaluated to be lesser than the RF predistorter and the results were not very promising, it was given less effort. However during the study some new findings were done, which are discussed in Chapter 8 in addition to the implementation of the baseband predistorter.

During the research several issues were encountered and results found that were could be applied to both RF and baseband predistorters. These are discussed in Chapter 9.

1.7 Research contribution

This thesis concentrates on studying two common digital predistortion systems, namely the RF-predistortion [8–10] and complex gain predistortion [11]. The both predistorters were implemented in hardware. The digital predistortion algorithms, and the analog and digital baseband circuitry of the systems used to control the the predistorters were implemented by the author. The

RF transmitters to be linearized plus the RF interface required for connecting the predistorters to the transmitter, including PAs, upmixers, power dividers and other RF parts, were designed by Miika Koivisto. The studies were conducted under the supervision and guidance of Dr. Jouko Vankka and Prof. Kari Halonen.

The implemented RF-predistortion system is reminiscent of the one presented in [10]; however, the phase modulator control in the system implemented by the author is adaptive and updated according to the phase difference between the PA input and output, measured using an analog phase detector. The system was designed originally designed for 420 MHz carrier frequency and 18 kHz signal bandwidth. The design is described in [12–15] and in Chapter 7.

The main research contributions for the RF-predistortion system are the study of the effect of the delay on the control signals of the predistortion element and the effect of the envelope detectors and the envelope filtering to the operation of the RF predistorter.

The study of the control signal delays included investigating various methods for implementing a digital predictor that would be able to compensate the delays. The author conducted simulations and calculations of the effect of the delays and implemented a polynomial predictive filter [16] to the RF-predistortion system. The study of the delays and the predictors and simulation and measurement results were published in [17, 18]; this study is extended in Chapter 5 of this thesis.

The results of the study of the effect of the envelope detectors and filters were published in [19]; this study is extended in Chapter 6 of this thesis. An improved LUT update algorithm for the RF predistorter was also developed by the author and is presented in Chapter 7.

The baseband complex gain predistorter [11] was implemented to 400 MHz carrier frequency and 18 kHz signal bandwidth. A quadrature modulator correction circuit [20] was also implemented. During the design of the complex gain predistorter the author investigated the effectiveness of different quadrature modulator correction circuit update methods and the effect of nonlinear quadrature modulator errors on the complex gain predistortion. The results were published in [21, 22] and in Chapter 8.

Finally, the author compared different LUT generation and interpolation methods for digital predistorters and developed an improved frequency-domain LUT update method for RF predistorters based on the comparison. The results were published in [23] and expanded in Chapter 9.

Chapter 2

Power amplifier

2.1 Introduction

The power amplifier is one of the most important parts of an RF transmitter and it is usually the largest single contributor to the power consumption of an RF transmitter and thus it would be desirable to maximize the efficiency. However, the more efficient power amplifier configuration is used the more nonlinear it usually is. This poses a problem in modern communications systems which usually use variable amplitude modulation methods [3].

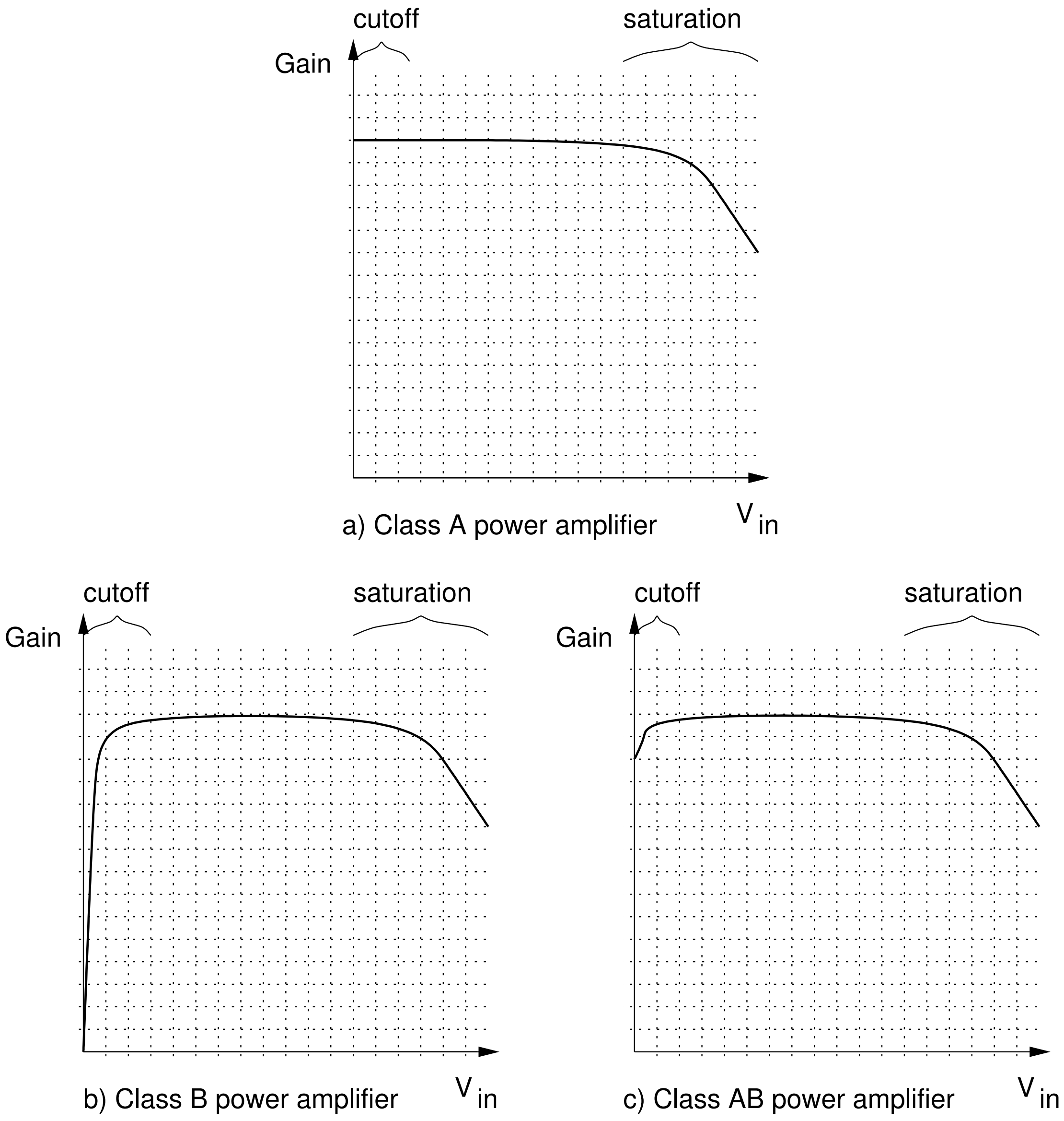
This chapter will introduce the most commonly used PA types and discuss the figures of merit of the PAs that are important for the linearization of PAs. Although the PAs consist of electronic components that can be simulated using a circuit simulator such as SPICE, the distortion of the PA can also be approximated using simplified computational higher level models. This reduces the time required for simulations, giving still much information of the distortion. Further these models allow more general simulations and calculations without having to know the exact component values. Commonly used computational models for PA distortion are presented in this chapter as well as the models used in the simulations in this thesis.

Although this thesis concentrates on memoryless PA models and predistortion, the memory is always present in real PA and may is some cases become a very significant effect. To give some insight to the operation and effects of the memory in PAs, it is briefly discussed in the end of the chapter.

2.2 Common power amplifier types

There are several different types of PAs that differ by their linearity and efficiency. Class A, B and AB amplifiers are fairly linear amplifiers that have from low to moderate efficiencies. These amplifiers conduct both current and voltage at least half of the signal cycle. Class C amplifiers sacrifice linearity to gain efficiency; they do this by reducing the portion of the signal cycle that both the current and voltage are conducting. Finally, there are class D, E, F and S PAs that switch the PA on and off so that the current and/or voltage have square waveform. These PAs are very efficient and nonlinear. In this thesis, the focus is on A, B and AB PAs that are commonly used with variable amplitude modulation methods due to their linearity and that are linear enough to be predistortable.

A class A amplifier has only nonlinearity at high amplitudes [4], since the transistors are used far from cutoff. Class AB and B amplifiers, on the other hand, have transistors biased in such a way that the transistors come near cutoff at low signal amplitudes [4, 24] and there they exhibit nonlinearities also. Finally, if a class AB or B amplifier is used with a large back-off signal the nonlinearities are mainly at low amplitudes.



a) Class A power amplifier

b) Class B power amplifier

c) Class AB power amplifier

Figure 2.1: Transfer functions of different power amplifier types

2.3 Figures of merit for power amplifiers

A very important figure of merit for the power amplifier is its efficiency which tells us how much power is wasted when a signal is amplified. The worse the efficiency, the more the amplifier heats up and the less the battery life. Often the input power is subtracted from the output power when the efficiency is calculated to avoid misleading results in some cases, this modified efficiency is called power added efficiency (PAE) and is defined as [25]

$$PAE = \frac{P_{out} - P_{in}}{P_{source}} * 100\%. \tag{2.1}$$

PAE will be used in this thesis as the measure of power amplifier efficiency.

Other significant figure of merit of a power amplifier is its nonlinearity. There are several methods to present the nonlinearity of a PA, such as third order input intercept point (IIP3), total harmonic distortion (THD) and adjacent channel power (ACP). In this thesis the ACP will be used as the measure of the nonlinearity. ACP is defined as the part of the signal power that lands on the adjacent signal band in relation to the signal power on the signal band. Figure 2.2 shows an example of the definition of adjacent channels of a broad band signal. The actual bandwidths and the spacings of the channels and thus the value of ACP depends on the communications standard that is used, which also defines the limits for the maximum ACP allowed. However, in general, ACP can be defined as

$$ACP = 10 \lg \frac{P_{adjacent}}{P_{signal}} dBc. \tag{2.2}$$

In this thesis, the signal band and adjacent channel definitions for the TETRA standard are used. This thesis also uses the power of the third-order intermodulation results compared to the signal power as a measure of nonlinearity in the case of a two-tone test.

Figure 2.3 shows the simulated third order distortion of a two tone signal compared to the signal power and the PAE of a class A [4] power amplifier as a function of the input signal power. The figure shows how the PAE of the amplifier increases as the input power increases and the amplifier is driven closer to saturation and, at the same time the third order distortion power increases. Previously when constant amplitude signal modulation methods were widely used this behavior was acceptable, but with the recently more common variable amplitude modulation methods this poses a problem.

2.4 The effect of distortion on variable amplitude signals

The use of the signal amplitude to carry the information causes problems when combined with nonlinear power amplifiers that compress or expand the signal. This means that, if the nonlinearity is strong enough, the data points move out of or near the borders of the decision regions used at the receiver to demodulate the signal back to digital data symbols. This increases the bit error rate. Figure 2.4 show how the constellation of a 16QAM signal compresses when the

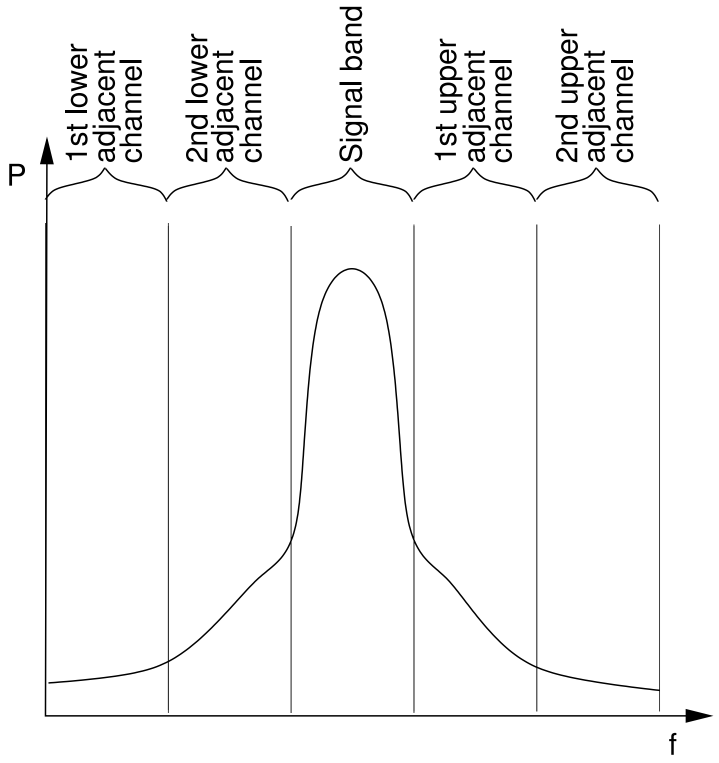

1st lower adjacent channel
2nd lower adjacent channel
Signal band
1st upper adjacent channel
2nd upper adjacent channel
P
f

Figure 2.2: Definition of the adjacent channels

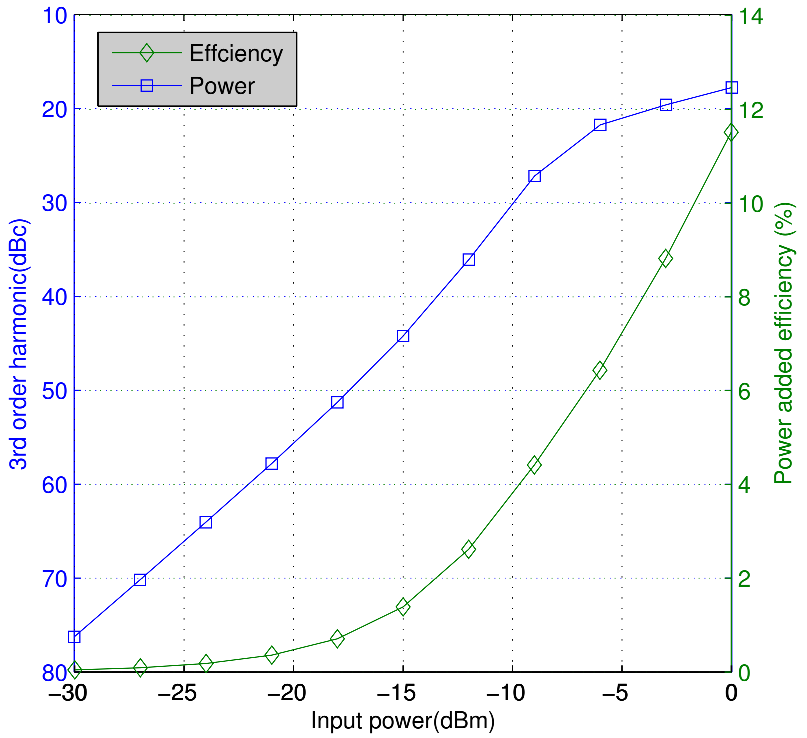


Figure 2.3: Third order distortion and PAE as a function of the input power

signal is driven through a nonlinear amplifier. Especially the corner values move very near to the borders of the decision region. When noise is added, there is a very great chance that the value will be misinterpreted. However, the amount of compression shown in the figure requires quite large distortion levels and, in low noise transmission paths, some amount of distortion might be tolerable.

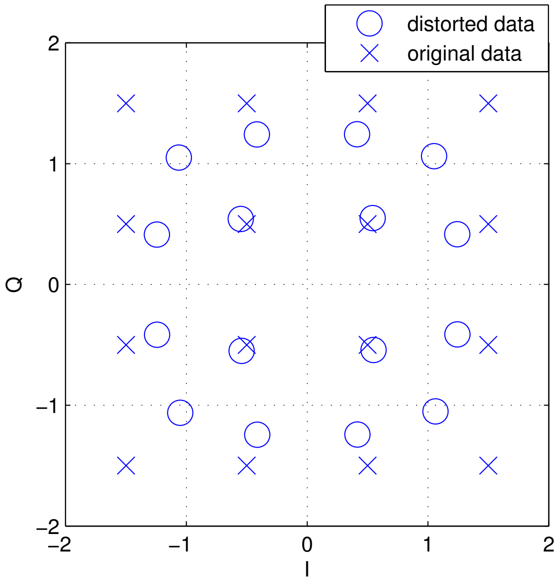


Figure 2.4: The effect of a nonlinear power amplifier on the constellation of a 16QAM signal

However, the distortion also has an effect on the spectrum of the signal, which reduces the signal's tolerance to the nonlinearity. When a broadband signal is fed to a nonlinearity, the spectral components mix with each other and generate new spectral components out of the signal band. This can be seen very easily by inserting a two-tone signal

$$f(t) = \cos((\omega + \Delta\omega)t) + \cos((\omega - \Delta\omega)t) \tag{2.3}$$

into a third order nonlinearity

$$f_{out}(t) = f(t) + af(t)^3 \tag{2.4}$$

We get as the output signal

$$\begin{aligned} f_{out}(t) = \quad & (1 + \tfrac{9a}{4})f(t) + \tfrac{a}{4}\cos(3wt - 3dwt) + \tfrac{a}{4}\cos(3wt + 3dwt) \\ & + \tfrac{3a}{4}(\cos(3wt - dwt) + \cos(3wt + dwt) + \cos(wt - 3dwt) + \cos(wt + 3dwt)) \end{aligned}$$

where we can see the generation of new spectral components to frequencies near to, but outside, the original signal band as well as components near to the third harmonic of the carrier frequency. This can be generalized to amplification of a broad-band signal, where we have an infinite number of tones in a limited bandwidth. Now the intermodulations of these tones fall both on the signal band and out of the signal band as with the two tone case. This effect is called spectral spreading or regrowth. The effect is illustrated in Figure 2.5.

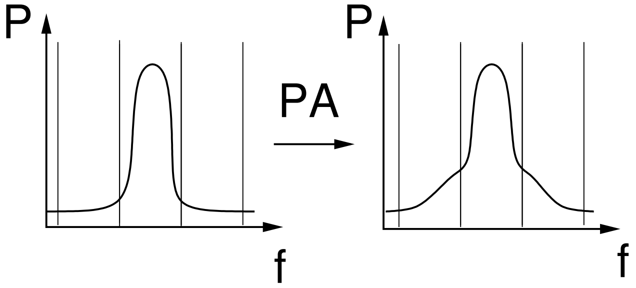


Figure 2.5: The spectral regrowth

If there was only one transmitter using a single transmission path at a time, there would not be a problem. However, as said above the variable amplitude modulation methods are used because of their spectral efficiency and to enhance the most of this efficiency, several signals should be transmitted on adjacent signal bands as close as possible to each other, thus also allowing a good overall spectral efficiency. Now, when these closely spaced signals are driven through a nonlinear amplifier, the spectral regrowth causes them to start interfering with each other, increasing the in-band noise BER of the other signals. There are several solutions to this problem. We can use a more linear and inefficient amplifier, thus increasing heating problems and power consumption, or change the modulation scheme to a more robust, but usually spectrally less efficient one, or increase the spacing of the signals in frequency-domain so that the interference reduces but also so the spectral efficiency is reduced (Figure 2.6). However, usually the signal spacing and the modulation methods are defined by the communications standard that is being used. It also defines the maximum ACP and thus the only design parameter the end product designer has is the implementation of the power amplifier.

So, to minimize the power consumption, it would be profitable if some signal processing method could be used to reduce the nonlinearity of the more efficient power amplifiers without significantly affecting the efficiency. This thesis concentrates on one of the main methods to implement this kind of signal processing function. Chapter 3 gives an overview of the other methods for linearization of nonlinear PAs also.

2.5 Power amplifier models

To be able to calculate and simulate the effect of the power amplifier on the transmitted signal, one has to model the power amplifier somehow. The most straightforward and accurate method would be to use the transistor level model of the power amplifier. There are, however, several problems with this method. Often when using commercial power amplifiers, the transistor level model is not available; the use of such a model requires the use of a transistor level simulator which can be very slow. Another point is that theoretic calculations are difficult. Furthermore, the transistor level model is not able to model temperature effects, which are important when studying PAs with memory and the transistor level modeling makes it hard to generalize the results. For

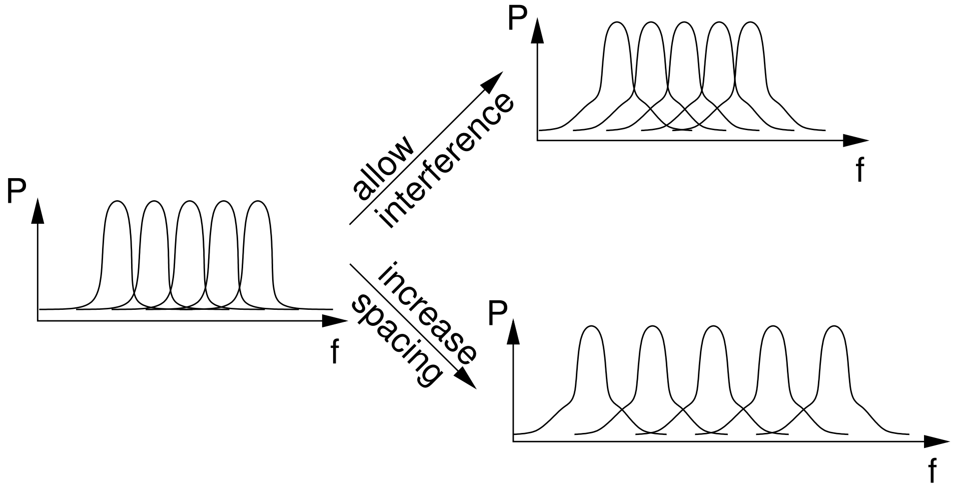


Figure 2.6: The effect of PA nonlinearity in multicarrier systems

these reasons, the power amplifiers are often modeled using higher level models with a limited number of parameters obtained by measurements.

The power amplifier models can be divided into two main categories: memoryless models and models with memory. The models with memory will be discussed in more detail in Section 2.8.

The memoryless (or often, actually, quasi-memoryless) models assume that the previous values of the signal to be amplified do not affect the current and future PA output signal values. The memoryless models separate the distortion into two components: amplitude-to-amplitude (AM-AM) distortion and amplitude-to-phase distortion (AM-PM). Also, phase-to-amplitude (PM-AM) and phase-to-phase (PM-PM) distortion is possible, but usually these components are negligible and ignored. However, a quadrature modulator may generate strong PM-PM and PM-AM distortion and should be taken into account (Chapter 8), especially in systems using predistortion.

AM-AM distortion depicts the compression or the expansion of the signal envelope as a function of the input signal amplitude and is caused by, for example, the output signal compression near the supply voltages and transistor cut-off region. The AM-PM distortion, on the other hand, describes the phase shift of the signal as a function of the input signal envelope. Actually, as the definition of the signal phase is dependent on the previous values of the signal, the AM-PM distortion is not strictly speaking memoryless [26]. However, the memory is very short and the AM-PM distortion can be approximated to be memoryless. For this reason, the PA models taking into account the AM-PM distortion are sometimes called quasi-memoryless [26].

The dependence of the memoryless PA models on only the amplitude of the input signal can be utilized in both modeling and linearization of the power amplifiers, by using functions dependent only on the amplitude of the signal.

As the distortion components around the harmonics of the carrier frequency can be fairly easily filtered away it is usually considered to be enough to model only the distortion components close to the carrier frequency. This kind of model is called a passband model [5]. The passband model uses real-valued signals and operates at carrier frequency. This may slow down the simulations significantly if there are also low frequency signals present, which is the case in several linearization methods. Therefore, the power amplifier is often modeled using a baseband model, which is a complex baseband frequency approximation of the nonlinearity [5]. This makes it possible to reduce the frequency range required for the simulation and simplifies calculations, although some of the accuracy is of course lost due to approximation. In this thesis, the power amplifier will be modeled in the simulations and calculations using baseband models. The following sections present some commonly used baseband PA models.

2.5.1 Polynomial PA model

Probably the most straightforward way to describe the distortion of a power amplifier is to use a polynomial function [27, 28]

$$v_{out} = \sum_{n=1}^{N} a_n v_{pd} \left|v_{pd}\right|^{n-1} \tag{2.5}$$

where v_{pd} is the PA input voltage and also the predistorter output voltage, v_{out} is the output voltage and a_n is the distortion coefficient; all the symbols can have complex values and the absolute values of the baseband and v_{out} signals are normalized to range from 0 to 1. The distortion coefficients are found by, for example, least squares fit of the amplitude and phase measurements of the PA. A low-order polynomial model is a fairly accurate model for linear (class A) amplifiers.

Often, only odd-order terms are used in modeling power amplifiers based on the assumption that the second-order distortion caused by the power amplifier does not generate distortion around the carrier frequency and thus has no effect on the baseband model.

However, Ding et al [29] suggest that the polynomial order can be reduced by including the even-order terms also. If we inspect (2.5) more closely, we note that, actually, the even-order terms are of the form

$$x\left|x\right|^{n-1} = x\left(xx^*\right)^{\frac{n}{2}-1}\sqrt{xx^*}. \tag{2.6}$$

These terms contain a square root function and thus have an infinitely wide spectrum. By using the power series expansion [30] of the square root function, and assuming that $xx^* < 1$, we can expand (2.6) to

$$\begin{aligned}
x\left|x\right|^{n} &= x\left(xx^{*}\right)^{\frac{n}{2}-1}\sum_{m=1}^{\infty}\left(\begin{array}{c}\frac{1}{2}\\ m\end{array}\right)\left(xx^{*}-1\right)^{m}\\
&= x\left(xx^{*}\right)^{\frac{n}{2}-1}\sum_{m=1}^{\infty}\left(\begin{array}{c}\frac{1}{2}\\ m\end{array}\right)\sum_{k=0}^{m}\left(\begin{array}{c}m\\ k\end{array}\right)\left(xx^{*}\right)^{m-k},\\
&= \sum_{m=1}^{\infty}p_{m}x\left(xx^{*}\right)^{m+\frac{n}{2}-1}\\
&= \sum_{m=1}^{\infty}p_{m}x\left|x\right|^{2m+n-2}
\end{aligned} \tag{2.7}$$

where p_m is a constant.

Thus we can see that the “even-order” terms in (2.5) actually generate infinite number of odd-order distortion terms. This conclusion can be used to ones advantage, as this actual means that the odd-order terms of (2.5) can be used to cancel the low-order distortion coefficients of the even-order terms of (2.5) and then the remaining higher order coefficients of (2.7) are utilized to generate the higher order terms required in modeling. It should, however, be noted that calculation of the odd order powers of the absolute value of complex number is quite complex operation requiring the square root operation.

The simplest polynomial distortion function is the third-order polynomial

$$v_{out} = v_{pd} + a_3 v_{pd}\left|v_{pd}\right|^2 = v_{pd} + a_3 v_{pd}^2 v_{pd}^*, \tag{2.8}$$

where a_3 is the complex third-order distortion coefficient. The second form of the function is achieved by transforming the absolute value into complex conjugate form. Due to its simplicity, the third-order polynomial is used in several analytic calculations in this thesis to model the PA.

The distortion coefficient, as the name says, defines the nonlinearity of the amplifier model. The simplicity of the third-order distortion function makes it feasible to use in theoretical calculations, although it does not take into account the nonlinearities at the low amplitudes near the cutoff region.

It may be difficult to see directly the effect of the distortion coefficient on the nonlinearity of the amplifier, as it also depends on the peak-to-average ratio of the signal and the signal amplitude. We can estimate the level of distortion caused by third order nonlinearity by using a two-tone input signal

$$v_{pd} = B\sin(\omega_{env}t)\cos(\omega_{rf}t) = \frac{B}{2}\sin(\omega_{rf}t + \omega_{env}t) - \frac{B}{2}\sin(\omega_{rf}t - \omega_{env}t) \tag{2.9}$$

where B is the signal amplitude, ω_{env} is the baseband frequency and ω_{rf} is the carrier angular frequency. When it is inserted into (2.8), distortion function becomes

$$\begin{aligned}v_{out} = (\frac{B}{2} + \frac{9a_3B^2B^*}{32})\sin(\omega_{rf}t + \omega_{env}t) - (\frac{B}{2} + \frac{9a_3B^2B^*}{32})\sin(\omega_{rf}t - \omega_{env}t)\\ -\frac{3a_3B^2B^*}{32}\sin(\omega_{rf}t + 3\omega_{env}t) + \frac{3a_3B^2B^*}{32}\sin(\omega_{rf}t - 3\omega_{env}t) + \frac{3a_3B^2B^*}{32}\sin(3\omega_{rf}t + \omega_{env}t)\\ -\frac{3a_3B^2B^*}{32}\sin(3\omega_{rf}t - \omega_{env}t) - \frac{a_3B^2B^*}{32}\sin(3\omega_{rf}t + 3\omega_{env}t) + \frac{a_3B^2B^*}{32}\sin(3\omega_{rf}t - 3\omega_{env}t)\end{aligned} \tag{2.10}$$

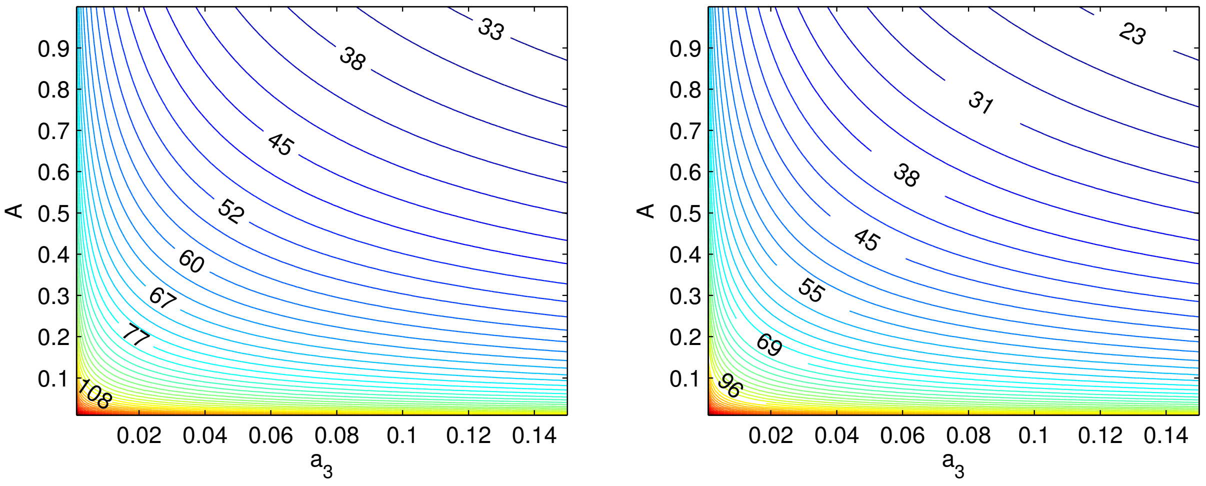
from which we can calculate the in-band and the out-of-band third order distortion on the adjacent channels in dBc. The in-band distortion level is given by the difference of the multiplier of $\sin(\omega_{rf}t + \omega_{env}t)$ and $\frac{B}{2}$ (the original amplitude of the corresponding signal component) and the adjacent channel distortion is given by the multiplier of $\sin(\omega_{rf}t + 3\omega_{env}t)$ thus

$$P_{inband} = 20\log(\frac{\frac{B}{2}}{\frac{9}{32}a_3B^2B^*})dBc = 20\log(\frac{16}{9a_3BB^*})dBc \tag{2.11}$$

and

$$P_{oob} = 20\log(\frac{\frac{B}{2}}{\frac{3}{32}a_3B^2B^*})dBc = 20\log(\frac{16}{3a_3BB^*})dBc \tag{2.12}$$

It can be seen that the distortion level depends on the square of the signal amplitude as well as on the distortion coefficient. The in-band and out-of-band distortion powers in dBc as the function of the distortion coefficient and the amplitude are plotted in Figure 2.7. The strong dependence on the amplitude can be clearly seen and- as was illustrated in Figure 2.3, the linearity of the amplifier increases and the efficiency decreases as the input power decreases.



(a) Out of band distortion

(b) In band distortion

Figure 2.7: The level of out of band and in band distortion of a two tone signal in dBc as a function of the distortion coefficient and signal amplitude

The signal type also affects the distortion level of the amplified signal. This is due to the fact that types of different signals have a different probabilities to be at certain amplitude level and thus they experience different level of distortion. One figure that describes this amplitude distribution is the peak-to-average ratio (PAR) or crest factor [31],

$$PAR = \frac{max(|v(t)|^2)}{\frac{1}{t}\int_t |v(t)|^2\, dt}, \tag{2.13}$$

which tells the value of the maximum amplitude compared to the average amplitude value. The higher the PAR, the more time the signal spends on low amplitudes, and the higher and sparser the peaks are. Thus if several signals with different PARs are normalized to have the same maximum amplitude, and are driven through a third-order nonlinearity, the signals with the highest PAR are least distorted. Although this may seem like a good quality, actually the high PAR means low average signal power and thus lower efficiency, which is not desirable. On the other hand, if the powers of the signals are normalized, the signals with the lower PAR have lower peak amplitude and experience less distortion.

The reason not to use signals with a low PAR is that we would like to use signals with high spectral efficiency to transfer as many bits as possible in as narrow a bandwidth as possible. Unfortunately, as the spectral efficiency increases, so does the PAR [32]. Table 2.1 [32] shows spectral efficiencies and PARs for several different modulation methods and, as can be seen, the more efficient modulation we have, the higher the PAR gets.

Table 2.1: The spectral efficiency and PAR of several modulation methods[32]

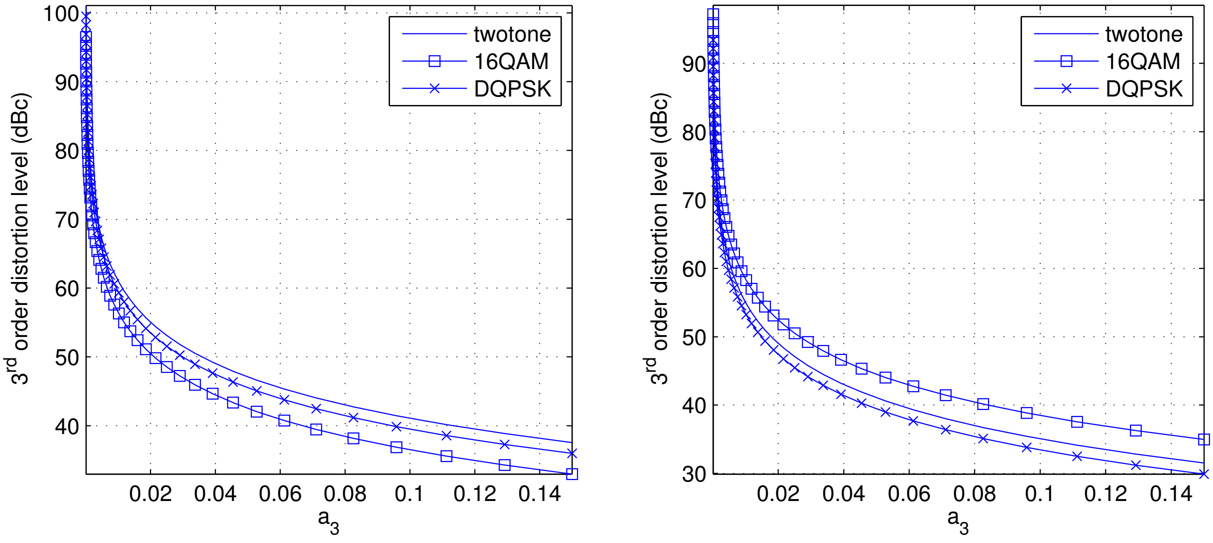
Method (RRC, roll off=0.5)	QPSK	16QAM	64QAM	256QAM
Spectral efficiency $\frac{bit}{s}/Hz$	1.3	2.7	4	5.3
PAR	3.1	5.2	5.9	6.2

Figure 2.8 shows the ACPs for two tone sine signal, DQPSK signal, and a 16QAM signal. As can be seen, when the amplitudes are normalized to one, the 16QAM signal gives the best ACP, but, when the powers are normalized, the 16QAM gives the worst ACP.

Thus there are conflicting requirements for high power efficiency resulting in a nonlinear amplifier and high spectral efficiency resulting in high PAR.

2.6 Advanced PA models

The modeling ability of the polynomial PA model is quite limited unless the order of the polynomial is high, which increases the computational complexity. Especially modeling the cut-off nonlinearity increases the required order of the polynomial significantly. It is therefore often beneficial to use a less general model optimized for power amplifiers that has a lower number of coefficients. Several models have been proposed that are suitable for different types of power amplifiers. These will be discussed in the following sections.



(a) Equal power

(b) Equal amplitude

Figure 2.8: Power of third-order distortion as a function of the third-order distortion coefficient with different signals

All of the models are expressed as the amplitude-dependent gain, $G_{PA}(|v_{pd}|)$, and phase shift, $\Phi_{PA}(|v_{pd}|)$, functions, by which the complex PA amplification function can be expressed as

$$A_{PA}(|v_{pd}|^2) = G_{PA}(|v_{pd}|)e^{2\pi\Phi_{PA}(|v_{pd}|)} \tag{2.14}$$

and PA output signal as

$$v_{out} = A_{PA}\left(|v_{pd}|\right) \cdot v_{pd}. \tag{2.15}$$

2.6.1 Saleh model

The Saleh model [33] is a commonly used power amplifier model, that is designed especially for traveling wave tube (TWT) amplifiers. The Saleh model is recommended as the standard PA model by the IEEE broadband wireless access group [34].

The Saleh model is

$$G_{PA}(|v_{pd}|) = \frac{a_A\,|v_{pd}|}{1 + b_A\,|v_{pd}|^2},\; \Phi_{PA}(|v_{pd}|) = \frac{a_\Phi\,|v_{pd}|^2}{1 + b_\Phi\,|v_{pd}|^2}, \tag{2.16}$$

where a_A, a_Φ, b_A and b_Φ are the distortion coefficients that are fitted to the measured data.

Often-used values [34] for the coefficients are $a_A = 2.1587$, $b_A = 1.1517$, $a_\Phi = 4.033$ and $b_\Phi = 9.104$, which were presented by Kaye et al [35]. The problem with this model is that it is optimized to TWT amplifiers so it is not as well suited for describing solid-state amplifiers [36].

2.6.2 Rapp model

The Rapp model [37] is a PA model designed for solid state power amplifiers [34]. Only the gain function of the model,

$$G_{PA}(|v_{pd}|) = \frac{1}{(1+(\frac{|v_{pd}|}{a_A})^{2b_A})^{\frac{1}{2b_A}}}, \tag{2.17}$$

has been presented and no general parameters have been suggested [34]. The model exhibits very linear behavior at the low amplitude values, which is often not the desired behavior.

2.6.3 Ghorbani-model

The Ghorbani model is another PA model designed for solid-state PAs. The gain and phase functions for this model are [38]:

$$G(|v_{pd}|) = \frac{a_A\left|v_{in}^{c_A}\right|}{1+b_A\left|v_{in}^{c_A}\right|} + d_A\left|v_{in}\right|, \; \Phi(|v_{pd}|) = \frac{a_\Phi\left|v_{in}^{c_\Phi}\right|}{1+b_\Phi\left|v_{in}^{c_\Phi}\right|} + d_\Phi\left|v_{in}\right| \tag{2.18}$$

a_A, b_A, c_A, d_A, a_Φ, b_Φ, c_Φ and d_Φ are the nonlinearity parameters. The standard values [36] for the parameters are $a_A = 8.1081$, $b_A = 1.5413$, $c_A = 6.5202$, $d_A = -0.0718$, $a_\Phi = 4.6645$, $b_\Phi = 2.0965$, $c_\Phi = 10.88$ and $d_\Phi = -0.003$. The Ghorbani model is very suitable for modeling FET amplifiers and can also model the low amplitude nonlinearity [36].

2.7 Models used in simulations

Three different power amplifier models were used in the simulations conducted in this thesis, one with high nonlinearity at low amplitudes (PA1), one with moderate nonlinearity at low and high amplitudes (PA2) and one with high nonlinearity at high amplitudes (PA3). These models were selected to inspect different aspects of the power amplifier nonlinearity. The nonlinearities were modeled using a modified Ghorbani model

$$G(|v_{pd}|) = \frac{a_A\left|v_{in}^{b_A}\right|}{1+c_A\left|v_{in}^{e_A}\right|} + d_A\left|v_{in}\right|, \; \Phi(|v_{pd}|) = \frac{a_\Phi\left|v_{in}^{b_\Phi}\right|}{1+c_\Phi\left|v_{in}^{e_\Phi}\right|} + d_\Phi\left|v_{in}\right|. \tag{2.19}$$

The difference to the original model is that the exponent of the denominator is made different from the exponent of the numerator, which makes the function more flexible.

2.7.1 Model parameters

The following sections present the parameters used in generating the PA models. The AM-AM and AM-PM functions of the models are plotted in Figure 2.9.

2.7.1.1 PA1

PA1 has strong nonlinearity at low amplitudes but is linear at high amplitudes, so it represents an amplifier that is driven near the cut-off of the transistors. For this model $a_A = 1.92$, $b_A = 1.74$,

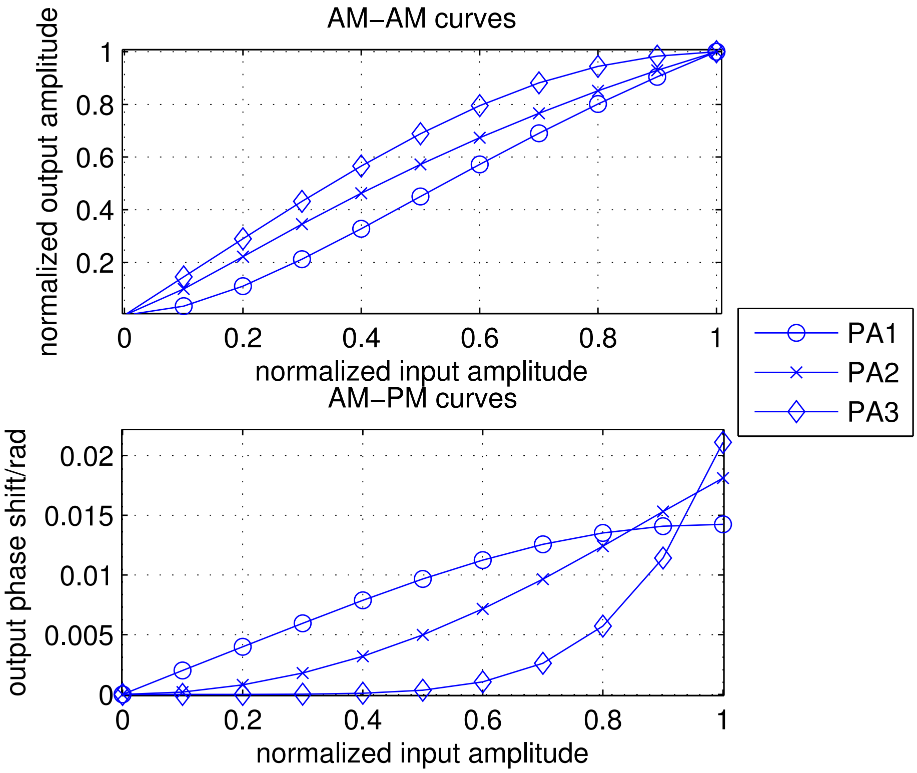


Figure 2.9: AM-AM and AM-PM curves of the amplifiers used in simulations

$c_A = 0.92$, $d_A = 0$, $e_A = 1.74$, $a_\Phi = 0.02$, $b_\Phi = 1$, $c_\Phi = 0.4$, $d_\Phi = 0$ and e_Φ =3.5

2.7.1.2 PA2

PA2 has nonlinearity at both low and high amplitudes, and represents a class-AB, -B or -C amplifier. For this model, $a_A = 1.62$, $b_A = 1.24$, $c_A = 0.82$, $d_A = -0.009$, $e_A = 1.24$. The phase distortion function was implemented by subtracting two phase functions of form 2.20 from each other so that $\Phi(|v_{pd}|) = \Phi_1(|v_{pd}|) - \Phi_2(|v_{pd}|)$. This was done to make it possible to implement a phase distortion function with rapid changes at high and low amplitudes and fairly constant value at the middle values. The coefficients for the phase distortion function are $a_{\Phi 1} = 0.33 \cdot 10^{-2}$, $b_{\Phi 1} = 0$, $c_{\Phi 1} = 0.36 \cdot 10^{-2}$, $d_{\Phi 1} = 0$, $e_{\Phi 1}$ =1, $a_{\Phi 2} = -0.76$, $b_{\Phi 2} = 0$, $c_{\Phi 2} = 6.7$, $d_{\Phi 2} = 0$ and $e_{\Phi 2} = 1$.

$$G(|v_{pd}|) = \frac{a_A \left| v_{in}^{b_A} \right|}{1 + c_A \left| v_{in}^{e_A} \right|} + d_A \left| v_{in} \right|, \; \Phi(|v_{pd}|) = \frac{a_\Phi \left| v_{in}^{b_\Phi} \right|}{1 + c_\Phi \left| v_{in}^{e_\Phi} \right|} + d_\Phi \left| v_{in} \right|. \tag{2.20}$$

2.7.1.3 PA3

PA3 has nonlinearity at high amplitude and is linear at low amplitudes. This model represents a class A amplifier. The parameters used in this model are $a_A = 1.92$, $b_A = 1$, $c_A = 0.46$, $d_A = 0$, $e_A = 3$, $a_\Phi = 0.023$, $b_\Phi = 6$, $c_\Phi = 0.1$, $d_\Phi = 0$ and e_Φ =2

2.8 Memory effects of power amplifiers

In this thesis, the power amplifier is considered to be memoryless. However, in wideband and high-power systems and systems requiring high linearity, the memory should be taken into account.

The memory effects are basically frequency-domain fluctuations in the transfer function of the power amplifier or time dependence of the transfer function. The effect of memory on the PA output can be described with a frequency-domain plot of the PA output shown in Figure 2.10 [39, 40]. IM_L and IM_U are the intermodulation results. The height of the intermodulation results represents the power of the components and the angles ϕ_L and ϕ_U represent the phase shift of the intermodulation results. The power of the intermodulation results and the phases are not exactly the same and also vary depending on the frequency separation of the two-tone signals. This generates problems with memoryless linearization systems, since it tries to compensate upper and lower intermodulation results similarly and thus at least one of them is inferiorly compensated. This shows as significantly different upper and lower intermodulation distortions. The problems and predistortion systems that are designed alleviate the problems are discussed in section 3.6.

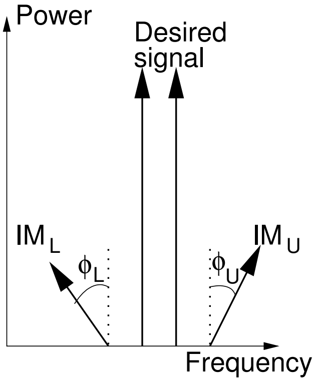


Figure 2.10: The effect of memory on distortion of a two-tone signal [39, 40]

The dependence of the intermodulation distortion components on the separation of the two-tone signals can be used to discover the memory effects in a PA [41] by testing the PA with two-tone signals with different spacings on the wanted signal band and noting the behavior of the intermodulation results. However, modeling a PA with memory requires more complicated measurements, which are described in, for example, [41, 42].

2.8.1 Sources of memory in PAs

As can be expected from the fact that the memory manifests itself as the phase and amplitude fluctuation of the intermodulation results as a function of the frequency, one of the sources for the memory effects are the capacitances and inductances in the amplifier chain or the frequency dependent impedances in the PA chain [39]. These are called the electrical memory effects. One source of these variable impedances are the bias networks of the transistors that can not be made infinitely wideband and at some point their impedance starts to change with the frequency [39]. By proper design of the bias networks the limit frequency at where the impedance starts to fluctuate can be pushed at higher frequencies [39] and thus the fluctuation affects the wide band transmitters the most [43].

Another source of the memory effects are the thermal fluctuations of the power amplifier due to the signal level [39, 43]. The dissipated power in the power amplifier changes with the signal level and, due to this, the temperature of the transistors and other components fluctuate, changing their electrical characteristics, such as the generated distortion [39]. However, the heat sinks and the packaging of the device do not heat up instantaneously; thus the past changes caused by the increased power dissipation affect the upcoming signal values also[39]. Due to the slowness of the heating process, the thermal feedback is of lowpass type and its effects show on the bandwidths up to 1 MHz [39, 43].

In conclusion, the electrical memory effects affect systems using wide-band signals (bandwidth >5MHz) and the thermal memory effects affect systems using narrow band signals (bandwidth <1MHz), in the middle range (1MHz<bandwidth<5MHz), the memory effects are quite small [43]. However, it should be noted that the wideband signals also include low-frequency components that are affected by the thermal memory.

2.8.2 PA models with memory

To include the effect of memory on the PA model, it must be designed to also have a time-dependent component. There are several methods that can be used to model the PA with memory.

2.8.2.1 Volterra series

The Volterra series is a multivariate polynomial series of the current and previous signal values [4, 44–46]. It is expressed using a discrete time step and the previous signal values used in the calculation deviate from the current value by integer multiples of this step. The series is [45]

$$v_{PA}(t) = \sum_{k=1}^{K} \sum_{m_1=0}^{M-1} \cdots \sum_{m_k=0}^{M-1} h_k(m_1, \ldots, m_k) \prod_{l=1}^{k} v_{pdRF}(t - m_l t_s), \tag{2.21}$$

where m_k are the delays in discrete time, $h_k(m_1, \ldots, m_k)$ are the coefficients for the terms, v_{pdRF} is the PA input RF signal, t_s is the time step, M is the number of delays and K is the order of the polynomial. By increasing M and K and reducing t_s, the accuracy of the model can be improved, but at the same time its complexity increases. However, with high enough order, the Volterra series is the most versatile modeling method.

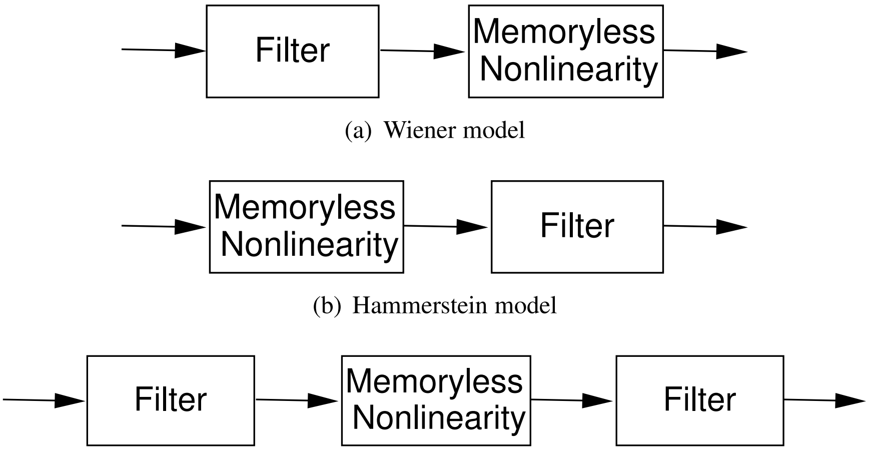
The coefficients can be found using, for example, least squares fitting or some recursive method such as RLS. (2.21) can be written in baseband form by replacing v_{pdRF} with its baseband equivalent,

$$v_{pd} = \Re\left\{e^{j\omega_0 t}\tilde{x}(t)\right\}, \tag{2.22}$$

where ω_0 is the center frequency.

2.8.2.2 Wiener, Hammerstein and Wiener-Hammerstein models

Wiener, Hammerstein and Wiener-Hammerstein models use simplifying approximations to reduce the complexity of the memory model. All of these models are based on separating the memory from the memoryless portion of the nonlinearity. This is done by dividing the model into a filter part and a memoryless PA model (Sections 2.5 and 2.6). The Wiener model assumes a memoryless nonlinearity preceded by a filter, the Hammerstein model assumes a filter preceded by a memoryless nonlinearity and the Wiener-Hammerstein model assumes a memoryless nonlinearity between two filters [45]. The system block diagrams are shown in Figure 2.11 [45].



(a) Wiener model

(b) Hammerstein model

(c) Wiener-Hammerstein model

Figure 2.11: PA models with memory based on filtering

The systems simplify the model significantly, as, instead of requiring several cross product terms even for a mildly nonlinear amplifier with short memory, these models require only the filter parameters and the memoryless model parameters.

However, these models have some important limitations. The Wiener model and the Wiener-Hammerstein model cause the filter parameters seen in the output to be nonlinear, which makes the system identification more difficult. Also, the decoupling of the memory and the nonlinearity does not correspond closely to the real situation and does not take into account the change of the filtering effects with different power levels [45]. However, the models have been fairly widely used [47–51].

2.8.2.3 Memory polynomial

The memory polynomial simplifies the Volterra series by exploiting the fact that the nonlinearities in the PA are almost completely phase independent. Thus the baseband Volterra series can be simplified to contain only powers of $|v_{pd}|$ still retaining more of the accuracy of the Volterra series than the Wiener and Hammerstein models [45]. The model can be written as [45, 52]

$$v_{out} = \sum_{k=0}^{K-1} \sum_{l=0}^{M-1} a_{kl} v_{pd}(t - m_l t_s) \left| v_{pd}(t - m_l t_s) \right|^k . \tag{2.23}$$

The memory polynomial can also be seen as a number of parallel wiener filters, thus it is an extension of these and takes into account of the memory characteristics changing with the signal level. The memory polynomial can be described using the block diagram shown in Figure 2.12 [53].

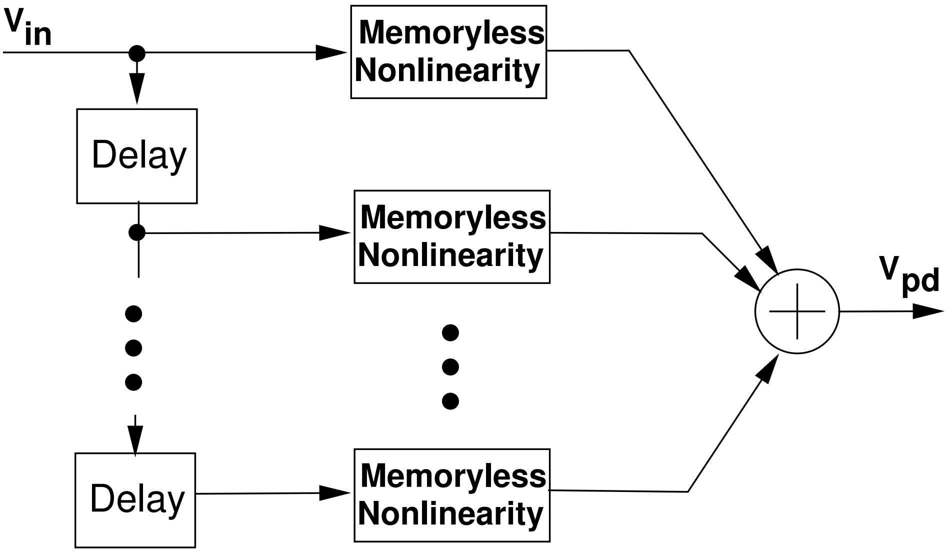


Figure 2.12: Block diagram of the hardware implementation of a memory polynomial

A memory polynomial has been used to model several PA and predistortion systems with memory [45, 53–59].

2.8.2.4 Other models

There are also other less common models for power amplifiers with memory such as memory polynomial combined with Wiener filtering [60], generalized memory polynomial [45], memory polynomial with memoryless nonlinearity [61] and neural networks [62]. All of these methods are fairly recent and have not seen wide use.

2.9 Conclusions

This chapter discussed the effect of the nonlinear PA on modern communication systems using variable amplitude signal modulation methods. It was seen how the nonlinearity distorts the data signal and also interferes with other signals in nearby channels. These effects can be reduced by the design of the power amplifier, which, however, results in more power hungry design, as well as by selection of the modulation method and signal separation in frequency-domain, which results in spectrally less efficient designs.

This chapter also presented several commonly used simulation models for memoryless power amplifiers including those used in the simulations presented in this thesis. The effect of memory on the power amplifier nonlinearity as well as on some PA models designed to take into account the memory effects were discussed.

The following chapters will discuss the reduction of the adverse effects of nonlinearity by linearization of the power amplifier.

Chapter 3

Linearization of a power amplifier

3.1 Introduction

It would be beneficial to use some signal processing method to compensate or reduce the distortion caused by the PA or even make it irrelevant. This would allow the use of very efficient amplifiers without exceeding the spectral efficiency and error rates required by the application.

This chapter will give an introduction to this kind of linearization methods. First, the linearization of power amplifiers in general and some of the most common linearization methods in particular will be discussed. After this, the chapter will concentrate on the predistortion linearization. First, different predistortion methods are discussed, mostly from analog point of view, and then the implementations of the predistortion function. Following this the chapter will briefly discuss the digital predistortion methods (on which the rest of this thesis will concentrate).

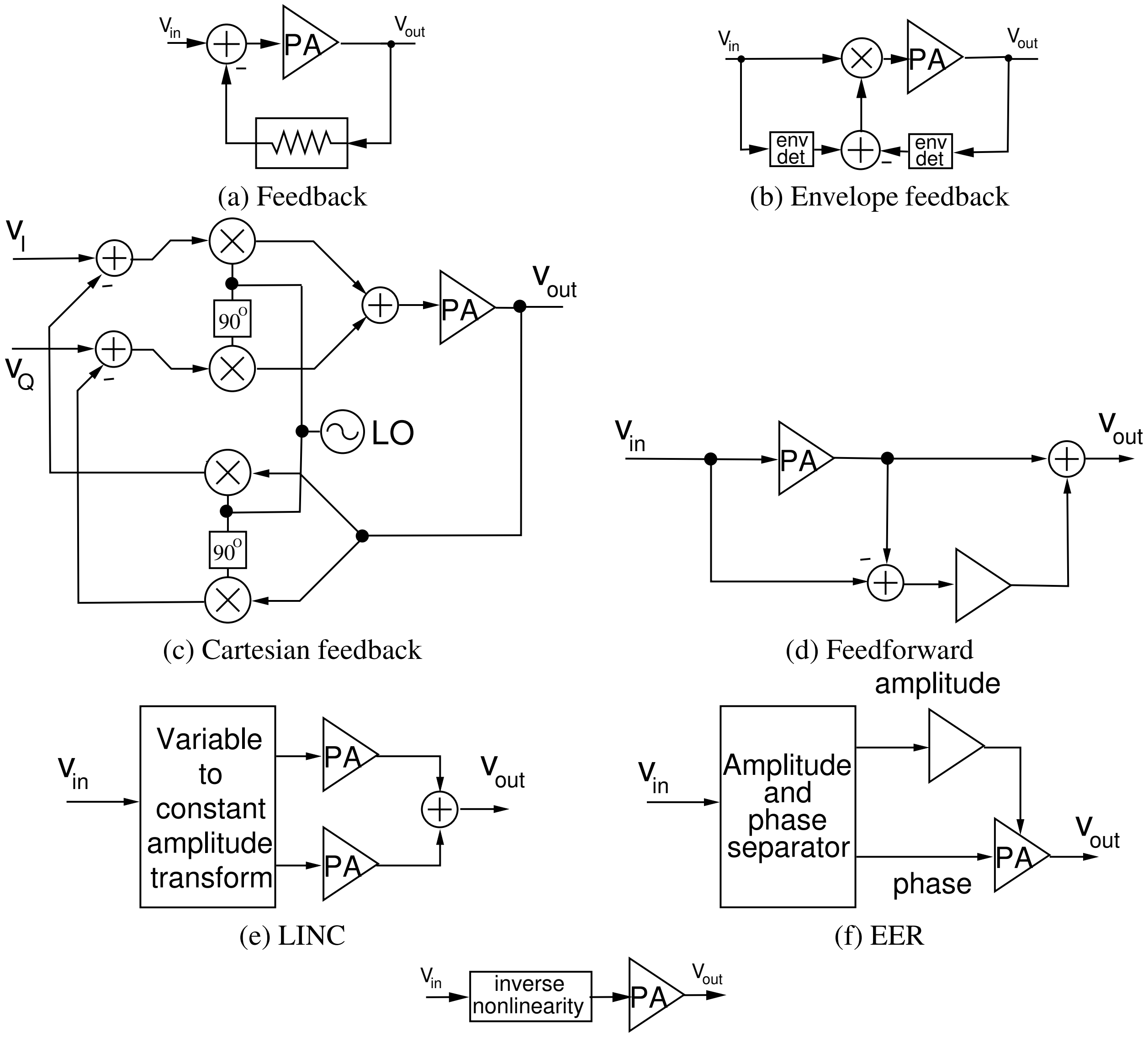
According to previous chapter, all power amplifiers have memory effects and in some applications these significantly affect their operation. In these applications the predistorters have to be able to compensate also the memory effects. Although this thesis concentrates on memoryless applications a brief overview of predistortion methods with memory will be given.

Finally, a survey and comparison of published predistorters (including the predistorters implemented during this thesis work) will be presented.

3.2 Linearization methods

Many of the PA linearization methods are originally fully analog methods invented several decades ago. However, due to increased use of wideband variable amplitude modulation methods and sufficiently advanced analog and digital components, the interest towards those has increased significantly in the recent years [3].

Figure 3.1 illustrates several commonly used PA linearization methods [3]. The first methods suggested for PA linearization were feedforward [63] and feedback [64] systems. The feedback systems employ, at simplest, a linear negative feedback of the PA output to the input [3]. A basic feedback linearization system is shown in Figure 3.1a. This method is quite suitable for low frequency applications, but, at higher frequencies, the method has significant stability problems [3]. To alleviate stability problems, the method has been modified to use signals at baseband frequencies in the linearization feedback.



(a) Feedback

(b) Envelope feedback

(c) Cartesian feedback

(d) Feedforward

(e) LINC

(f) EER

(g) Predistortion

Figure 3.1: Common PA linearization methods [3]

Envelope feedback (Figure 3.1b) was presented for electron tube [65] amplifiers in [66] and for solid state amplifiers in [67]. It uses the difference between the PA input and output envelopes to adjust the RF signal to compensate for the nonlinearities. With the help of envelope feedback linearization over 10 dB improvements in the carrier to interference ratios have been achieved [68].

Another feedback linearization method based on the baseband feedback is the Cartesian feedback [3, 69] (Figure 3.1c), which uses the difference of the baseband quadrature signals to compensate the nonlinearity. Thus the system requires both a quadrature modulator and a demodulator. The Cartesian feedback has been able to achieve over 30dB improvements in ACP [3].

Although there are also other improvements to the basic feedback linearization, the envelope feedback and Cartesian feedback have a special importance to the digital predistortion systems discussed in this thesis. The RF- and baseband-predistortion systems discussed in the following chapters have the same layout of the analog paths as these two feedback systems, but the feedback path from the PA output to the PA input replaced by the digital adaptation algorithm.

The feedforward linearization systems [63] (Figure 3.1d) calculate the difference between the PA input and output like the feedback systems, but, instead of feeding the difference signal to the PA input, they subtract the difference from the PA output to compensate the distortion. The feedforward was not used much for several decades as its matching and linearity requirements for the feed forward path made it more complex than the feedback and offered no significant advantage for narrow band signals [3]. However, due to the improvement in electronic components and the increase in use of wideband signals in telecommunications, the feedforward has seen more use due to its unconditional stability, wide bandwidth and its retaining the original amplification of the PA [3, 70]. The disadvantages of the feedforward systems are the complexity and the matching and linearity of the error amplifier and the fact that the system cannot adapt to changes in the PA without additional control functions [3]. Feedforward systems have been able to achieve 15dB improvement in ACP [71].

The Linear amplification using Nonlinear Components (LINC) (Figure 3.1e) and Envelope Elimination and Restoration (EER) (Figure 3.1f) are PA linearization methods that use heavy signal processing to transform the variable amplitude signal into constant amplitude signals for the nonlinear amplification [3]. These methods often perform the transformation of the signal on the baseband and include also the up conversion of the signals, thus being actually linearization methods for the whole transmitter [3].

LINC splits the variable amplitude signal into two constant amplitude phase modulated signals that are amplified separately with nonlinear high efficiency amplifiers and then combined to regenerate the variable amplitude signal [72]. Although the system is basically simple, the generation of the constant amplitude signals is quite a difficult task using analog components. This has reduced the usability of the LINC system [3]. However, as the digital circuitry is nowadays fast enough to be able to generate the desired signals in the digital domain, the LINC system has become more feasible [3]. The LINC system has been able to achieve 20 dB ACP improvements [73].

The EER systems (Figure 3.1g) avoid the nonlinearity of efficient amplifiers by separating the amplitude and the phase of the signal into two signals [74]. The phase signal is amplified with a nonlinear RF amplifier and the amplitude signal with linear audio amplifier. The amplitude signal is used to modulate the power source of the RF amplifier to regenerate the amplitude modulation [74]. As is the case with LINC, the analog signal separation generates problems, such as nonlinear envelope detection and delay differences between the signals [3]. Generation of the signals in digital domain can alleviate these problems. These kinds of digital implementations of

EER are commonly called polar transmitters [75]. EER has been able to achieve 8-10 dB ACP improvements [75, 76].

Finally Figure 3.1h illustrates the predistortion linearization system. This system generates an inverse transfer function for the PA before the amplification, thus generating linear amplification [3]. The predistorter has low hardware complexity and can be implemented to be unconditionally stable. The predistorter can also be adaptive so that the changes in the PA nonlinearity can be compensated. For these reasons, the predistorter was chosen as the subject of study in this thesis. The following sections will discuss the operation of the predistorter more thoroughly.

3.3 Operation of the predistorter

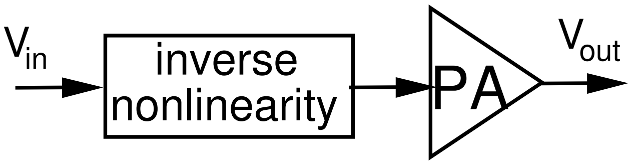


Figure 3.2: Basic block diagram of a predistorter

The predistorter linearizes the PA by generating a nonlinear transfer function that is inverse to the PA in such way, that when it precedes the PA, the overall amplification of the system is nearly linear (Figure 3.3a) [3]. By using equation 2.14, this can be written in the form of the following equation

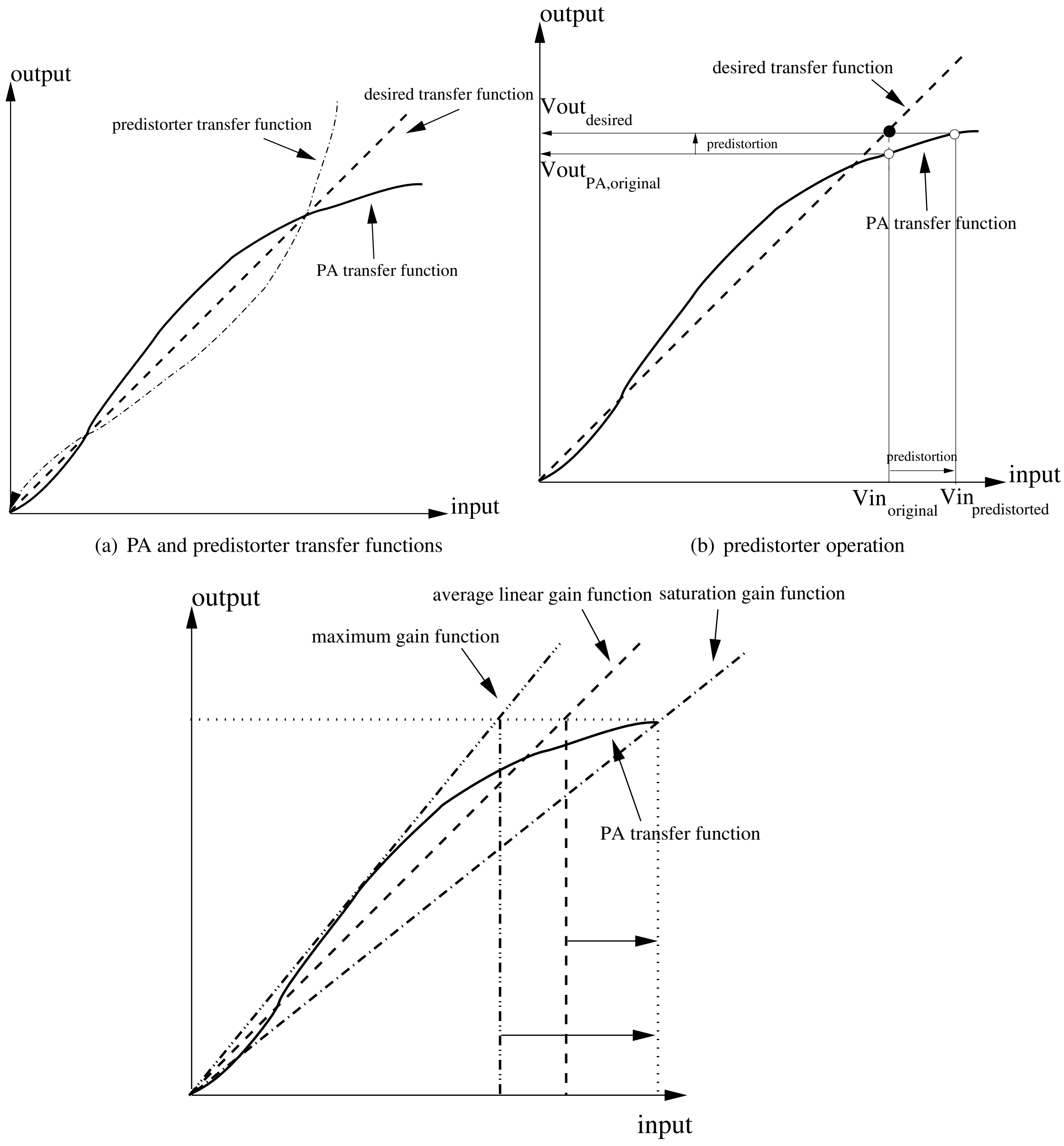
$$A_{PD}\left(v_{in}\right) \cdot A_{PA}\left(v_{pd}\right) = K, \tag{3.1}$$

where $\mathrm{A_{PD}}(\cdot)$ and $\mathrm{A_{PA}}(\cdot)$ are the predistorter and PA transfer functions respectively and v_{IN} is the input signal to the system. K is a complex constant. As the phase-dependent amplitude distortion and phase-dependent phase distortion are negligible, v_{IN} and v_{PD} can be replaced with $|v_{IN}|$ and $|v_{PD}|$. The predistorter output voltage, v_{PD}, is defined as

$$v_{pd} = A_{pd}\left(|v_{in}|\right) \cdot v_{in} \tag{3.2}$$

Predistortion can also be seen as an operation that shifts the PA input signal values in such a way that the PA output signal values correspond to those expected from a linear amplifier [77]. This point of view is illustrated in Figure 3.3b.

The desired linear amplification can be chosen in many different ways. Some possibilities are shown in Figure 3.3c [78]. The figure presents the linear system gain selection by the maximum amplification of the PA, average gain of the PA and the gain at the saturation point of the amplifier. In an optimal case the linear gain is chosen in such a way that the PA input signal amplitude spans the maximum linearizable amplitude range, as this offers the best efficiency. In the optimal case, the maximum output power should be the saturation power. However, it may not be beneficial to try to accommodate the gain required for the saturation power into the predistortion system as



(a) PA and predistorter transfer functions

(b) predistorter operation

(c) predistorter normalization

Figure 3.3: Operating principle of a predistorter. [3, 77, 78]

the gain required from the predistorter increases rapidly when the PA gets closer to saturation. In [78], it is suggested that the gain is adjusted for different input signal levels for maximum efficiency.

Other factors that affect the selection of the constant gain are the input and output power and gain specifications for the system. If the input signal level is already fixed to saturation before predistortion, then the gain should be selected by the saturation gain of the amplifier. This ensures that the whole amplitude range is used. This means that the gain with the predistorter is smaller than the nonlinearised gain. If the gain required of the amplifier is fixed, then the linear gain should be set to this value. Often, if there are no constraints for the amplification of input and output power levels, the system gain is set to the maximum gain of the PA [78]. This, however, may not be beneficial, especially in the case of digital predistorters, as the signal amplitude range has to accommodate the predistorted signal, which reduces the amplitude range reserved for the original signal, thus increasing quantization noise.

3.3.1 Adaptive predistortion function

Although the predistorter operates independently of the PA output, thus allowing unconditionally stable operation, often, especially in high-linearity systems, a feedback from the PA output to the update of the predistortion function is implemented. This is done in the light of the fact that the transfer function of the amplifier changes with the operation temperature, and due to aging, impacts etc. Therefore, even an accurately matched predistortion function becomes evidently only approximate and the linearity deteriorates [3]. This is not acceptable in high-linearity systems.

The feedback is usually implemented in such way that there is no direct connection between the PA output and the predistorter output, but, instead, the function is updated slowly independent of the predistorter output. This reduces the risk of instability. Possible methods for determining the required update are, for example, comparison of the time-domain input and output signals [11, 79], adjacent channel power measurements [28] and temperature measurements [80]. Often the adaptation is implemented using digital circuitry, but also analog adaptive algorithms have been published [81]. The adaptation methods are discussed in more detail in Sections 3.5.2 and 4.4.2.

3.3.2 RF, baseband and data predistortion

The predistortion systems can be divided into three main categories, namely RF/IF, baseband and data predistortion, according to the placement of the predistortion function. The RF/IF-predistortion introduces the predistortion function to the up-mixed RF or IF signal. Both the RF and IF predistortion systems operate similarly, the main difference being the placement of the system before or after the final up-conversion stage of the transmitter [3]; the placement is dictated by the system specifications and available components. Earlier, the RF/IF predistortion was a more common choice due to the fact that it can be implemented with simple analog circuitry, although the linearity improvement is limited [8, 82–84].

At simplest, the RF/IF predistorters can use simple RF diodes or transistors as the predistortion elements [3], but more complex systems use phase and amplitude modulator circuits [9, 10, 81, 85] or quadrature modulators [28, 86]. The control signals of the RF predistorter can be implemented to be adaptive. These kinds of non-static controls are usually implemented with the help of digital signal processing (DSP) [8–10, 28, 86], although analog implementations have been presented also [81].

Advantages of the RF/IF predistortion are the simplicity of the analog circuitry, independence of the baseband and PA implementations, which enables development of stand-alone predistortion chips and the possibility of implementation without any up- or down-conversion operations. The disadvantages include the restrictions to the shape of the predistortion function imposed by the analog circuitry and delay problems caused by large frequency differences between the signals (especially in non-static predistorters). The RF-predistortion will be discussed more thoroughly in Chapters 4-7.

Data predistorters are simple digital predistortion systems that try to adjust the transmitted baseband data symbols so that their distortion after the amplifier is minimized [3, 87–89]. The correction is made before up sampling and filtering of the data; thus the data predistorter is not able to compensate for the adjacent channel distortion. The advantage of the data predistorter is that it requires only a low clock rate DSP and a very simple LUT for predistortion function generation. Disadvantages are the inability to reduce ACP and the dependence of the predistortion function on the modulation scheme [3], thus their applicability is limited. The data predistorters are not investigated further in this thesis due to their operation in the non upsampled and filtered original data signals, thus being very much related to the modulation and coding which are not within the scope of this thesis.

A baseband predistorter performs the linearization of the PA by altering the signal at the baseband before any up-mixing operations; in modern quadrature transmitters, this means that the predistortion is done to two baseband signals [3]. The accuracy and matching of the predistortion functions required for acceptable linearity is hard to achieve using analog components and thus analog implementations have been rare [3].

The digital baseband predistortion [11, 79, 90, 91] is a refinement of the data predistortion principle, but, instead of altering the data symbols, the predistortion is moved closer to the D/A conversion in the digital signal path, after the filters. Thus, the baseband predistortion is able to also correct distortion in the actual signal envelope, assuming that the clock rates are high enough to accommodate the required spectral spreading. Due to being implemented fully in the digital domain, the baseband predistortion offers great flexibility for the predistortion function and therefore there are several different predistortion functions commonly used in baseband predistorters. The memory predistorters are also usually based on the baseband predistortion systems [53, 54].

The advantages of the baseband predistorters are the flexibility and accuracy of the digital predistortion functions. However, the baseband predistorters suffer from the fact that they include the quadrature modulators and up-conversion functions on the signal path. This means that they have to also deal with the nonidealities of the mixers and other analog components in addition to the PA.

The baseband predistortion will be discussed more thoroughly in Chapter 8.

3.3.3 Memoryless phase and amplitude predistortion implementations

As was discussed in Chapter 2, the main distortion sources in memoryless power amplifiers are the AM-AM and AM-PM distortions. Therefore, the predistorters are usually based on reducing amplitude-dependent distortion. A notable exception is the digital mapping predistortion which also is able to compensate for phase-dependent distortion.

A basic predistorter can be implemented with a very small number of components. This is enabled by the fact that the most dominating distortion mechanism in RF power amplifiers is the AM-AM distortion; thus the rudimentary linearity improvement can be gained by generating a simple approximation of the inverse amplitude distortion function of the PA. In fully analog RF predistorters a third order amplitude distortion function is often used [3]; this can be simply implemented with a properly biased nonlinear diode or single transistor. However, this method, although it has been quite widely used [3, 85, 92–97], is not very accurate and thus has only a limited correction ability [3]. More complex AM-AM predistortion methods have also been presented[3, 98].

When more linearity is required, the AM-PM distortion also has to be taken into account and thus efficient correction methods use separate phase and amplitude distortion circuits [85, 99–101]. The phase and amplitude correction circuits can be implemented with the help of the previously mentioned diode and transistor circuits combined with circulators, hybrids, power combiners, capacitive elements etc. However, the restrictions of the analog components on the available predistortion functions still limit the correction ability [3].

The problem can somewhat be alleviated by using a curve fitting approximation of the nonlinearity [3]. However, the analog implementation of a curve-fit predistorter is quite cumbersome and requires special circuits to implement it so that the signal to be predistorted can have negative amplitude values [3]. Another method to implement the predistortion function more flexibly is to use an analog phase and amplitude modulator [81] or quadrature modulator as the predistortion element.

To make the implementation of the predistorter function more flexible, the control of these vector modulators can be implemented digitally. This requires a DSP circuit that implements a piecewise constant or polynomial control of the analog predistortion element. This method is used in RF-predistorters (Chapter 4). The digitalization can be accomplished even more extensively by implementing the whole predistortion operation in digital domain, which is the method used in the baseband predistorters (Chapter 8).

The following sections will discuss the different predistortion functions in more detail.

3.4 Implementation of the predistortion function

There are several ways to implement the actual distortion functions. Usually, the inverse transfer function can not be directly implemented, but it has to be approximated with some other function instead.

One simple method is to use a polynomial function to approximate the nonlinearity. The polynomial function can be implemented using analog components [85, 92–97] or digitally [102, 103].

Another commonly used method is to generate a piecewise constant approximation of the required nonlinearity [3]. Using this method, it is possible to generate functions that would require very high order polynomials. However, as the solution is basically discrete and only approximates the function it increases the overall noise floor of the predistortion. Another problem with the linear approximation is that it often requires more parameters than the polynomial one. The digital predistortion systems are commonly based on this kind of predistortion function due to their fairly easy implementation using look-up tables (LUT) [9, 11, 79, 86, 91, 104, 105]. Despite their complexity, analog implementations have been presented as well [106].

The predistortion function can be applied to the PA input signal using vector modulators, complex multipliers, diodes etc.

3.4.1 Polynomial predistortion

As is the case with PA models, a polynomial predistortion function is probably the simplest method to approximate the function required to compensate the PA nonlinearity. It can be described with a formula similar to the polynomial distortion function of a power amplifier

$$v_{pd} = \sum_{n=1}^{N} a_{PDn} v_{in} \left| v_{in} \right|^{n-1}, \tag{3.3}$$

where a_{PDn} are the predistortion coefficients and N is the polynomial order. This method would seem to be quite suitable as usually the power amplifier nonlinearity can be approximated with a low-order polynomial. However, if this formula is substituted in (2.8) and the resulting function is solved for $v_{out} = v_{in}$, it is found that the equation has a solution only when $N = \infty$ [24]. If only a limited number of low-order coefficients are to be compensated, a lower order polynomial can be used. The drawback of this is that finite order compensation generates new distortion components that have a higher order than the original distortion and predistortion polynomials.

If we set as our goal to compensate the third order distortion coefficient and suppress any new fifth order distortion coefficients, we can use a fifth order predistortion polynomial (3.3). When this is inserted to (2.8) and solved for third and fifth order distortion equal to zero, we get the following predistortion coefficients:

$$a_{PD3} = -a_{PA3} \tag{3.4}$$

and

$$a_{PD5} = a_{PA3}(2a_{PA3} + \overline{a_{PA3}}), \tag{3.5}$$

where $\overline{a_{PA}}$ is the complex conjugate of the PA distortion coefficient. The predistortion generates seventh and higher order distortion components and they limit the maximum linearity [24]. The fifth order polynomial predistorter will be used throughout the thesis as a basic predistorter model to simplify the mathematical calculations.

Since the PA distortion functions are usually not invertible when using a finite bandwidth, the problem of generating new distortion components affects all predistortion systems. The higher order the distortion components the predistorter is able to compensate, the lower the residual distortion level. If the PA is fairly linear, a low-order polynomial is often enough for good enough linearity. The possibility of using a low-order polynomial means a lower number of parameters and eases the calculation of the function and the update of the function and makes the adaptive update converge faster. However, if the PA is nonlinear and especially if the nonlinearities are strong at the low signal amplitudes, the required order for the polynomial as well as the computational complexity become high.

3.4.1.1 Linearization ability of a polynomial predistorter

For one to be able to linearize the power amplifier with a predistorter, the distortion function of the PA must be such that the output signal is able to achieve the maximum of the input signal v_{in} with some value of predistorter output signal, v_{pd}.

The maximum value of third-order nonlinearity coefficient, a_3, in (2.8) that fulfills this requirement can be estimated by assuming the signals to be real and also the distortion coefficient to be real. If we set v_{out} to the maximum of the input signal, $v_{out} = 1$ and v_{pd} to be the maximum of the input signal multiplied by a real coefficient, $v_{pd} = B$, we can solve B as a function of a_3 from (2.8).

$$1 = B - a_3 B^3 \tag{3.6}$$

By using the general solution for the roots of a quadratic equation [30], we get three solutions for (3.6). Now, if we find the values of a_3 for which the solutions are real valued, in other words, realizable, we find that the maximum distortion coefficient is 0.148. Although the calculations were performed for real signals and coefficients, we can still use as a rule of thumb the requirement that for the amplifier to be predistortable the absolute value of a_3 has to be less than 0.15. Also, if we plot the required values of v_{pd} (Figure 3.4), we see that the required v_{pd} increases rapidly and the requirements for the predistorter get more and more demanding as the PA approaches saturation.

As the PA approaches saturation, the order of the polynomial required for accurate linearization increases as well. Figure 3.5 shows the residuals of the different order polynomials used to approximate the inverse transfer function of a third order polynomial with $a_3 = 0.098$ and $a_3 = 0.148$. As can be seen in the former case, the third order polynomial is able to achieve better fit than a ninth order polynomial in the latter case.

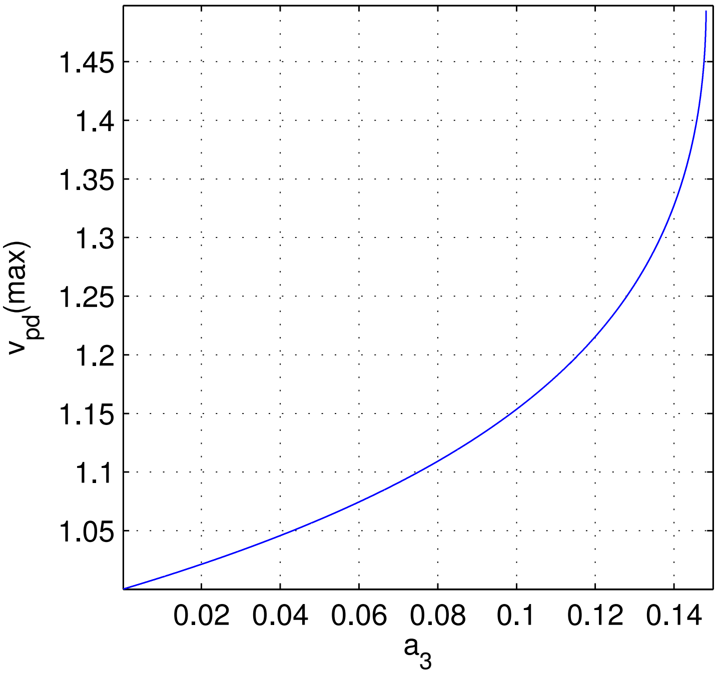
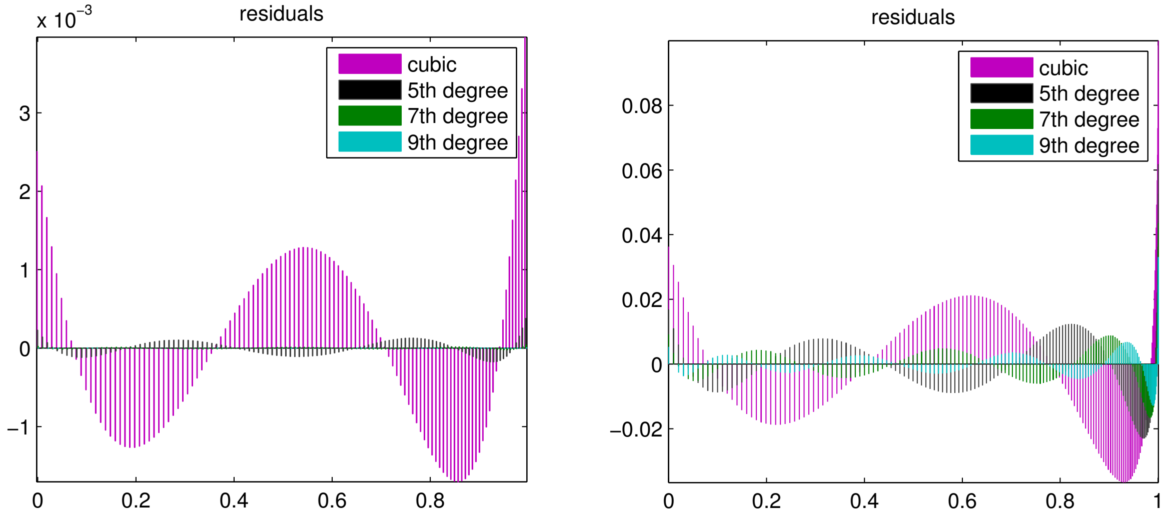


Figure 3.4: Required predistorter output voltage for different values of a_3



(a) polynomial with a3=0.098

(b) polynomial with a3=0.148

Figure 3.5: The residuals of polynomial approximations of different polynomial inverse nonlinearity functions

The higher order distortion caused by the predistorter causes also lower order harmonic distortion. This means that, even if the low-order distortion is completely compensated, the distortion power increases with the power and, at some point, the distortion power passes the maximum allowed distortion level. To reduce the distortion, a higher order polynomial has to be used. This is illustrated in Figure 3.6. The figure shows the powers of third, fifth and seventh harmonic distortion components of a sine signal amplified with an amplifier with third order polynomial AM-AM distortion function with $a_3 = 0.148$. The amplifier is linearized with third and fifth order polynomial predistorters.

It can be seen that, in the absence of predistortion, only the third harmonic is present. The addition of the predistorter generates the higher order harmonic components. It can be seen that the third order predistorter removes the third order distortion as the slope of the remaining third order harmonic power curve now is the same as for the fifth order harmonic power curve and thus is caused by the fifth order distortion. The same can be seen for the fifth order predistorter with the exception that the slopes of the third and fifth order harmonics have the same slope as the seventh order harmonic thus signifying the compensation of the lower order distortion. As can be seen, the third order harmonic distortion caused by the higher order distortion increases more and more rapidly with the signal power. This means that, to reach a certain distortion level, the required order of the polynomial increases more and more rapidly as the signal amplitude increases. Finally, it can be seen that, at the point where the PA becomes non-predistortable, the distortion level curves cross each other.

When the PA distortion is not polynomial, the amplitude of the signal affects the order of the polynomial required for good approximation of nonlinearity. When the PA is far from saturation, the distortion can be approximated with a low-order polynomial, but, when the PA approaches saturation, the required order of the polynomial increases. Figure 3.7 illustrates this effect. The figure shows the RMS error of polynomial approximations of the Saleh distortion model (Section 2.6.1). As can be seen, the required order of polynomial for certain approximation accuracy increases with the amplitude. This increase in required polynomial order is in addition to the effects described in the previous paragraph.

3.4.2 Piecewise constant predistortion function

The accuracy of the approximation of the predistortion function can be increased by using a piecewise constant approximation of the required transfer function instead of polynomial. This makes it easier to linearize PAs having nonlinearities at the low amplitudes or operating near to saturation as the shape of the function can be selected more freely.

In the piecewise constant approximation, the predistortion function therefore has the form

$$A_{PD} = A_{PA}^{-1}(\lfloor N \left| v_{in} \right|^{m} \rfloor / N), \tag{3.7}$$

where $A_{PA}^{-1}(\cdot)$ is the inverse transfer function of the PA nonlinearity, N is the number of different

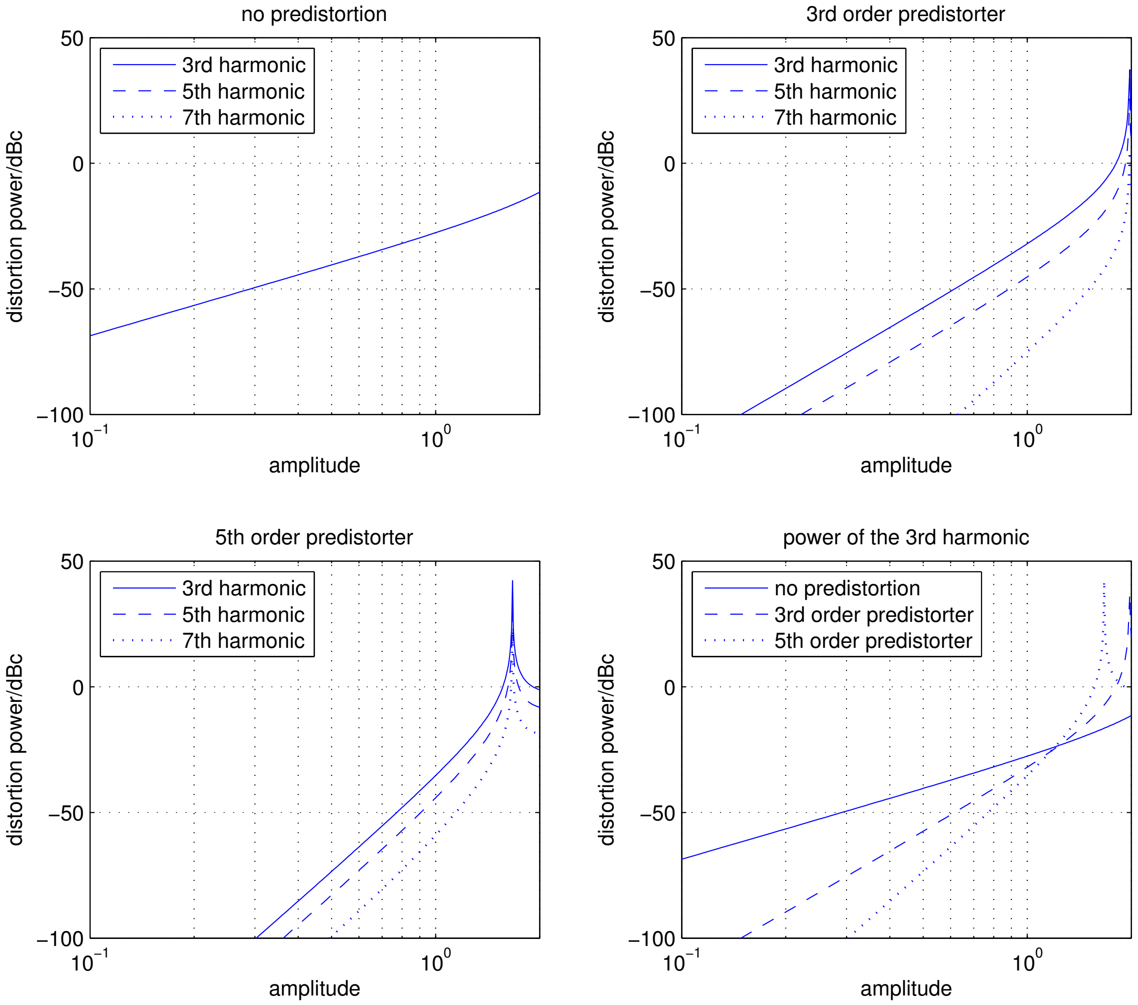


Figure 3.6: The distortion components of a sine signal amplified with a third order polynomial PA with a=0.148

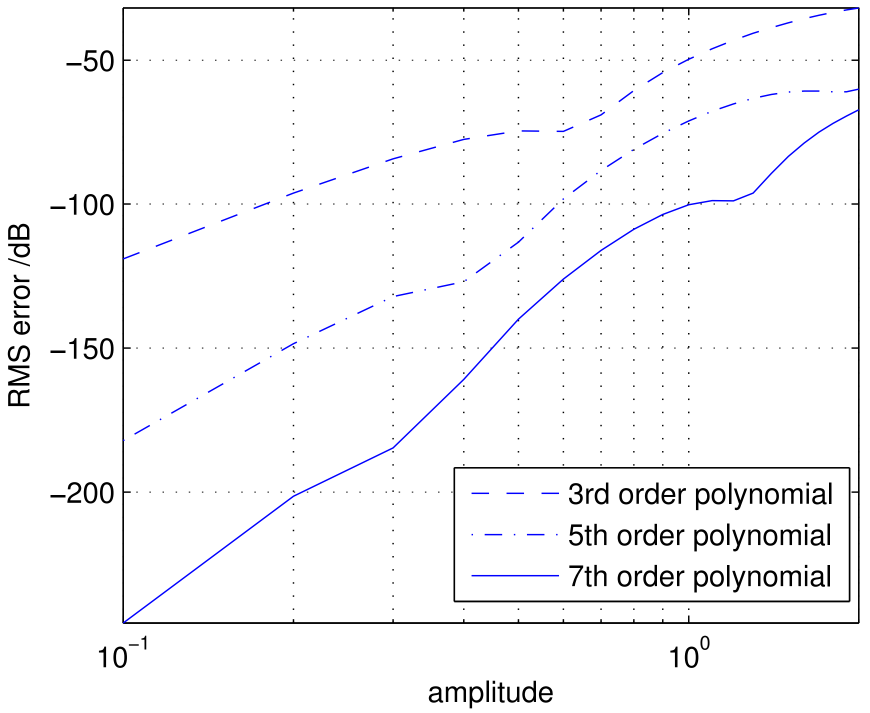


Figure 3.7: The RMS error of polynomial approximations of the Saleh PA

values in the piecewise constant approximation and m is 1 or 2, depending on if the signal amplitude or power is used as the selection parameter for the proper part of the piecewise constant function.

In analog domain, the piecewise constant predistortion function can be implemented with a resistor-selected gain in an operational amplifier circuit [106] or separate amplifiers for each value of the piecewise constant function [3].

Figure 3.8 shows an example of an inverse transfer function of a third order PA nonlinearity and piecewise constant approximation of it with 16 levels. As can be seen, the approximation follows the original function well, but when the derivative increases the steps become larger and the quantization error increases. The transform into piecewise constant function corresponds to quantization of the input parameter of the transfer function. By increasing the number of different values in the function, the steps can be made smaller and the quantization errors reduced [107]. There have been also proposals to reorganize the middle points of the steps in such a way that the step size of the quantized function is constant throughout the whole amplitude range or is minimized at the most probable amplitudes [78, 108–110]. This is discussed in more detail in Chapter 9.

However, both of these methods require complicated hardware if implemented using analog circuits; thus the accuracy of these methods in analog domain is fairly limited. The piecewise constant approximation, however, is very suitable for implementation using digital circuits. This will be discussed in the next section.

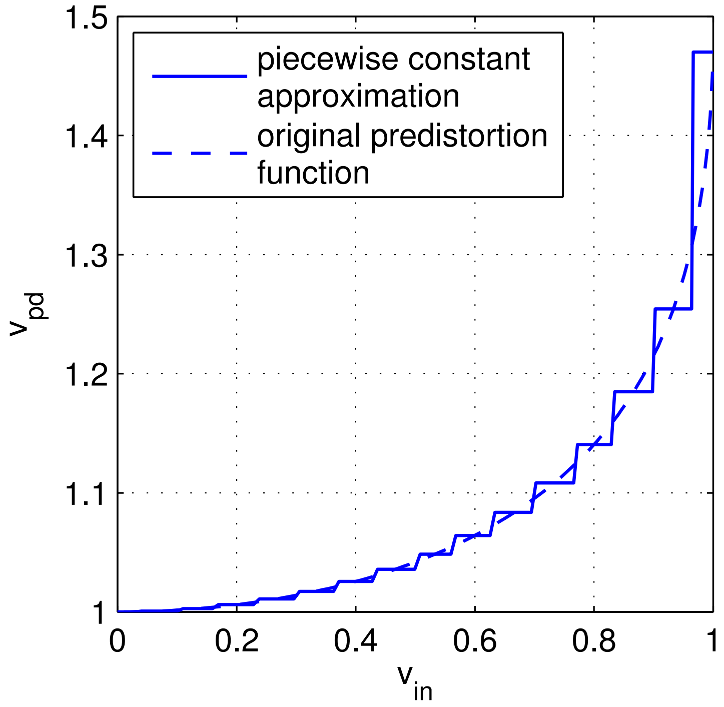


Figure 3.8: Piecewise constant predistortion function

3.5 Digital predistortion

Implementing either the control of the predistorter or the whole predistorter digitally can make the predistortion much more effective. The digital implementation allows more complex predistortion functions and arbitrarily high accuracy if the necessary computation capacity is available. Digital predistortion also easily allows dynamic control signals for the predistortion element, whereas analog predistorters usually use static control [3]. The main types of digital predistortion systems are the baseband predistortion systems and RF-predistortion systems.

The baseband predistortion systems implement the predistortion operation fully digitally at the baseband, which makes it possible to implement diverse predistortion functions. The digital baseband predistortion systems can be further divided into three categories: mapping predistorters, complex gain predistorters and polar predistorters. Mapping predistorters [79] use the complete complex baseband signal to generate the predistortion function and thus are also able to correct phase-dependent distortion such as modulator errors but are quite hardware inefficient. Complex gain predistorters [11] take advantage of the amplitude dependence of the distortion and use only the absolute value of the signal to generate the complex valued predistortion function, thus reducing the hardware requirements. Polar predistorters [91] use the amplitude of the signal for predistortion function generation and use the polar form of the complex signal for the predistortion. The different baseband predistorters are discussed more thoroughly in Chapter 8. Figure 3.9 shows the basic structure of a digital baseband predistortion system.

The digital RF predistorters usually implement the actual predistortion using analog compo-

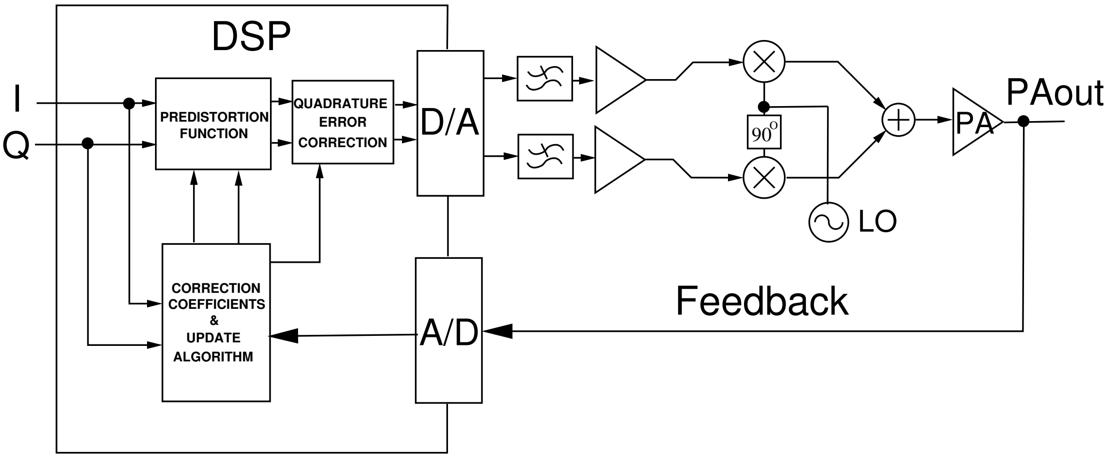

DSP
I
Q
PREDISTORTION FUNCTION
QUADRATURE ERROR CORRECTION
D/A
CORRECTION COEFFICIENTS & UPDATE ALGORITHM
A/D
90°
LO
PA
PAout
Feedback

Figure 3.9: Block diagram of a baseband digital PA predistorter.

nents and only use DSP for the control signal generation. This is due to the fact that it usually is not feasible to transform the RF signal to digital domain for predistortion. However, fully digital RF predistorters have also been published [111, 112]. Usually, the predistortion element consists of a phase and amplitude modulators [104] or quadrature modulators [102]. The controls for these predistortion elements are generated according to the A/D converted detected envelope of the RF signal; thus the RF predistorters resemble the complex gain and polar baseband predistorters. Figure 3.10 presents a basic block diagram of an RF predistorter. The RF predistorters are discussed in more detail in Chapter 4.

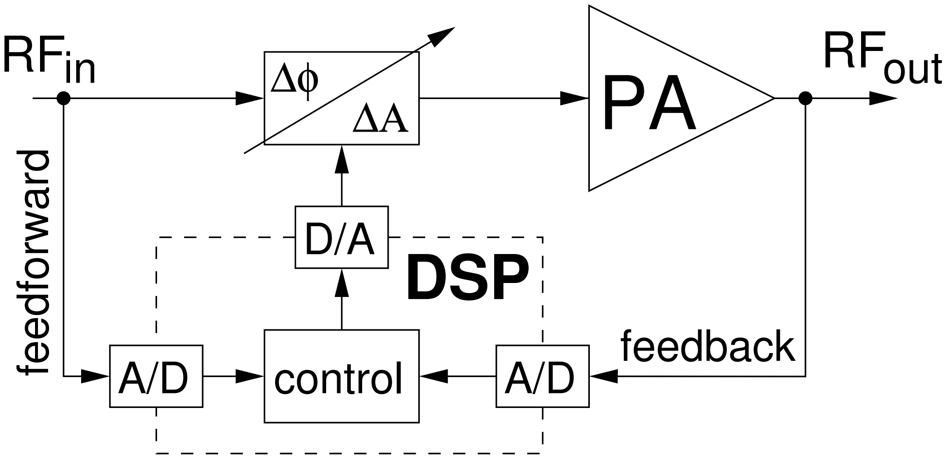


Figure 3.10: Basic block diagram of digitally controlled RF predistorter

The most common method to implement the predistortion function in digital domain is the piecewise constant approximation. This is due to the fact that the piecewise constant function can be easily implemented using a LUT that is indexed with the parameter of the predistortion function (complex signal or the signal amplitude) and contains the predistortion function values at the points defined by the index values. So, each entry corresponds to one gain value in the analog implementation (Section 3.4.2). The digital implementation allows the number of steps in the function to be increased with much smaller hardware consumption, thus reducing the quantization errors described in Section 3.4.2 [107]. Also, the redistribution of the pieces of the function to reduce the quantization errors becomes easier with the help of digital algorithms [78, 108–110].

It is possible to implement a polynomial predistortion function directly in digital domain using multipliers and adders and especially many digital predistorters with memory actually use memory polynomials or Volterra series for the predistortion [53, 56, 86, 105, 113, 114]. However, often a piecewise constant approximation of the polynomial is used and the polynomial is calculated only when the LUT is updated. This avoids the calculation of the value of the polynomial for every sample, that may be a limiting factor for the clock rate, especially for high-order polynomials. Instead, the value of the polynomial is calculated once when the predistortion function changes and is stored in a LUT; thereafter, the system behaves exactly as any other predistorter based on piecewise constant approximation. The reasons for using a polynomial function to calculate the LUT values are discussed in Chapter 9.

3.5.1 Errors caused by digital predistortion

Although the use of DSP allows generation of more complex predistortion functions than the analog predistortion, the use of digital signals and LUTs also introduces errors to the predistortion. The most obvious error source is the quantization of the predistortion signals. In the baseband predistorters, this is not a very significant error source as the signal word lengths are already decided by the baseband circuitry. In the RF-predistorters on the other hand the predistortion is implemented using analog control signals that normally have infinite quantization accuracy, but when digital control is used they are A/D- and D/A-converted and quantization noise is introduced.

Another source of quantization error is the piecewise constant predistortion function. As was discussed in Section 3.4.2, this corresponds to quantization of the signal used as the function parameter (the LUT index) into as many discrete steps as there are entries in the LUT. Thus, using a 256-entry LUT corresponds to quantizing the indexing signal to 8 bits. Due to large hardware consumption, the number of LUT entries is usually less than the word length of the predistorter input and output signals would allow. A basic and widely known formula for calculating the SNR caused by the quantization is

$$SNR = (6.02N_{bit} + 1.76 + 10\lg OSR)\, dB, \tag{3.8}$$

where N_{bit} is the word length in bits and OSR is the oversampling ration of the signal. This formula can be used to approximate the effect of the quantization on the noise floor of the predistorted signal. This formula can be also used to approximate how many bits of word length a change in the SNR or OSR corresponds to:

$$\Delta N_{bit} = \frac{\Delta SNR}{6.02} \tag{3.9}$$

$$\Delta N_{bit} = 10\lg \frac{OSR_{new}}{OSR_{old}} \tag{3.10}$$

These formulas can be used to evaluate the usefulness of a change in the predistortion design that affects the SNR or OSR of the system and reduces the size of the LUT. The quantization effects and methods to reduce them have been discussed in for example [78, 107–110] and are also discussed in Chapter 9.

Also the discrete time may affect the performance of the predistortion systems. This makes it impossible for the time delay of the digital part to be adjusted freely, and possible only in discrete steps defined by the clock frequency. The achievable linearity of a digital predistortion system with time-domain feedback signals, on the other hand, is dependent on how accurately the delays of the feedback and original signal can be matched before comparison [79, 115]. To match the signals accurately enough fractional delay filters or an increase in the clock frequency may be required. In RF-predistortion systems, in addition of the feedback delay matching, the delay matching of the predistorter control signals and the RF signal also affect the linearity and thus this is adversely affected by the discrete time. The effects of the delay will be discussed in more detail in Chapter 5.

3.5.2 LUT update

The use of DSP for the predistortion eases implementation of an adaptive predistorter that can follow the changes in the PA nonlinearity. The adaptive predistortion can be performed by updating the predistortion function stored in the LUT one entry at a time or the whole LUT at once. The update can be made according to ACP or other spectral-domain measurements of the PA output or the measured time-domain differences between the predistorter input and PA output signals. The spectral-domain measurements are more suitable for methods in which the whole LUT is updated simultaneously, such as polynomial predistortion, whereas the time-domain measurements are suitable for updating each LUT entry separately. Different LUT update methods are discussed in more detail in Chapter 9. However, the following paragraphs discuss two commonly used LUT update methods for updating LUT entries separately based on time-domain measurements namely linear update and the secant method. The linear update is used mainly in the predistorters investigated in this thesis due to its simplicity.

LUT update based on comparison of instantaneous envelope values has been used mainly in baseband predistorters [11, 79, 91] but it is also suitable for RF-predistortion [10]. The most common methods for the LUT adaptation when using time-domain comparison are the secant method and linear iteration. Linear iteration can be derived by the method of successive substitutions [116] as

$$LUT_{n+1}\left(V_{in}(n)\right) = LUT_{n}\left(V_{in}(n)\right)\left(1 + a\frac{\left(V_{out}(n) - V_{in}(n)\right)}{V_{out}(n)}\right) \tag{3.11}$$

In polar- and RF-predistortion systems, the division by $V_{out}(n)$ can be omitted at the cost of slower convergence. However, removal of the division from the algorithm is a significant advantage. In the case of the baseband predistorter, this tends to cause instability. The updated algorithm can be written as

$$LUT_{n+1}\left(V_{in}(n)\right) = LUT_{n}\left(V_{in}(n)\right) - a\left(V_{out}(n) - V_{in}(n)\right), \tag{3.12}$$

this form is also used in mapping predistorters [79].

The secant method offers faster convergence but incurs the cost of increased hardware complexity as it requires information about previous LUT and signal values. The secant method[117] can be written as

$$\begin{array}{l} LUT_{n+1}(V_{in}(n)) = \\ LUT_{n}(V_{in}(n)) - a \cdot \Delta V(n) \frac{LUT_{n}(V_{in}(n)) - LUT_{n-1}(V_{in}(n))}{\Delta V(n) - \Delta V(n-1)} \end{array}, \tag{3.13}$$

where n is the index of the current iteration, $LUT(V_{in})$ is the value of LUT entry corresponding to the input amplitude value $V_{in}(n)$, $\Delta V(n) = V_{out}(n) - V_{in}(n)$, $V_{out}(n)$ is the PA output envelope and a is a constant that determines the convergence speed of the iteration. The adaptive formulas can be applied to both phase and amplitude LUTs.

Lee et al. [118] compare several LUT update methods in terms of adaptation speed and achieved linearity when using a baseband predistorter and an EDGE signal. The paper shows that the secant method offers clearly faster adaptation than the linear method, but the achieved linearity is slightly worse. Figure 3.11 shows the convergence of the secant method (3.13) and linear method (3.12) with a 64-entry LUT using the same convergence coefficient. The LUT error in the figure is shown as the RMS of the current LUT compared to the final value in dB. The faster convergence of the secant method is clear, but both methods achieve the same final value.

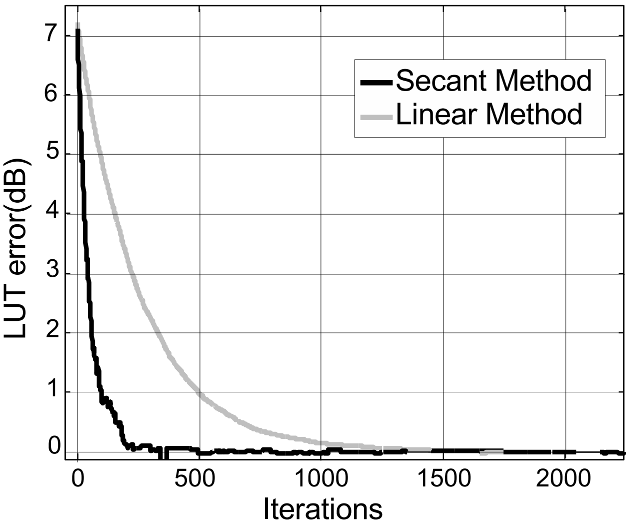


Figure 3.11: Example of convergence properties of (3.12) and (3.13) using a 6 bit LUT

Due to the much smaller hardware costs (no requirement for additional memory elements and less adders) and as the adaptation time was not considered critical for the system, the linear method was chosen as the most suitable adaptation algorithm for the circuits presented in this thesis.

3.6 Predistorters with memory

Recently, especially due to ever-wider bandwidths required for the transmitted signals, the memory predistortion has received considerable attention. The main idea of the memory predistortion is to also take into account the previous values of the predistorted signal to compensate for the nonlinearities of the PA. Without the memory, the predistorter generates the same compensating distortion to all of the frequency components of the signal. This means that as the distortion of the PA with memory is not equal for all frequencies, part of the distortion remains uncompensated. This is illustrated in Figure 3.12 [119]. This leads to unequal distortion powers on adjacent channels.

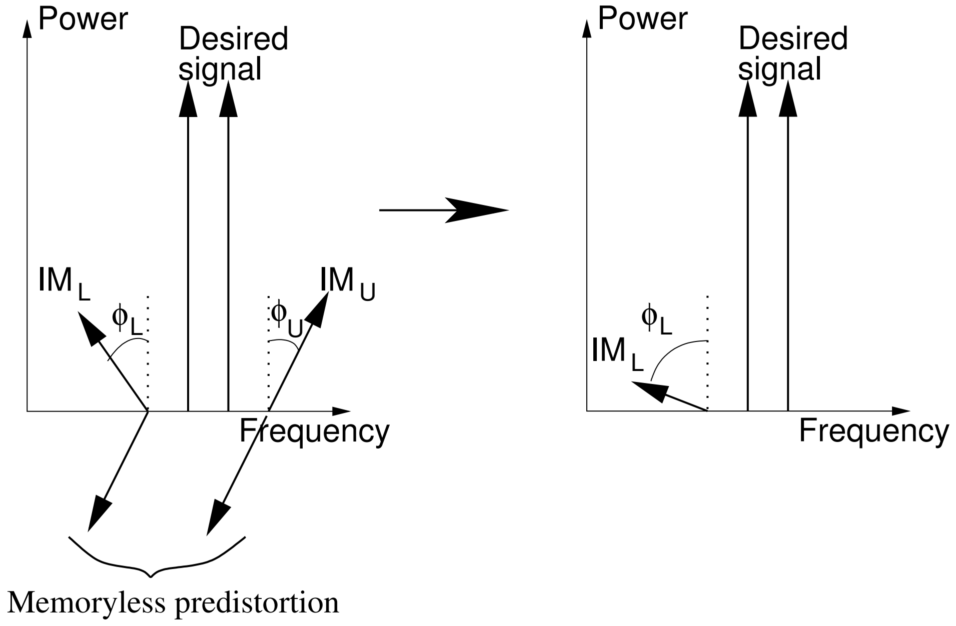
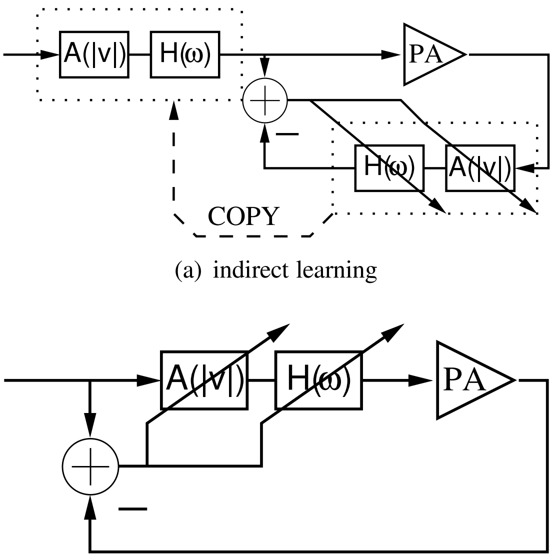


Figure 3.12: The effect of memoryless predistorter on a power amplifier with memory [119].

The generation of the predistortion function with memory is usually based on the same principles as the PA models with memory and thus the models described in Section 2.8.2 can be used also for the predistortion. The most common predistortion systems are the Wiener/Hammerstein/-Wiener-Hammerstein [47, 50, 120–122] and memory polynomial systems [45, 48, 52–56, 61, 123], although Volterra predistorters have also been implemented [46, 124].

The Wiener-Hammerstein and Hammerstein predistorters offer a fairly simple solution for the memory predistorter requiring only one or two filters and the memoryless nonlinearity, which can be implemented using a LUT. The filter coefficients and the LUT can be calculated iteratively, for example, using RLS or LMS methods or through matrix inversions [47, 125].

Commonly, the predistortion function identification is accomplished using the indirect learning algorithm (Figure 3.13a) [126]. The algorithm uses a proxy predistorter that linearizes the PA output signal and thus operates as a post distorter. The postdistorter is adjusted so that the error between the PA input and the postdistorter output is minimized. The postdistorter coefficients are then copied to the actual predistorter in the PA input. The system is based on the assumption that



(a) indirect learning

(b) direct learning

Figure 3.13: The update methods for the memory predistorters [126]

the system consisting of the PA and the predistorter is reversible without affecting the linearity and that the output measurements are noiseless. This, however, is usually not the case, which renders this kind of predistortion identification suboptimal [126].

Figure 3.14 shows an OFDM signal after a PA with memory, the same signal linearized with a memoryless predistorter, the signal after the postdistortion used for the memory predistorter identification and the signal after the predistortion with the memory predistorter identified with the postdistorter. As can be seen, the memory predistortion significantly improves the ACP compared to the memoryless one. However, it can be seen that the linearity of the signal at the postdistorter output is better than at the predistorter output. Thus, some of the linearity is lost, as the adaptation algorithm does not see this difference.

The alternative option for the predistortion identification is the direct learning algorithm (Figure 3.13b) [126], which updates the predistortion coefficients directly based on the difference between the predistorter input and PA output. This method, however, has the disadvantages that the derivatives of the PA distortion function have to be known for the update to be stable [126]; usually the derivatives are not known and thus the direct learning method can not be used straightforwardly. Several solutions have been suggested to find these derivatives [121, 127, 128], but they all increase the complexity of the system.

The Wiener and Hammerstein predistorters suffer from the same problems caused by oversimplification as the Wiener and Hammerstein PA-models (Section 2.8.2). The Volterra predistorters [46, 124] offer the possibility of a more accurate predistortion function, but require a large amount of hardware. Recently, a very common method for memory predistortion has been the memory polynomial [45, 48, 52–56, 61, 123], which is similar to the memory polynomial used for the PA identification (2.23). The memory polynomial-based predistortion systems can be implemented with the same kind of structure as the memory polynomial-based PA model (Figure 2.12). The

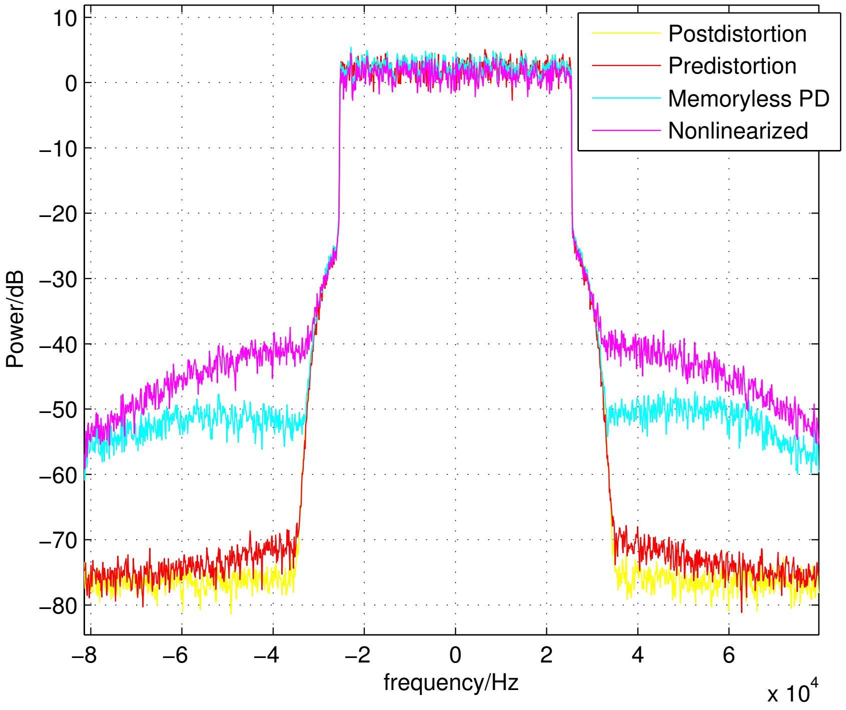


Figure 3.14: OFDM signal amplified with a PA with memory and linearized with a predistorter and postdistorter.

memoryless nonlinearities can be implemented with LUTs, and thus the whole design can be implemented with fairly low hardware costs.

The published predistorters with memory are invariably baseband predistorters [45–48, 50, 52–56, 61, 120–124, 129]. This arises from the fact that RF predistorters do not usually have information about the actual RF signal but, instead, use only the envelope of the signal. The problem can be seen with the following analysis: For simplicity we assume that the filter is an FIR filter and that we have a polynomial memoryless predistorter. Thus, the memoryless predistorter output is

$$v_{pdout}(t) = \sum_{n=0}^{\infty} a_n v_{in}(t) \left| v_{in}(t) \right|^n \tag{3.14}$$

and the filter output is

$$v_{filtout}(t) = \sum_{m=0}^{\infty} b_m v_{pdout}(t-m). \tag{3.15}$$

When these are combined, we get as the memory output of the predistorter

$$v_{filtout} = \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} b_m a_n v_{in}(t-m) \left| v_{in}(t-m) \right|^n . \tag{3.16}$$

The RF predistorted signal, on the other hand, has the form

$$v_{RFpdout}(t) = v_{in}(t)H\left(|v_{in}(t)|\right). \tag{3.17}$$

Now, if we modify (3.18) to correspond to the form of (3.17) we get

$$v_{filtout} = v_{in}(t)\sum_{m=0}^{\infty}\sum_{n=0}^{\infty} b_m a_n \frac{v_{in}(t-m)}{v_{in}(t)} \left|v_{in}(t-m)\right|^n. \tag{3.18}$$

The problem with this formula is that the predistorter would require information on the previous values of the signal, not just the envelope. The signal values could be extracted using a quadrature demodulator, but the complexity of this solution is so high that the baseband predistorter is clearly a better solution. We can approximate that $\frac{v_{in}(t-m)}{v_{in}(t)} = 1$, but this solution requires a high sampling frequency and short filters, otherwise significant error is introduced to the calculation. Similar analysis can be performed on the memory polynomial predistorter. Thus, an RF-predistorter with memory would require a completely new predistortion algorithm. This, however, is not the focus of this thesis and will not be explored further.

3.7 Survey of published digital predistortion systems

A large number of digital predistortion systems have been published since the first examples of these were presented. Table 3.1 presents some examples of the performance and specifications of published predistortion systems. As can be seen the table sizes for the RF, complex gain, polar and memory predistorters varies from 32 to 1024 entries. The predistorter word lengths are usually between 12 and 16 bits and the RF predistorters tend to have shorter word lengths. The measured ACP improvements vary between 10 dB and 20 dB and the simulations tend to give ACP improvements near to 30 dB. However, it should be noted that the improvements depend on the nonlinearity of the PA also.

3.8 Conclusions

This chapter presented the basics of PA linearization methods. The common methods were discussed and the chapter especially concentrated on the predistortion linearization. The general operation of analog and digital, memoryless and memory predistorters was discussed. A survey of published predistorters was presented and showed that the simulated performance of predistorters is significantly better than the measured performance.

The rest of this thesis will study the digital RF and baseband predistorters and their implementation in more detail and will include a detailed analysis of some of the problems related to the digital predistortion and suggest solutions for some of them.

Table 3.1: Survey of published predistortion systems

Ref	Predistorter type	RF/BB	Memoryless?	Clock frequency	Signal bandwidth	Measured?	ACP improvement	memory size	word length
[8]	Phase/Amplitude	RF	YES	105MHz	?	2tone	25dB	64	8bit
[79]	Mapping	BB	YES	128kHz	32kHz	Yes	$\approx$ 20dB	$2 \cdot 10^6$	10bit
[11]	Complex gain	BB	YES	240kHz	30kHz	No	$\approx$ 30dB	32	?
[91]	Polar	BB	YES	35kHz	2.18kHz	Yes	$\approx$ 20dB	64	16bit
[9]	Phase/Amplitude	RF	YES	?	40kHz	Yes	$\approx$ 20dB	?	?
This thesis	Phase/Amplitude	RF	YES	10MHz	18kHz	Yes	10dB	256	12bit
This thesis	Complex gain	BB	YES	1MHz	18kHz	Yes	15dB	64	16bit
[10]	Phase/Amplitude	RF	YES	?	1.2MHz	Yes	$\approx$ 8dB	?	?
[105]	Complex gain	BB	YES	125MHz	$\approx$ 20MHz	No	30dB	?	13bit
[53]	Memory poly	BB	No	?	3.84MHz	Yes	16dB	?	16bit
[114]	Quadrature	RF	Yes	100MHz	3.7MHz	Yes	12dB	1024	12bit
[129]	Hammerstein	BB	No	?	3.84MHz	Yes	$\approx$ 16dB	1024	?
[55]	Hammerstein	BB	No	?	15MHz	Yes	$\approx$ 20dB	?	16bit
[110]	Quadrature	RF	Yes	65MHz	15MHz	Yes	$\approx$ 10dB	256	14bit
[111]	Digital	RF	Yes	?	3.68MHz	No	$\approx$ 15dB	?	12bit
[130]	Mapping	BB	Yes	?	?	No	$\approx$ 30dB	$22 \cdot 10^4$	?

Chapter 4

RF/IF- predistortion

4.1 Introduction

Analog RF/IF predistortion has been in use in, amongst others, satellite communications systems, due to its ability to be implemented with very simple circuitry [3], such as constant biased diodes. However, when a more significant linearity improvement is required, usually the complexity of the RF/IF predistortion increases [3]. Digital implementation of the predistortion control simplifies design.

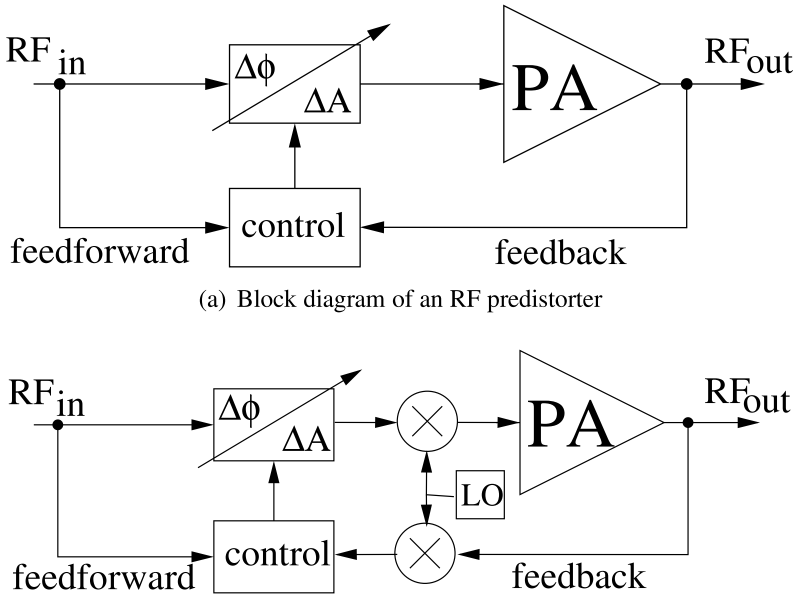
One of the goals of this thesis was to investigate the possibility to simplify the design of an TETRA transmitter by replacing the previously used linearization methods with a predistorter. The RF predistorter was considered to offer the possibility for a very simple linearizer and it also has promising characteristics that could be utilized to implement an universal predistortion circuit. Therefore it was chosen as the main research subject of this thesis.

This chapter gives an introduction to implementing PA predistortion on carrier or intermediate frequencies. First the benefits and problems present in RF predistorters in general are discussed, but after that the chapter will concentrate mainly on the more complex predistortion designs using variable control signals and especially on the digital implementations these. The common architectures for RF predistorters are presented. At the end of the chapter, some design considerations related to digital predistorters will be reviewed.

4.2 RF/IF-predistortion systems

An RF/IF- predistorter linearizes a PA by altering the up-mixed signals to counter the PA nonlinearity. As the name implies, the RF predistorter alters the PA input signal at the carrier frequency and uses an analog nonlinearity element to generate the predistortion function. A significant advantage of this method is that the predistorter does not necessarily require any up- or down-mixing operations [9]. This allows an implementation that is not dependent on the exact carrier frequency of the signal or the baseband circuitry. This is significant, since, without the dependence on the baseband circuitry, it is possible to develop a completely separate predistortion chip or a PA chip that includes the predistortion, thus, from the designer's point of view, looking like a highly linear power amplifier. This kind of chip would make the design of linear transmitters more clearly partitioned, as the baseband designer would not need to bother her- or himself with the linearity issues in the RF parts, just as the RF designer does not need to bother him- or herself with the baseband design to make the PA linear. Also, it would ease the design of linear transmitters from on-the-shelf components, as a suitable baseband design and then a linearized PA could be selected without dependence on the selection of the other part or having to design any additional feedback paths to the baseband.

The RF-predistorter also only has to operate on a single RF signal instead of several baseband signals, which eases the design when using analog elements [3]. However, in digital predistortion systems, this advantage is not so significant. An RF predistorter requires that the predistortion element has to be able to cope with the high frequencies, which poses some limits on the method [3]. Figure 4.1a shows the basic block diagram of an RF-predistortion system. If no suitable predistortion elements are available due to the frequency limitation, the predistortion can also be performed at a lower intermediate frequency before up mixing. This method is called IF-predistortion [3]. A block diagram of an IF-predistortion system is shown in Figure 4.1b. IF-predistortion gives up some of the advantages of the RF predistorter, especially since the feedback path for the possible adaptation may require a down-mixer.



(a) Block diagram of an RF predistorter

(b) Block diagram of an IF predistorter

Figure 4.1: basic RF

A basic RF predistorter can be implemented with a very small number of components. This is enabled by the fact that the most dominating distortion mechanism in RF power amplifiers is the AM-AM distortion, and thus the rudimentary linearity improvement can be gained by generating a simple approximation of the inverse amplitude distortion function of the PA. In fully analog RF predistorters, a third order amplitude distortion function is often used [3]. This can be simply implemented with a properly biased nonlinear diode or single transistor. This method, although it has been quite widely used [3, 85, 92–97], is not, however, very accurate and thus has only a limited correction ability [3]. More complex AM-AM predistortion methods have also been presented [3, 98].

When more linearity is required, the AM-PM distortion also has to be taken into account and thus an efficient correction method is to use separate phase and amplitude distortion circuits [85, 99–101]. The distortion function of the analog components still limits the correction ability. The problem can be somewhat alleviated by using a curve fit approximation of the nonlinearity [3]. However, the analog implementation of a curve fit predistorter is quite cumbersome.

Some of these problems can be solved by using control circuitry that adaptively generates the predistortion function and a vector modulator circuit that can be used to generate any required phase shift or amplitude distortion function [81, 131, 132]. However, complicated circuitry is again required and the predistortion function still has its limitations.

A solution to these problems is to use digital signal processing to generate a curve fit or high-order polynomial approximation of the control signals required to generate the inverse of the PA nonlinearity in the analog predistortion elements. The simplest way to implement this kind of digital control is to use an envelope detector to sample the power amplifier input signal and then A/D convert it to generate a digital control signal. The envelope is then used as a dynamic control signal for the predistortion function generation circuit. The predistortion function generation is usually based on storing predistorter control into one or more look up tables implemented by RAM or ROM blocks. Thus a digital RF predistorter is usually based on a piecewise constant approximation of the nonlinearity. The digital implementation of the control signals also makes it easier to implement an adaptive update of the predistortion functions.

4.3 Digitally controlled RF-predistortion system

Unlike the baseband predistortion system, it is not feasible to implement the RF-predistortion system fully digitally, as this requires transforming the analog RF signal into digital domain and back. Even if this kind of implementation were possible hardware-wise, usually it would be much more efficient to implement the whole transmitter chain up to the predistorter digitally. This is due to the fact that the conversions between digital and analog domains and the possible up- and down-mixing increase noise and reduce linearity. However, this kind of implementations have also been presented [111, 112].

For these reasons, only the control of the RF-predistortion system is usually digitalized [8–10, 28, 86, 102, 110, 114, 133–137]. The block diagram of a digitally controlled RF predistorter is shown in Figure 4.2.

The digital control enables more flexible control signal implementation and update than a

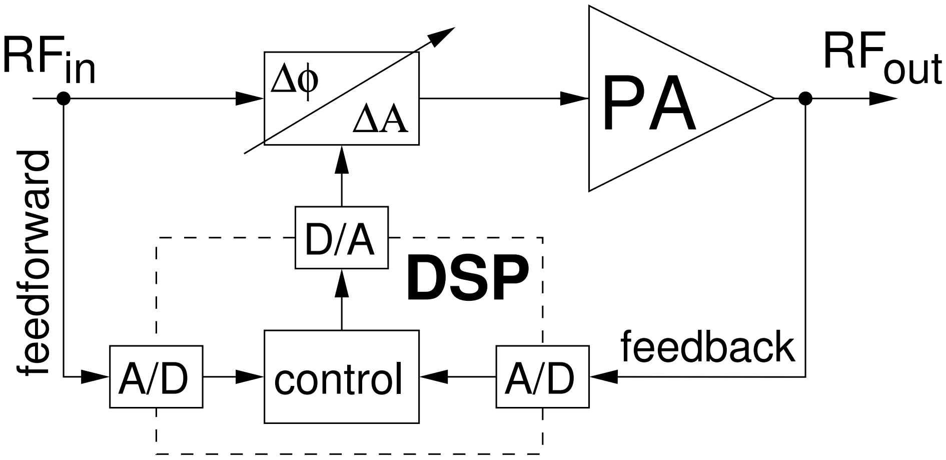


Figure 4.2: Basic block diagram of digitally controlled RF predistorter

fully analog solution. Usually, the control is based on the signal envelope [8–10, 28, 86, 102, 110, 114, 133–137], although temperature-dependent control circuits have also been presented [80]. As is the case with the analog counterpart, the implementation can be accomplished without knowledge of the exact carrier frequency.

4.4 Digitally controlled RF-predistortion system types

The digital RF-predistortion systems can be divided into several different categories depending on the type of the analog predistortion element, the feedback method, use of polynomial or piecewise linear LUT. The following sections discuss these categories, except for the selection between the polynomial and piecewise constant LUT that was discussed Chapter 3.

4.4.1 The implementation of the analog predistortion element

The predistortion device, with the help of the control signals, generates such a phase and amplitude distortion that the overall phase shift and gain of the system remains constant over the whole amplitude range. This can be achieved by using an amplitude and phase modulator [8–10, 134, 136] or a quadrature modulator [28, 86, 102, 110, 114, 133, 135, 137].

The amplitude-and-phase-modulator-based solution applies separate amplitude and phase distortion functions in series to the RF signal. If it is assumed that the phase modulator does not affect the amplitude of the signal and vice versa, then the modulators are completely independent of each other; thus their control signals can be calculated or updated separately. However, this is not usually the case. More often, the AM-AM and AM-PM correction functions affect each other, and thus an iterative approach for the calculation gives more optimal results.

The operation of an amplitude-and-phase-modulator-based predistorter [8–10, 134, 136] (Figure 4.3) can be described with the following formula:

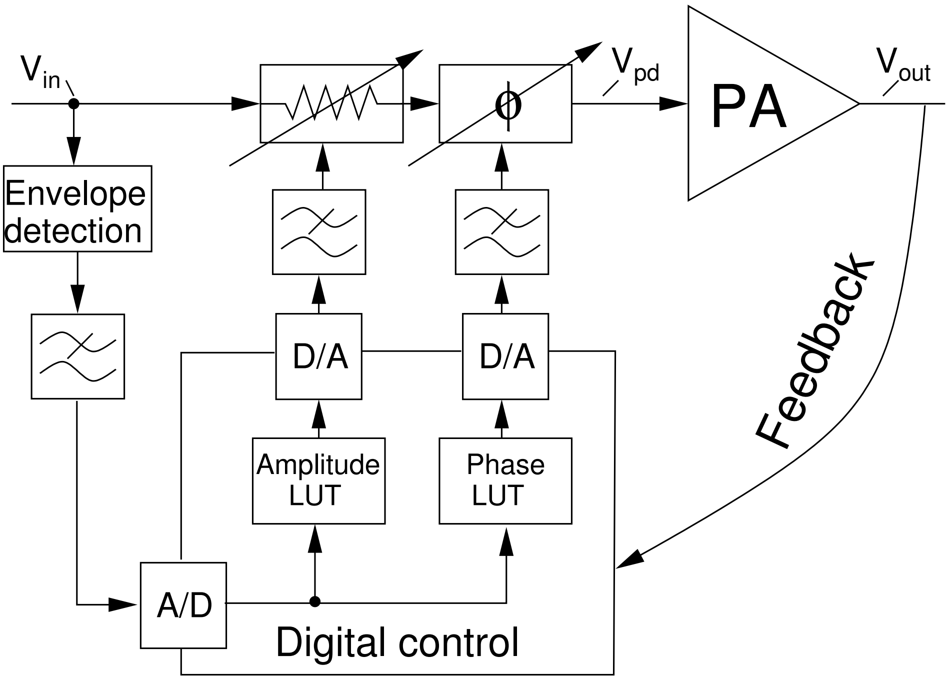


Figure 4.3: RF predistorter based on phase and amplitude modulator

$$\begin{cases} \mathrm{A}_{PD}\left(|v_{in}|\right) \cdot \mathrm{A}_{PA}\left(|\mathrm{A}_{PD}\left(|v_{in}|\right) v_{in}|\right) = K \\ \Phi_{PD}\left(|v_{in}|\right) + \Phi_{PA}\left(|\mathrm{A}_{PD}\left(|v_{in}|\right) v_{in}|\right) = P \end{cases}, \tag{4.1}$$

where $\mathrm{A}_{\mathrm{PD}}(\cdot)$ and $\mathrm{A}_{\mathrm{PA}}(\cdot)$ are the predistorter and PA gain, respectively, and $\Phi_{PD}\left(\cdot\right)$ and $\Phi_{PA}\left(\cdot\right)$ are the predistorter and PA phase shift, respectively. $|v_{in}|$ is the input signal envelope. K and P are constants. This can be interpreted as a requirement that the gain and the phase shift of the system is constant regardless the amplitude.

The quadrature-modulator-based predistorter [28, 86, 102, 110, 114, 133, 135, 137] (Figure 4.4) uses a 90^ophase shifter to separate the RF signal into two quadrature branches that are fed to a quadrature modulator. The two quadrature predistortion functions affect each other and therefore they should both be updated simultaneously.

For the quadrature-modulator-based system $A_{pd}\left(|v_{in}|\right)$ in (4.1) is replaced with

$$A_{pd}\left(|V_{IN}|\right) = \sqrt{\Delta I(|V_{IN}|)^2 + \Delta Q(|V_{IN}|)^2} \tag{4.2}$$

and $\Phi_{PD}\left(|V_{IN}|\right)$ in (4.1) is replaced by

$$\Phi_{PD}\left(|V_{IN}|\right) = \tan^{-1}\frac{\Delta Q(|V_{IN}|)}{\Delta I(|V_{IN}|)} \tag{4.3}$$

where $\Delta I(|V_{IN}|)$ and $\Delta Q(|V_{IN}|)$ are the predistortion functions for the I branch and the Q branch, respectively.

The main differences between the implementations are, that the amplitude-and-phase-modulator-based design is closer to the usually assumed model of the PA based on separate AM-AM and

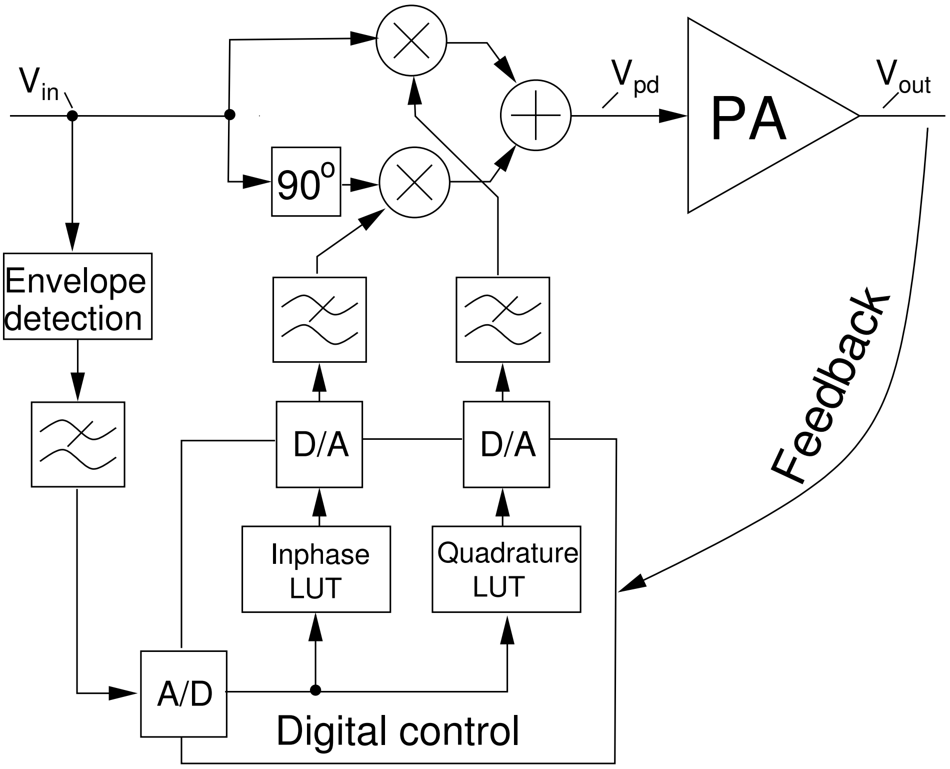


Figure 4.4: RF-predistorter based on quadrature modulator

AM-PM distortions. This also allows studying the effect of AM-AM correction and AM-PM correction separately [91]. The quadrature-modulator-based implementation, on the other hand, combines the amplitude and phase distortion into one complex valued function which simplifies the implementation of the correction and update as the calculations can be implemented using complex arithmetic.

In addition to these differences, the availability of the suitable phase and amplitude modulator and quadrature modulator devices for the required frequency, the signal power range, and the implementation of the feedback define which implementation should be used.

4.4.2 Time-domain and frequency-domain feedback

If the predistortion is to be implemented adaptively, a feedback of the linearity improvement is required. The feedback can be based on instantaneous time-domain information [10, 86, 102, 114] or frequency-domain information (or a similar measure of distortion collected over a longer time span). [9, 28, 110, 133–137]

The time-domain feedback method (Figure 4.5) compares phase and amplitude samples or samples of the quadrature demodulated signal at the predistorter input and PA output. Then the LUT is updated according to the difference of these measurements so that the error is minimized. The comparison and update can be performed either in real time or by collecting a larger number of samples and then doing a batch update to several LUT entries at a time [138]. The real-time update can be implemented with very simple hardware, but is more prone to instability a[10]. The time-domain update can be performed with simple secant or linear iteration methods, which are suitable for updating one entry at a time, or with more complex LMS, RLS etc. methods, which update the whole LUT simultaneously [10, 86, 102, 114].

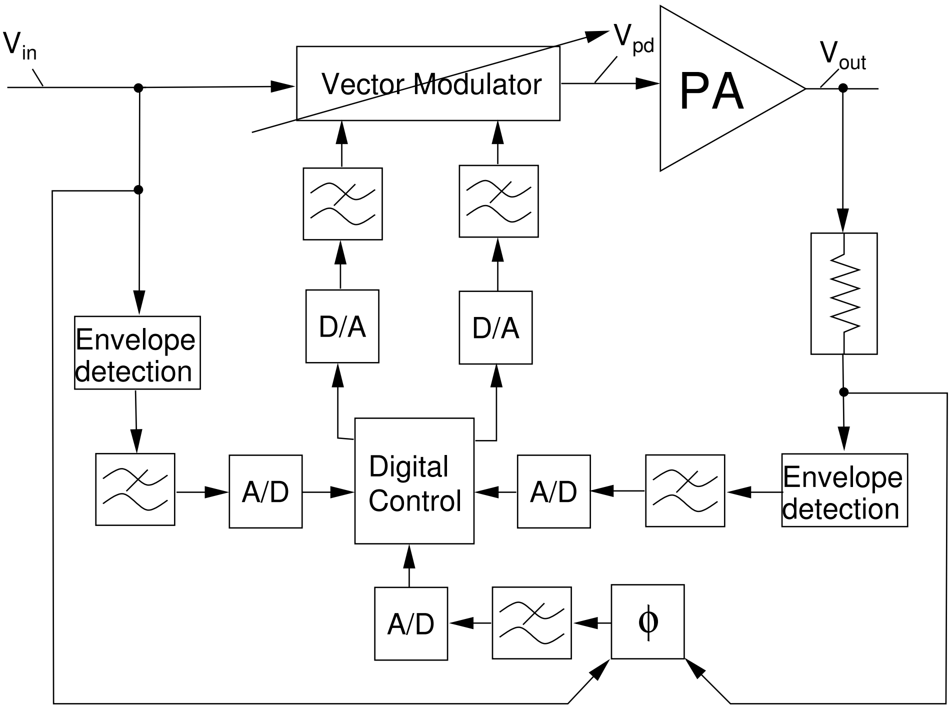


Figure 4.5: Time-domain feedback of envelope and phase

The time-domain feedback requires accurate matching of the input and output signals of the system in time-domain as well as accurate matching of the transfer functions of the input and output signal paths. Inaccurate matching reduces the linearity and may even cause instability.

In this thesis, the frequency-domain update is used as a common name for predistortion function update methods that use distortion measures that are slowly changing, independently of the input signal to the predistortion system, and amalgamate errors in several LUT entries simultaneously [9, 28, 110, 133–137]. These include methods that measure the transfer function of the power amplifier directly [110, 135, 137]. The reason is that the frequency-domain measure was the first to be presented [28, 133] and the independence of the input signal makes the methods similar. As these methods use an aggregate error measure such as adjacent channel power, all the LUT values [9, 28, 110, 133–137] or a group of LUT values [134] are updated simultaneously. This makes the polynomial control signal generation very suitable for frequency-domain update, as the number of parameters is low and, at best, only one measured parameter is required for the update. Thus, the hardware requirements for the update are reduced (at the expense of the accuracy of the predistortion function).

The frequency-domain update avoids the requirement for matching the detector and the delay as the input and output data of the system are not compared, but the measurement of the distortion metric may require complex circuitry. Proposed designs use, for example, measuring the out-of-band distortion through band pass filtering [9, 28, 133, 134, 136] or direct measurement of the PA nonlinearity [110, 135, 137].

In the out-of-band distortion measurement method (Figure 4.6), the signal band is removed from the measured signal with bandstop filters. This method only measures the output signal, so there is no delay-matching requirement. However, the signal band filters require steep transition bands and linear phase response not to cause error to the measurement. Also, any residual signal band power will affect the correction. As the signal band is filtered out, any in-band distortion is left uncorrected.

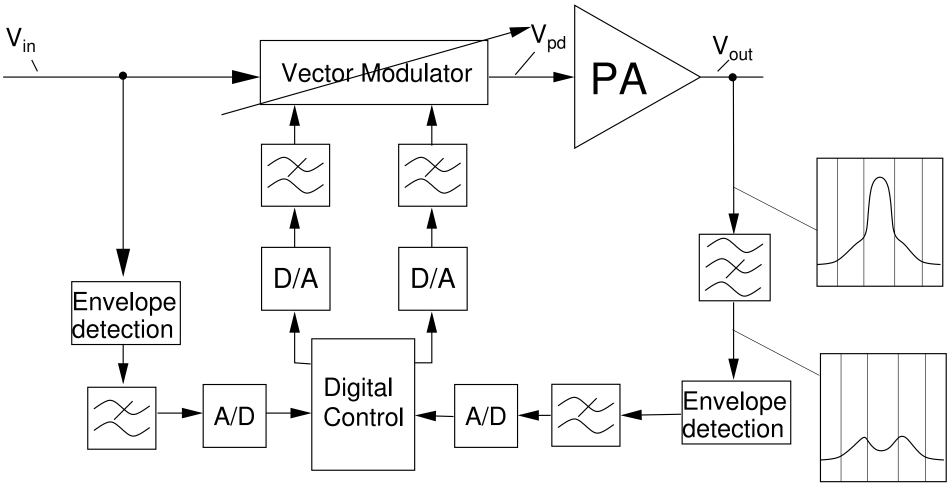


Figure 4.6: Frequency-domain feedback using filtering

The method based on the direct measurement of the nonlinearity uses, for example, a vector analyzer to measure the PA transfer function and then a microprocessor to calculate the inverse transfer function. The inverse transfer function is updated when the PA transfer function changes. The problem with this method is the complexity of the circuitry that is required for measuring the transfer function, and the calculation of the transfer function which may become complicated. Another problem is that there is no data available of the efficacy of the predistortion, thus the function can not be adjusted to optimally reduce the distortion.

4.5 Other implementation issues in RF-predistortion system

As the RF predistorter is based on an analog predistortion element, the analog parts have a significant effect on the operation of the predistorter.

The generation of the digital control signals requires the envelope information of the RF signal. The envelope detection is not an ideal operation, so instead one has to use approximate methods, such as diode detectors or logarithmic amplifiers accompanied by envelope filters. These detectors have different effects on the predistorter, depending on the signal and PA type. The effects of the detectors and filters are discussed in Chapter 6.

Another parameter affecting the operation of the predistorter is the implementation of the D/A and A/D converters, including the DSP word lengths, clock frequencies and the reconstruction and

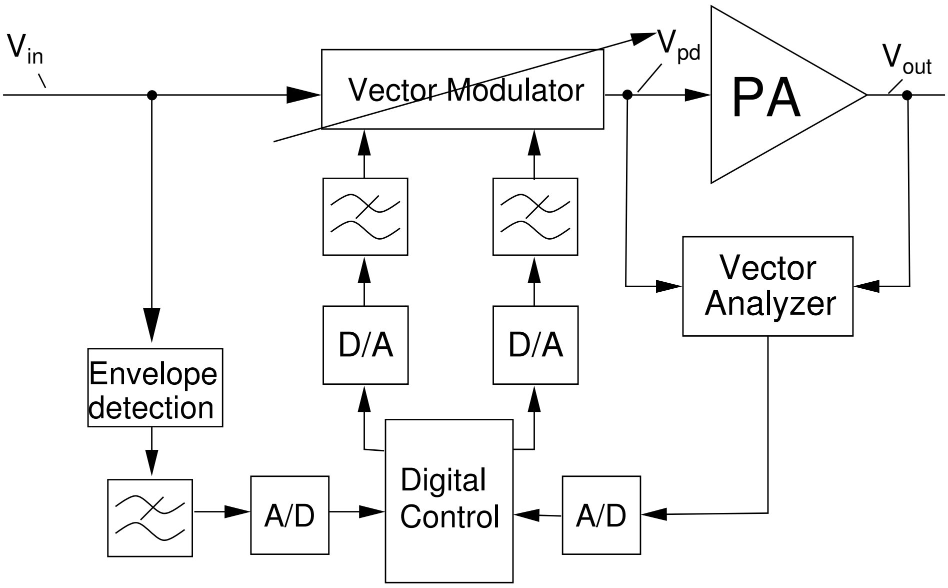


Figure 4.7: Frequency-domain feedback using direct PA nonlinearity measurement

anti-aliasing filters. The word lengths of the converters affect the noise floor and the maximum linearization ability of the predistorter. This will be discussed in Chapter 9. The clock frequency and the filter bandwidths affect the maximum signal bandwidth the predistorter is able to cope with and the maximum nonlinearity of the PA is allowed to be linearizable.

The filters required in the implementation do not have constant frequency responses over the whole signal band and their group delays are nonzero. This causes memory effects in the predistortion system. The non-constant frequency responses of the filters have been discussed in references [139–141].

The delays have a significant effect on the operation of an RF predistorter. This is due to the fact that the delays are generated to the baseband signals and, compared to the carrier frequency, the delays are very large. This means that the compensation requires fairly bulky or special methods. The most significant, and hardest to compensate, effect is the delay caused on the control signals of the analog predistortion element, which causes time offset between the predistortion function and the RF signal. Another effect of the delay is the time mismatch between the predistorter input and PA output signals which affects the update of the predistortion function. The effect of the delays will be discussed in more detail in Chapter 5.

4.6 Conclusions

This chapter discussed several commonly used RF-predistorter types from fully analog predistorters through digitally controlled analog predistorters to fully digital predistorters. The focus was on the digitally controlled analog predistorters and the benefits and problems of an digitally controlled analog predistorter were discussed. In addition, the possibility of implementing a stand-alone linearized PA chip using digitally controlled RF-predistortion was considered.

This thesis will mainly concentrate on RF predistorters using phase and amplitude modulators, time-domain feedback and digital piecewise constant approximation of the predistortion function. However, many of the results are applicable to other RF predistorter types. The following chapters discuss more thoroughly some of the RF predistorter design issues.

Chapter 5

The effect of delays on an RF predistorter

5.1 Introduction

As discussed in previous chapters, often the PAs are assumed to be memoryless. However, the RF-predistortion system has an intrinsic delay difference between the analog predistorter control signals and the RF signal. This means that there is phase difference between the predistortion function and the RF envelope and that the PA output signal has residual error that is dependent on the previous signal values, not directly on the PA distortion. This can be seen to be a memory effect of the predistorter.

One significant problem in developing a stand-alone RF-predistortion chip or a PA with integrated RF-predistortion is its sensitivity to the predistorter control signal delays. The control signal delay mismatch is generated by the filtering required in the detection of the input signal envelope as well as in the reconstruction of the digital signal to analog form and the processing delay in the data converters and the DSP. The delays are usually of the same order of magnitude as the DSP clock period, thus being usually much larger than the carrier period. This means that simple filtering can not generate large enough delays to the RF signal. Several methods for delay compensation have been suggested. All of them require bulky analog delay elements, such as delay lines or down conversion. These kinds of compensation methods are either not integrable on a single chip or require accurate information of the carrier frequency and thus hinder the development of a universal predistorter chip. In this chapter, a novel delay compensation method for the control signal delays will be presented. This method uses a polynomial predictive filter that is completely digital and can be integrated to the digital control of the predistorter, without requirement to external components. The previously presented compensation methods and the new method will be discussed in more detail in Sections 5.2 and 5.4.

If the predistorter function is updated adaptively using time-domain signals, the delay difference between the original and the fed back signals used for comparison affects the predistortion adversely as well. This mismatch causes that part of the desired signal not to be canceled during the error calculation, which further causes a time-dependent misadjustment to the predistortion function that can even make the adaptation unstable. This effect has been studied more than the control signal delay, since it affects the baseband predistorters also. This will also be briefly discussed in this chapter.

Figure 5.1 shows the main delay sources in an RF-predistortion system. The RF delay elements usually have at least an order of magnitude less delay than the baseband delay elements. This is due to the fact that the clock frequency, which defines the maximum corner frequency of the reconstruction filters and further the group delay of the baseband filters, is much lower than the carrier frequency. This restricts the maximum group delay in the reconstruction filters. The baseband delay elements are marked with lighter circles and the RF-delay elements are marked with the darker circle. The RF delay affects only the predistortion update and adds to the delay mismatch in the baseband feedback and feedforward paths.

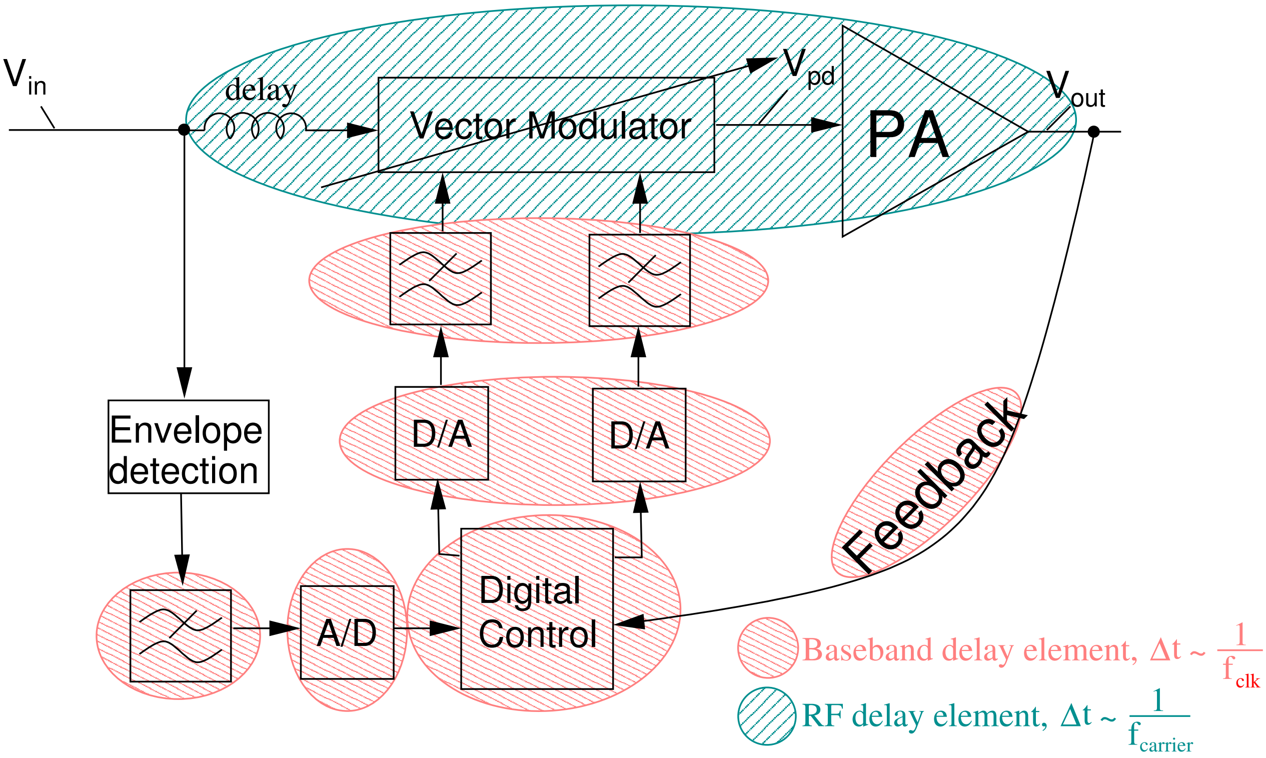

V_{in}
delay
Vector Modulator
V_{pd}
PA
V_{out}
Envelope detection
D/A
D/A
A/D
Digital Control
Feedback
Baseband delay element, $\Delta t \sim \frac{1}{f_{clk}}$
RF delay element, $\Delta t \sim \frac{1}{f_{carrier}}$

Figure 5.1: Main delay sources in an RF-predistortion system

5.2 The delayed control signals on an RF-predistortion system

When there is delay in the predistorter control compared to the RF signal, wrong signal values are altered at the predistorter. Figure 5.2 shows the effect of control signal delay on a highly nonlinear third order amplifier in time-domain. The upper part of the figure shows the original and distorted signals and the linearized signals with and without delays. The lower part shows the original and delayed predistortion functions. As can be seen, the delay shifts the peaks of the sine and leaves the resulting signal clearly distorted.

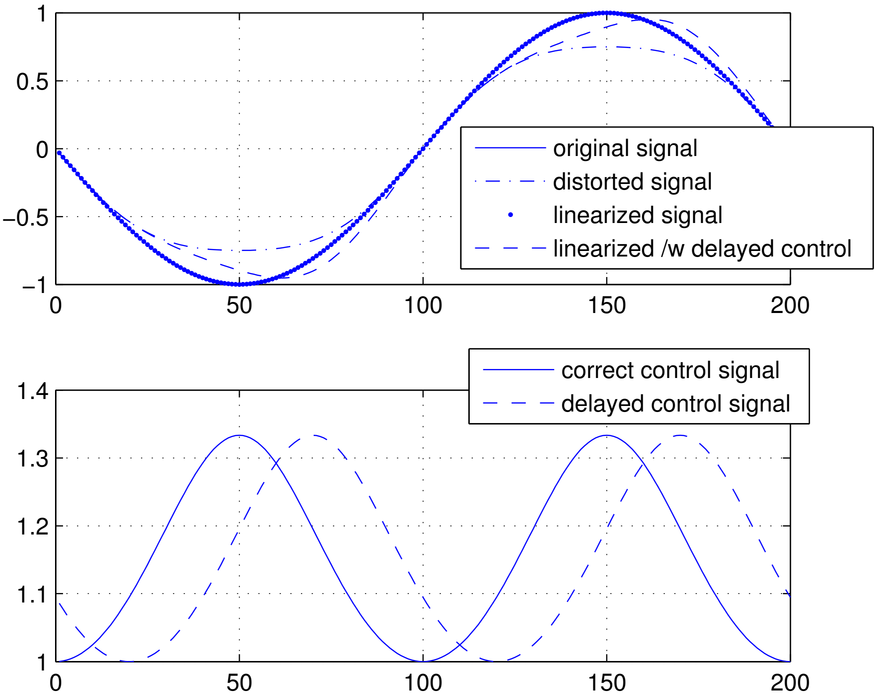


Figure 5.2: The effect of delayed predistortion function on a predistorted sine signal driven through a third-order polynomial distortion.

An important factor that defines the error caused by the delay is how rapidly the signal changes, as this defines how much the signal value to be adjusted has deviated from its optimal value before linearization is applied. It is important to notice that, not only the derivative of the complex envelope affects the error, but also the derivative of the absolute value of the envelope. The most important thing affecting the speed of the change is the bandwidth of the signal; thus wider band signals tolerate less delay than narrow band signals. However, it should be noted that the speed of the change depends also on the signal type. The envelope period will be used as the metric for the changing speed of the signal in this thesis. This is defined as follows:

$$t_{env} = \frac{1}{f_{env}} \tag{5.1}$$

to be the inverse of the signal bandwidth and thus the period of the highest frequency component of the signal.

What is important to note is that the signal derivatives also may have correlation with the signal real amplitude. For example, in the 16QAM signal envelope, the derivative tends to be larger at the low amplitudes than at high amplitudes, and thus the delay affects the lowest amplitude values the most.

Figure 5.3 illustrates how the derivatives of a 16QAM signal are distributed as a function of the amplitude. As can be seen the average derivative is largest at the low amplitudes and decreases towards the higher amplitudes and thus the error increases. However, it should be noted that the derivative of the predistortion function at the current amplitude affects the instantaneous error caused by the delay as well. If the value of the predistortion function does not change significantly during the delay error period, the error in the linearization remains small also. Thus if the PA to be linearized exhibits nonlinearity at the low amplitudes, the error is concentrated at the lower amplitudes also. If, on the other hand, the PA is linear at the low amplitudes, the error diminishes.

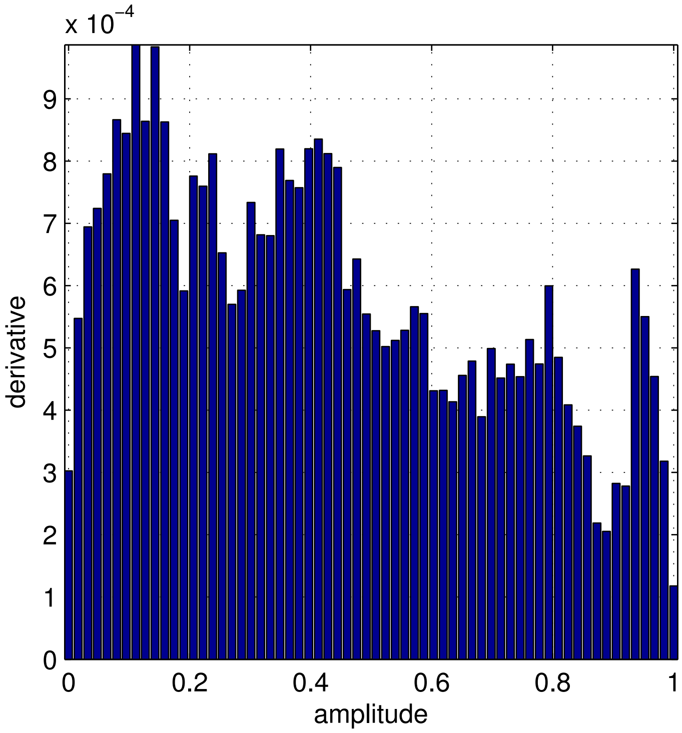


Figure 5.3: The distribution of the derivative of a 16QAM signal versus the signal amplitude

The delay can be seen as phase rotation of the spectral components of the predistortion function which causes the distortion to be incompletely canceled or even increased. The relation between the time and frequency-domain signals and their delayed versions is

$$\begin{array}{ccc} v_{out}(t) = A_{PA}(t)A_{PD}(t)v_{in}(t) = v_{in}(t) & \leftrightarrow & V_{out}(f) = A_{PA}(f) \otimes A_{PD}(f) \otimes V_{in}(f) \\ \downarrow delay & & \downarrow delay \\ v_{out}(t) = A_{PA}(t)A_{PD}(t-T)v_{in}(t) \neq v_{in}(t) & \leftrightarrow & V_{out}(f) = A_{PA}(f) \otimes (A_{PD}(f)e^{-j2\pi fT}) \otimes V_{in}(f) \end{array} \tag{5.2}$$

As can be seen, the frequency-domain output signal becomes dependent on the bandwidth of the signal as the error caused by the time difference depends on the location of the signal on the frequency axis. The effects can be seen to be a basic form of memory effects. However, there is an important difference compared to the more complex memory effects: as the control signal delay does not cause the distortion to spread over time, it can be compensated with adjusting the delay of the RF or baseband signals without the requirement for complicated inverse filtering.

The facts concerning the distribution of the delay error according to the amplitude and predistortion function can be also concluded from simulations. Figure 5.4 shows the ACP for a 16QAM signal with full-scale amplitude and half amplitude and for a full-scale DQPSK signal. The simulations were performed with a static PA with a 256-entry LUT for storing the predistortion function. PA models PA2 and PA3 were used. What can be seen is that the ACP deteriorates rapidly at small delay levels. The delay matching should therefore be performed as accurately as possible. What is also important to notice is the how the delay increases the ACP of PA2 more rapidly than the ACP of PA3. This is due to the fact that the delay error has the greatest effect for a 16QAM signal at the low amplitudes and, as the PA2 has nonlinearity at the low amplitudes also, the delay error significantly interferes with the predistortion. The DQPSK signal, on the other hand, has less low-amplitude peaks due to the non zero crossing properties of the signal; thus, for the DQPSK signal, the results for PA1 and PA2 differ less from each other.

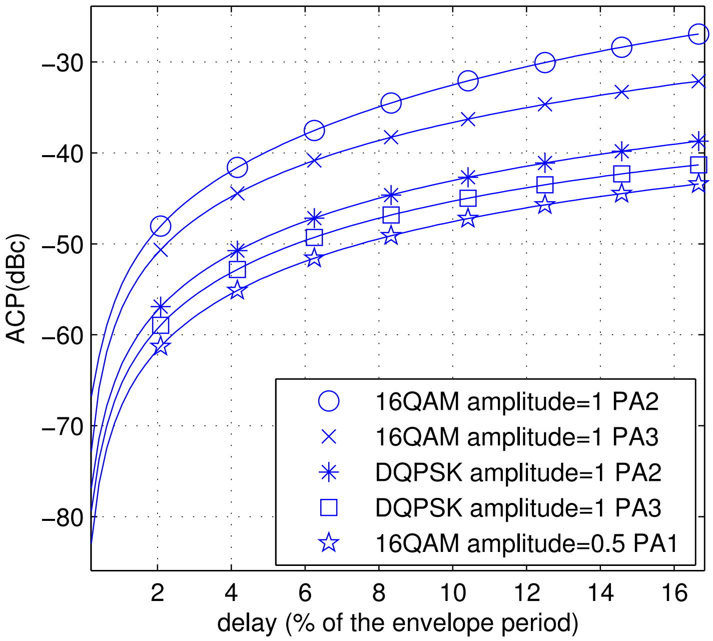


Figure 5.4: The effect of the control signal delay on the ACP of an RF predistorted power amplifier

Figure 5.5 shows the control signal delays for different filter configurations. The filters used are Butterworth filters with different orders and the clock frequency is defined by finding the frequency at which the stop-band attenuation reaches 40dB. The oversampling ratio defines the ratio between the filter corner frequency and the clock frequency. The first plot shows the delay of the control signal as percents of f_{env}. The figure assumes that the filter corner frequency is four times higher than f_{env} due to spectral spreading caused by the envelope detector (Chapter 6).

As can be seen, high order filters and low oversampling ratios cause delays of several envelope periods, which according to Figure 5.4, is clearly unacceptable.

There are several methods to reduce the delay. The delays caused by the envelope, reconstruction and anti-aliasing filters can be minimized; however, this usually means relaxing the requirements for the transition band width and increases the clock frequency. The clock frequency can also be used to directly affect the control signal delay as it reduces the clock cycle time and thus reduces the time consumed by the digital part of the system. However, increasing clock frequency increases the power consumption of the digital part, which is not desirable. What is also notable is that, due to the fact, that as the clock frequency and oversampling ratio increase, the number of clock cycles corresponding to the time delay also increases. This causes the delay measured in clock cycles to have a minimum at the oversampling ratio of four. This is important for predictor circuits, which benefit from minimizing the delay in clock cycles. For these reasons, these remedies cannot be applied infinitely and there will be significant amount of delay left in the system.

Due to the previously mentioned problems, methods have been proposed to reduce the delay difference by delaying the RF signal with the amount the baseband delay. However, the amount of delay a filter can generate has its limits.

The transfer function of a filter can be written as a product of its poles and zeros as

$$\begin{aligned} H(\omega) &= \prod_n \frac{j\omega - z_n}{j\omega - p_n} && (5.3) \\ &= \prod_n (-\Re(z_n) + j(\omega - \Im(z_n)))\frac{\Re(p_n) + j(\omega - \Im(p_n))}{-\Re(p_n)^2 - (\omega - \Im(p_n))^2}. && (5.4) \end{aligned}$$

where z_n are the complex zeros and p_n are the complex poles. The phase transfer function can be calculated with the help of (5.3) as

$$\begin{aligned} \phi(\omega) &= \arctan\frac{\Im(H(\omega))}{\Re(H(\omega))} && (5.5) \\ &= \sum_n -\arctan\frac{(\omega - \Im(z_n))}{\Re(z_n)} + \arctan\frac{(\omega - \Im(p_n))}{\Re(p_n)}. \end{aligned}$$

Further the group delay of the filter can be calculated to be

$$\begin{aligned} \tau &= -\frac{d}{d\omega}\phi(\omega) \\ &= \sum_n \frac{\Re(z_n)}{\Re(z_n)^2 + (\omega - \Im(z_n))^2} - \frac{\Re(p_n)}{\Re(p_n)^2 + (\omega - \Im(p_n))^2}. && (5.6) \end{aligned}$$

We get as the maximum delay

$$\tau_{max} = \sum_n \frac{1}{\Re(z_n)} - \frac{1}{\Re(p_n)}. \quad (5.7)$$

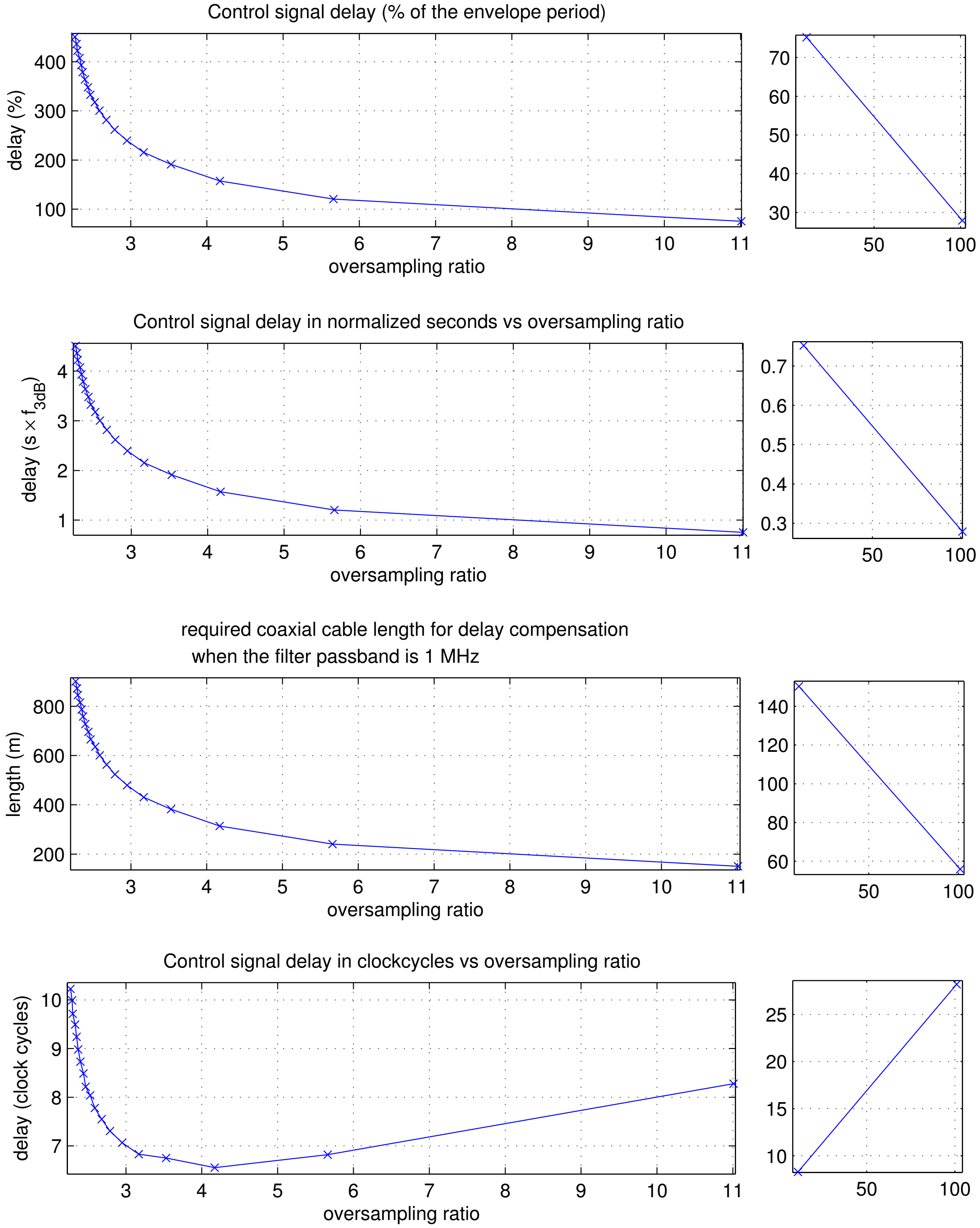


Figure 5.5: Control signal delay with different oversampling ratios

$\Re(p_n)^2$ should be negative for the filter to be stable, and the real part of the zero should be positive to contribute positively to the delay. In principle, any delay can be achieved if the real part of the pole or zero is close enough to zero.

What can be seen from (5.6) is that the imaginary part of the poles and zeros define the frequency of the maximum delay point. Another important fact that can be discovered from the formula is that, the smaller the real part of the zero or pole is, the faster the delay falls from the maximum value. To be able to compensate for the control signal delay effectively, the delay filters should have such a group delay function that the delay does not change within the signal bandwidth more than is required for the target ACP. This poses a limit on the minimum value of the real part of the poles and zeroes and thus on the maximum achievable delay. On the baseband, at least one of the filter poles should be at the same order of magnitude as the sampling frequency of the baseband circuitry for one to be able to implement efficient anti-aliasing and reconstruction filtering. On the RF path the filters have to be implemented using passive components and the carrier frequency is hundreds or even thousands of times larger than the maximum baseband frequency. This requires the zero or pole to have even thousands of times larger imaginary part than the real part to move the delay peak from the DC to the carrier frequency. Therefore, the required component values would be unrealizable with sufficient accuracy using inductors and capacitors [142].

Nevertheless, by using mechanical resonator-based SAW or BAW filters, this kind of RF filters with large delays can be implemented. This kind of delay elements are bulky [143] and have strong reflections [144], which limits their usability. However, if the design already requires a bandpass SAW filter, this same filter can also be used for delay compensation thus no additional components are required. This kind of solution has been used by, for example, Kusunoki et al. [10].

A more commonly used method for delay compensation is a passive coaxial delay line [86, 114, 137]. This is basically a length of cable that has a large-enough propagation delay. Since the propagation speed of electromagnetic waves in a coaxial cable is commonly 65%-90% of light speed [145] which means delays from 3.7 ns/m to 5.1 ns/m. This means that for delays of hundreds of nanoseconds the cable lengths become tens of meters. As can be seen in Figure 5.5 to implement, for example, a 300 ns delay with coaxial cable with a propagation speed of 66% of the speed of light, a cable length of 60 meters is required. If we use a coaxial cable with 2mm radius, this device requires at least 9cm x 9cm x 9cm cube, which is not suitable for portable devices.

There are also similar delay problems in EE&R transmitters due to separate RF and baseband signals. Raab et al. [146] suggest down converting the RF-signal to baseband and then filtering it with a filter with large group delay and then up-converting the signal. The problem with this method is the required up and down conversion of the signals which makes the design more complex and dependent on the carrier frequency and thus significantly limits the advantages of the RF-predistortion over the baseband predistortion.

Finally, Kim et al. [147] propose generating the vector modulator control signals at the baseband in digital domain to form a hybrid of the baseband and RF predistorters. This makes it possible to delay the RF signal digitally, but makes the predistorter dependent on the baseband circuitry, so one of the advantages of the RF-predistorter is lost. However, compared to the baseband predistortion system, the up-converter requires a narrower bandwidth[147].

To conclude, several methods to compensate the delay have been proposed, but they all either require additional large hardware, which makes complete integration of the predistorter hard or even impossible, or require forfeiting some important advantages of an RF predistorter over baseband predistorter. Thus they are best suited for base-station applications.

5.3 The effect of delay mismatches on the feedback path

If the update of the predistortion function is based on subtraction of the PA input and output signals, either at the RF or at the baseband, the delay mismatch between the signals to be compared causes unwanted spectral components to be present in the calculation of the error. This error is due to incomplete suppression of the wanted signal during the subtraction of the PA input and output signals.

The residual signal shows itself as an increased level of in-band distortion that is not level dependent but, instead, depends on the previous signal values. Thus it resembles memory effects in the PA. The approximate power of the unwanted signal can be calculated as follows. We assume that the PA input signal is $v_{in}(t)$ and the output signal is $v_{out}(t)$. It is also assumed that the output signal can be constructed from the input signal by adding an error signal, $v_{err}(t)$. As the most important factors are the spectral components of the signal, it is feasible to perform the analysis in frequency-domain. The Fourier-transformed versions of the signals are $V_{out}(\omega)$, $V_{in}(\omega)$ and $V_{err}(\omega)$, respectively. Thus,

$$V_{out}(\omega) = V_{in}(\omega) + V_{err}(\omega). \tag{5.8}$$

Now we can calculate the error:

$$E(\omega) = V_{out}(\omega) - V_{in}(\omega) = (V_{in} + V_{err})(\omega) - V_{in}(\omega) = V_{err}(\omega) \tag{5.9}$$

When the output is delayed by T, (5.9) is transformed into the form

$$\begin{aligned} E_{delay}(\omega) &= V_{out}(\omega)e^{-Tj\omega} - V_{in}(\omega) \\ &= (V_{in}(\omega) + V_{err}(\omega))e^{-Tj\omega} - V_{in}(\omega) \\ &= V_{in}(\omega)(e^{-Tj\omega} - 1) + V_{err}(\omega)e^{-Tj\omega} \\ &= V_{in}(\omega)(1 - \cos(T\omega) + j\sin(T\omega)) + V_{err}(\omega)e^{-Tj\omega}, \end{aligned} \tag{5.10}$$

where t_{rf} is the carrier period. What we can see is that the delay causes the input signal to be multiplied with a rotating phasor. $V_{err}(\omega)e^{-Tj\omega}$ represents the delayed amplitude-dependent error signal and thus the unwanted residual error can be written in the form

$$E_{res}(\omega) = V_{in}(\omega)(1 - \cos(T\omega) + j\sin(T\omega)) \tag{5.11}$$

We can estimate the effect of the delay on the suppression of the signal band by calculating the power spectrum of the residual error respective to the delay.

$$|H_{res}(\omega)|^2 = \left|\frac{E_{res}(\omega)}{V_{in}(\omega)}\right|^2 = \left((1 - \cos(T\omega))^2 + (\sin(T\omega))^2\right) = 2 - 2\cos(T\omega) \tag{5.12}$$

If we consider the case where the subtraction is performed on the RF using analog components, then ωin (5.12) can be replaced with $2\pi f_{rf}$, where f_{rf} is the carrier frequency and f_{rf} can be further be replaced with the inverse of the carrier period, t_{rf}. Now if we assume the signal to be fairly narrow band compared to the carrier frequency, we can approximate the attenuation of the data signal component after the comparison to be the value of (5.12) at single point corresponding to the delay T :

$$A_{res}(T) = 10\log\left|H_{res}\left(\frac{2\pi}{t_{rf}}\right)\right|^2 = 10\log\left(2 - 2\cos\left(\frac{2\pi T}{t_{rf}}\right)\right) dB \tag{5.13}$$

The power of the residual signal compared to the distortion power is plotted in Figure 5.6.

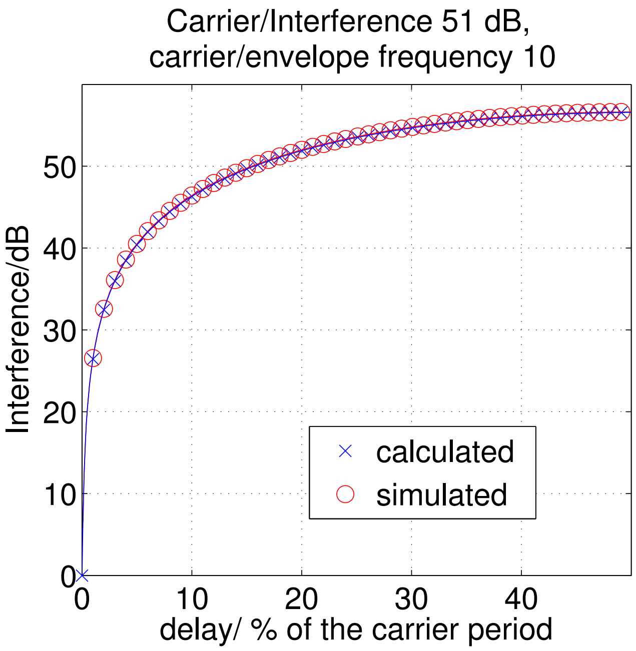


Figure 5.6: The effect of delay mismatch on comparison of input and output signals on RF

The effect of delay on the comparison done on the baseband is more complicated as the signals usually span a large number of frequencies from DC up to the envelope frequency, f_{env} and the $|H_{res}(\omega)|^2$ changes very rapidly near DC. Therefore, for signals with continuous spectrum, the effect should be calculated by integrating (5.12) from $-f_{env}$ to f_{env}. This gives us

$$P_{res}(T) = 10\log\left(2 - \frac{\sin\left(\frac{2\pi T}{t_{rf}}\right)}{\pi\frac{T}{t_{rf}}}\right). \tag{5.14}$$

The calculated results of (5.14) and (5.12) are plotted in Figure 5.7. For comparison, simulation results for delayed 16QAM, two-tone and 100-tone signals are plotted in the figure. It can be seen that, as the tones get denser, the results follow Equation (5.14) more closely and the spectrally continuous 16QAM signal follows the calculations almost exactly. A detailed analysis of the effect of feedback delay on the update of baseband predistorters can be found in references [79, 115].

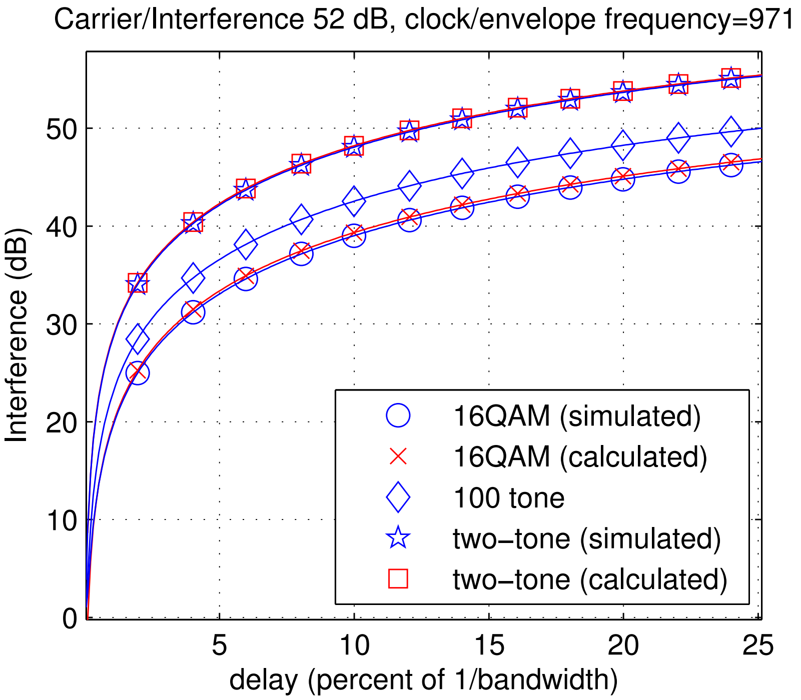


Figure 5.7: The effect of delay mismatch on comparison of input and output signals on baseband

5.4 Predictor

Instead of delaying the RF signal, the control signal delay can be compensated by expediting the control signals. This can be done by predicting of the signal used to index the control signal LUT. The following sections examine the implementation of this kind of prediction circuit using polynomial functions and adaptive filtering and their applicability to RF-predistortion.

5.4.1 Polynomial predictor

One of the simplest ways (from theoretical point of view) to predict the future value of a signal is to fit a function to the past values of the signal and use that function to calculate the estimates for the signal. The complexity of the estimation is defined by the function used. It should be noted that the complexity of the estimation function does not correlate with the goodness of the estimation as the future signal values may differ considerably from the function even though the fit of the previous values would be perfect.

Polynomials are very simple and flexible functions that can be used to predict the future values of the signal. A significant advantage of polynomial prediction functions is that the polynomial coefficients and the next value can be calculated simultaneously with a constant coefficient FIR or IIR filter [148]. The bandwidth of the filter depends on the order of the polynomial [148]. The higher the order of the polynomial, the higher bandwidth signals the filter is able to predict, but the hardware requirements also increase by one filter coefficient for each additional polynomial order. What should be noted is that a basic polynomial predictive filter is a high pass filter with gain on the high frequencies and unity gain near DC. This means that the predictor is sensitive to out-of-band noise and, as will be seen later in this section, as the polynomial order increases, the sensitivity to noise increases also.

The simplest (first-order) polynomial predictor can be constructed by linear extrapolation or by calculating the slope of the line defined by the two most recent signal values and by continuing into the direction of this line for the number of clock cycles to be predicted [16]. This can be written as

$$v(n+N) \approx v(n) + N\left(v(n) - v(n-1)\right) = (N+1)\,v(n) - Nv(n-1) \tag{5.15}$$

where N is the number of clock cycles to be predicted and $v(n)$ is the signal value at time instant n. Figure 5.8 shows the frequency response for a first order polynomial predictor with one clock cycle prediction ability. The high-pass transfer function of the predictor is clearly visible and the out-of-band gain is over 9 dB at the maximum, corresponding to decreasing the word length by 1.5 bits (see Equation (3.10)). Thus this predictor is suitable for prediction of LUT address lengths of at least two bits less than the input word length. Also evident in the figure is the fact that the phase difference is linear, and thus the group delay is constant up to $0.05\,f_s$.

The filter coefficients for a higher order predictor can be solved by fitting the past signal values to a polynomial instead of a line [16]. The future values are then calculated by inserting the desired number of clock cycles to this polynomial and calculating the predicted value. Using this formulation one can derive the following formulas for second and third order polynomials:

$$v(n+N) \approx \left(\frac{1}{2}N^2 + \frac{3}{2}N + 1\right) v(n) - (2N - N^2)v(n-1) + \frac{N+N^2}{2}v(n-2) \tag{5.16}$$

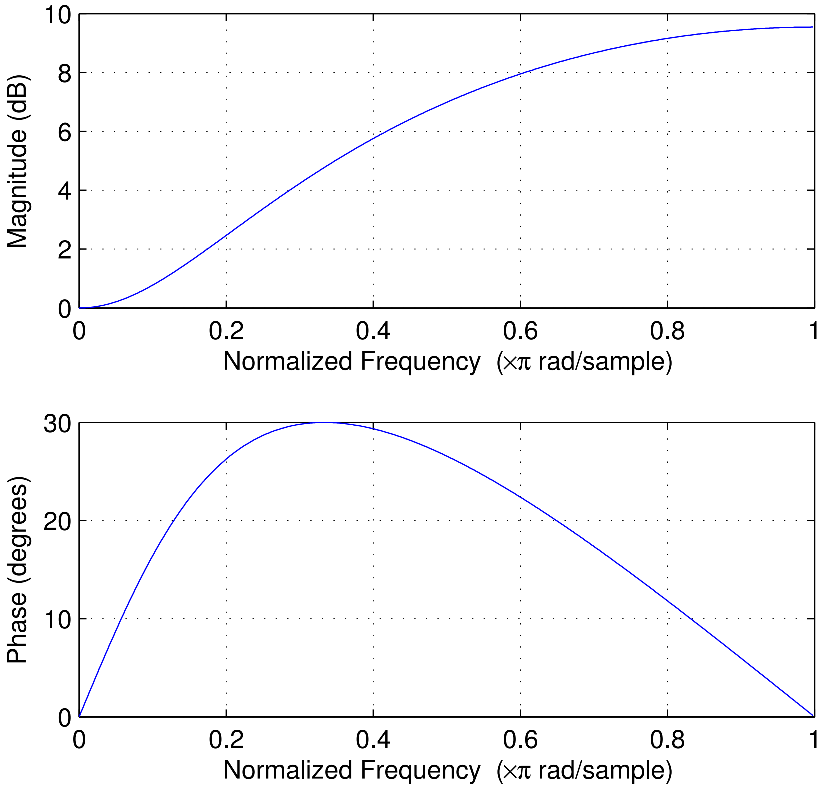


Figure 5.8: Frequency response of a first order polynomial predictor

for a second order predictor and

$$\begin{aligned} v(n+N) \approx{} & \\ & \left(10 - -\frac{47}{6}N + 2N^2 - \frac{1}{6}N^3\right) v(n) + (-20 + 19N - \frac{11}{2}N^2 + \frac{1}{2}N^3)v(n-1) \\ & + (15 - \frac{31}{2}N + 5N^2 - \frac{1}{2}N^3)v(n-2) + (-4 + \frac{13}{3}N - \frac{3}{2}N^2 + \frac{1}{6}N^3)v(n-3) \end{aligned} \tag{5.17}$$

for a third order predictor.

Figure 5.9 shows the operation of the first, second and third order polynomial predictors. It can be seen that, when the signal has strong curvature, the higher order polynomials give better prediction results, as these rapid changes in the signal have wide instantaneous bandwidth. On the other hand, the more closely the signal approximates a straight line, the more strongly the high-pass-type frequency response of the higher order predictors starts to affect the prediction adversely. The higher order predictors amplify the frequency components generated by the small fluctuation in the signal and they start to deviate significantly from the actual signal value when the prediction period is long. The first order predictor filters out the small fluctuation and follows the general direction of the signal. It should be noted that, with short time prediction, the high-order polynomials give, in all the cases depicted in the figure, better results than lower order predictors.

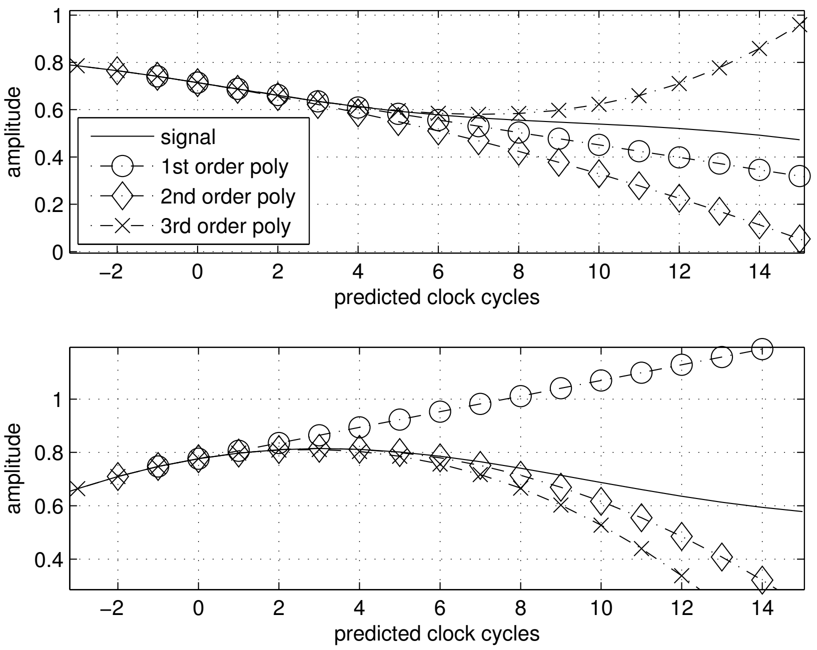


Figure 5.9: The prediction ability of polynomial predictors

If the control signal delays are compensated using analog delay elements, then there is a direct trade-off between the size of the analog delay element and the signal oversampling ratio (i.e. the clock frequency). In the case of a digital predictor, the optimization cannot be achieved purely by minimizing the time delay. The reason for this is that the prediction error of a polynomial predictor caused by noisy measurements is proportional to the prediction length.

The number of clock cycles predicted should be minimized, as this affects directly the high frequency gain of the predictive filter, as can be seen by plotting the frequency responses of (5.19), (5.21) and (5.23). According to Figure 5.5, this means that we cannot minimize the delay in the DSP part of an RF predistorter by infinitely increasing the clock frequency. Actually, according to the figure, the minimal number of predicted clock cycles is achieved with a fairly low OSR.

Also it should be noted that the time delay does not directly affect the hardware costs of the predictor. Instead, the order of the polynomial and the required oversampling ratio define the hardware costs. In fact, as the oversampling ratio defines the clock frequency of the whole predistorter, it significantly affects the total hardware cost of the predistorter.

The effect of the oversampling ratio was examined with numerical calculations. Butterworth filters with order varied 1 to 20 were selected to be the anti-aliasing and reconstruction filters in the calculations. The signal over-sampling rate was defined as $OSR_{sig} = \frac{f_{clk}/2}{f_{max,sig}}$. A quantity called the filter oversampling ratio was calculated by dividing the frequency where the stop band attenuation had reached 40 dB (this was chosen to be the clock frequency) by the filter corner frequency. The filter oversampling ratio illustrates the relation of the filter bandwidth to the clock frequency. The control signal delay was calculated by multiplying the filter group delay by two (to take into account anti-aliasing and reconstruction filters) and adding three clock cycles (one clock cycle each for the A/D converter, D/A converter and DSP). The signal bandwidth was defined to be the maximum bandwidth at which the difference between the group delay of the predictor and the delay of the predistortion system was less than 0.5 $\%$ of the envelope period, which, according to Figure 5.5, should cause a negligible ACP loss. Figure 5.10 presents the calculated required signal oversampling ratios, minimum control signal delays as percentage of t_{env}, the improvement in ACP of a 16QAM signal amplified with PA2, caused by the predictor (it should be noted that the actual resulting ACP is the same for all cases) and the increase of noise floor caused by the predictor's high frequency amplification transformed into reduction of bit accuracy calculated with (3.9), as a function of the filter oversampling ratio. The increase in noise floor was calculated by integrating the noise floor power out-of-signal band before and after prediction. The results for oversampling ratios from 1 to 11 are in a scale different from that of the results for oversampling ratios from 11 to 100.

What can be seen is that the minimum of the signal oversampling ratio is achieved with a filter oversampling ratio of little over 4 which corresponds to a fourth order Butterworth filter. Also it can be seen that the percentile delay stays fairly constant. Furthermore, as can be seen from the figure, this actually means that the ACP improvement caused by the prediction is almost the same regardless of the filtering if the signal bandwidth is the same as the predictor bandwidth. Finally, it can be seen that the decrease of accuracy due to the high pass filtering also stays constant regardless of the filter oversampling ratio, but it should be noted that the SNR improves as the oversampling ratio increases, so the accuracy actually improves somewhat as the OSR increases. Although the results were achieved with a delay of a set number of clock cycles in the digital part of the predistorter, it can be verified by simulations, that the filter OSR defines the shape of the curves in Figure 5.10. Thus, the minimum stays at the same location and only the value of the minimum changes.

When the results are compared to Figure 5.5, it can be seen that all the minima are located at the filter oversampling ratio at which the delay in clock cycles is at minimum.

Figure 5.11 shows the maximum prediction ability of first to third degree polynomial predictors. The figure shows the maximum compensable delay from figure 5.10 transformed into clock cycles as a function of the signal OSR in the same figure. As can be seen, there is a linear relationship between the OSR and the prediction ability. By fitting a line to the results, we get as the maximum prediction abilities

$$\begin{aligned} N_{max(1^{st}\ order)} &= 0.10OSR - 0.47 \\ N_{max(2^{nd}\ order)} &= 0.13OSR - 0.86 \\ N_{max(3^{rd}\ order)} &= 0.23OSR - 1.30 \end{aligned} \tag{5.18}$$

It can be seen how the prediction ability of the third order predictor is over twice as good as a first order predictor at the same OSR. The increase in noise power is less straightforward but, as can be seen, the number of predicted clock cycles should be kept as low as possible and that the third

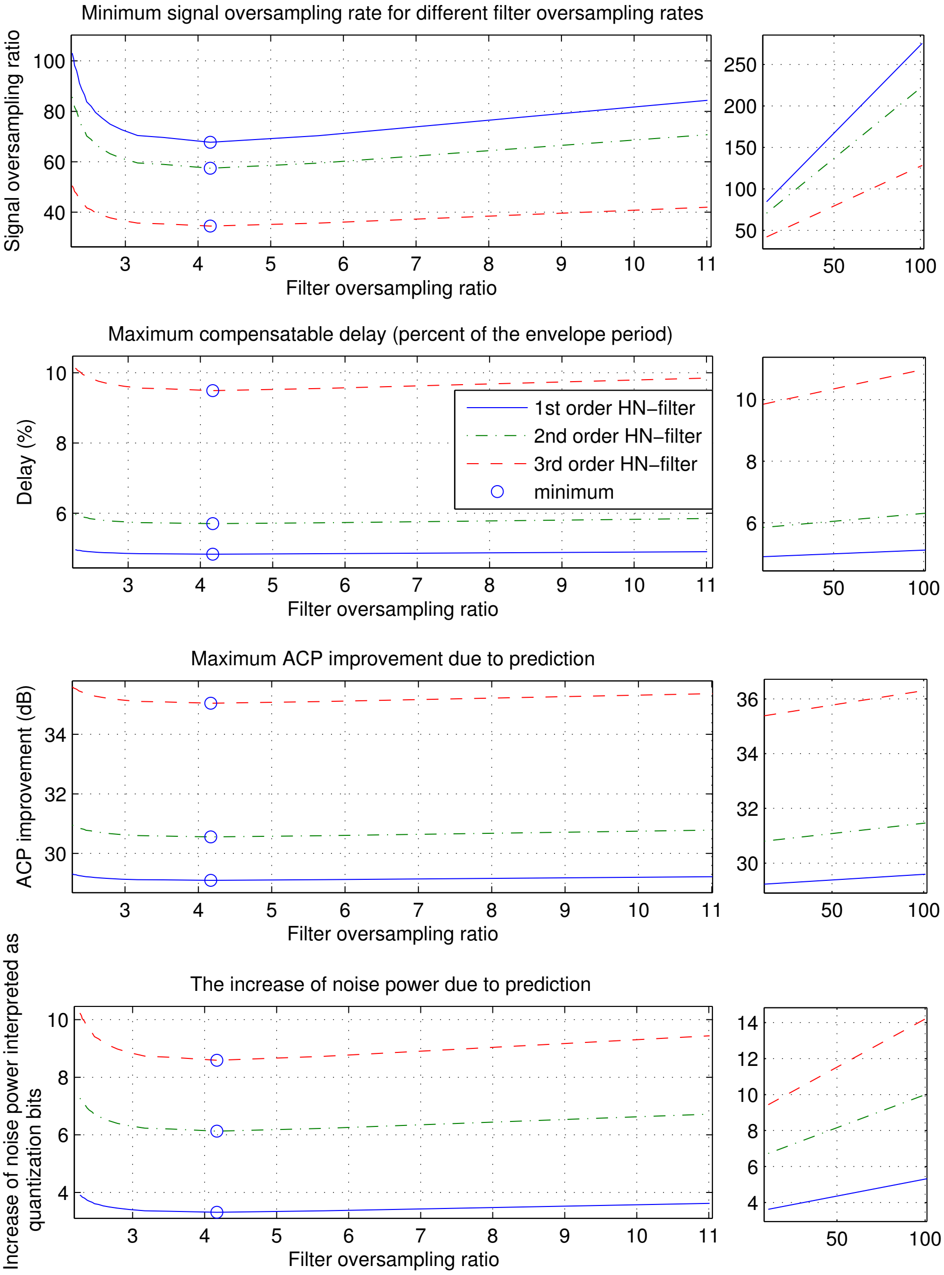


Figure 5.10: The required signal oversampling ratios, control signal delays and for different filter bandwidths

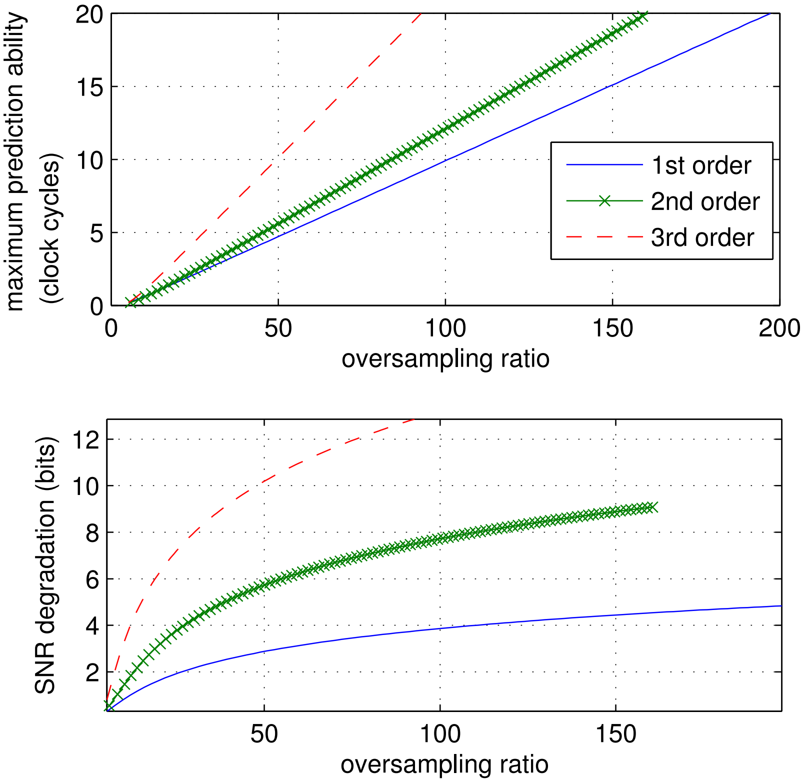


Figure 5.11: The relation of the oversampling ratio and the maximum prediction ability of a polynomial predictor.

order predictor is significantly more sensitive to noise than the first order predictor.

The high oversampling ratios required by the first order polynomial predictor may be a problem in some solutions. The OSR can be reduced by using a higher order polynomial, but this also increases the noise gain, as can be seen from Figure 5.11. Heinonen et al. [16] propose introducing noise suppression to the predictor by implementing the filter with excessive order and using the additional coefficients for averaging. This predictor is called a Heinonen-Neuvo (H-N) filter. The principle of the H-N filter is that the noise suppressing predictor filters the previous samples in order to reduce the noise and uses the filtered samples for the polynomial fitting. The design formulas for the first-to-third order filters are [16]

$$H_{1st}(z) = \sum_{i=1}^{N}(c_0 + c_1 i)z^{-i}, \tag{5.19}$$

where

$$\begin{aligned} c_0 &= \tfrac{2(3D+2N-2)}{N(N-1)} \\ c_1 &= -\tfrac{6(2D+N-1)}{(N+1)N(N-1)} \end{aligned}, \tag{5.20}$$

$$H_{2nd}(z) = \sum_{i=1}^{N}(c_0 + c_1 i + c_2 i^2)z^{-i}, \tag{5.21}$$

where

$$\begin{aligned} c_0 &= \tfrac{3(10D^2+(-14+12N)D+6-9N+N^2)}{N(N-1)(N-2)} \\ c_1 &= -\tfrac{6(30(N+1)D^2+(32N^2-38)D+6N^3-11N^2-9N+14)}{(N+2)(N+1)N(N-1)(N-2)} \\ c_2 &= \tfrac{30(6D^2+6(N-1)D+N^2-3N+2)}{(N+2)(N+1)N(N-1)(N-2)} \end{aligned}. \tag{5.22}$$

and

$$H_{3rd}(z) = \sum_{i=1}^{N}(c_0 + c_1 i + c_2 i^2 + c_3 i^3)z^{-i}, \tag{5.23}$$

where

$$\begin{aligned} c_0 &= \tfrac{-140\,\mathrm{D}^3+(240\,N+540)\,\mathrm{D}^2+(-600\,N-120\,N^2-760)\,\mathrm{D}+16\,(N+4)\,(N+3)\,(N+2)}{(N-3)\,(N-2)\,(N-1)\,N} \\ c_1 &= \tfrac{1680\,N^2+3080+4200\,N)\,\mathrm{D}^3-60\,(N+3)\,(45\,N^2+99\,N+68)\,\mathrm{D}^2}{N\,(N-1)\,(N-2)\,(N-3)\,(N+3)\,(N+2)\,(N+1)}+ \\ &\quad \tfrac{(1200\,N^4+17440+37200\,N+10800\,N^3+31440\,N^2)\,\mathrm{D}-40\,(N+3)\,(N+2)\,(3\,N^3+36\,N^2+58\,N+37))}{(N\,(N-1)\,(N-2)\,(N-3)\,(N+3)\,(N+2)\,(N+1))} \\ c_2 &= \tfrac{(-4200\,N-4200)\,\mathrm{D}^3+720\,(N+3)\,(9\,N+8)\,\mathrm{D}^2}{N\,(N-1)\,(N-2)\,(N-3)\,(N+3)\,(N+2)\,(N+1)}+ \\ &\quad \tfrac{(-2700\,N^3-47100\,N-21960\,N^2-24960)\,\mathrm{D}+60\,(N+3)\,(N+2)\,(4\,N^2+51\,N+35)}{N\,(N-1)\,(N-2)\,(N-3)\,(N+3)\,(N+2)\,(N+1)} \\ c_3 &= \tfrac{2800\,\mathrm{D}^3+(-4200\,N-12600)\,\mathrm{D}^2+(19880+1680\,N^2+12600\,N)\,\mathrm{D}-140\,(N+13)\,(N+3)\,(N+2)}{N\,(N-1)\,(N-2)\,(N-3)\,(N+3)\,(N+2)\,(N+1)} \end{aligned} \tag{5.24}$$

respectively. D is the number of predicted clock cycles and N is the order of the filter. The basic polynomial filters are a special case of H-N filters with N=polynomial order+1.

The filtering reduces the prediction ability and thus increases the OSR, but, by properly selecting the filtering, the noise can be reduced for the higher order polynomials close to the level of the basic first order polynomial predictor, while keeping the OSR at a lower level . The disadvantage of this is the increased hardware cost as the number of filter coefficients increase. However, Campbell et al. [148] present implementations with constant hardware consumption for first and second order predictors. The implementations require three multipliers and five adders and five multipliers and twelve adders, respectively. The third order predictor can be implemented similarly. According to Smith [149], the silicon area consumption of these constant hardware implementations is profitable compared to the standard filter implementation when the filter order of the first order predictor is higher than 5 and the filter order of the second-order predictor is higher than 6. Although the constant hardware implementations use parallel IIR filters to implement the FIR, Campbell et al. [148] state that the resulting filter is always stable as long as there are no rounding errors.

Figure 5.12 presents the required signal oversampling ratios, $OSR_{sig} = \frac{f_{clk}/2}{f_{max,sig}}$, minimum control signal delays as percentage of t_{env}, the improvement in ACP due to the predictor of a 16QAM signal amplified with PA2 and the increase of noise floor power caused by the predictor's high frequency amplification transformed into reduction of bit accuracy calculated with (3.9) as a function of the filter oversampling ratio when the predictor is an H-N filter with four additional filter coefficients or N=polynomial order+5. When compared to the results in Figure 5.10, the accuracy impairment of the filtered second- and third-order H-N predictors are improved below the unfiltered first- and second-order polynomial filters, respectively, but the required OSR of the second- and third-order filters remain significantly below the first- and second-order polynomial filters, respectively.

5.4.2 Adaptive prediction

In addition to the constant coefficient polynomial prediction filters, digital prediction can be implemented using adaptive filtering. The adaptive digital prediction algorithms are based on measuring the filter output that is delayed by the number of clock cycles that we want to predict and comparing it to the current input signal value. The filter coefficients are adjusted to minimize the error between these (Figure 5.13). Common adaptation algorithms are recursive least squares (RLS) and least mean squares (LMS) algorithms, from which the LMS is simpler but the RLS has better convergence properties.

The LMS predictor updates the filter coefficients constantly to minimize the mean square error between the predicted and actual signal using an instantaneous estimate of the gradient vector. The LMS algorithm can be written as [125]

$$\begin{aligned} x_{out}(n) = &\ \overline{W}_{pr}^{T}(n)\overline{X}_{in}(n) \\ e = &\ x_{in}(n) - x_{out}(n-D) \qquad , \\ \overline{W}_{pr}(n) = &\ \overline{W}_{pr}(n) + \mu e \overline{X}_{in}(n-D) \end{aligned} \tag{5.25}$$

where $x_{out}(n)$ is the predicted signal at time instant n, $\overline{W}_{pr}(n)$ is a vector of length N containing the filter coefficients, $\overline{X}_{in}(n)$ is a vector containing N most recent values of x_{in} at time n and

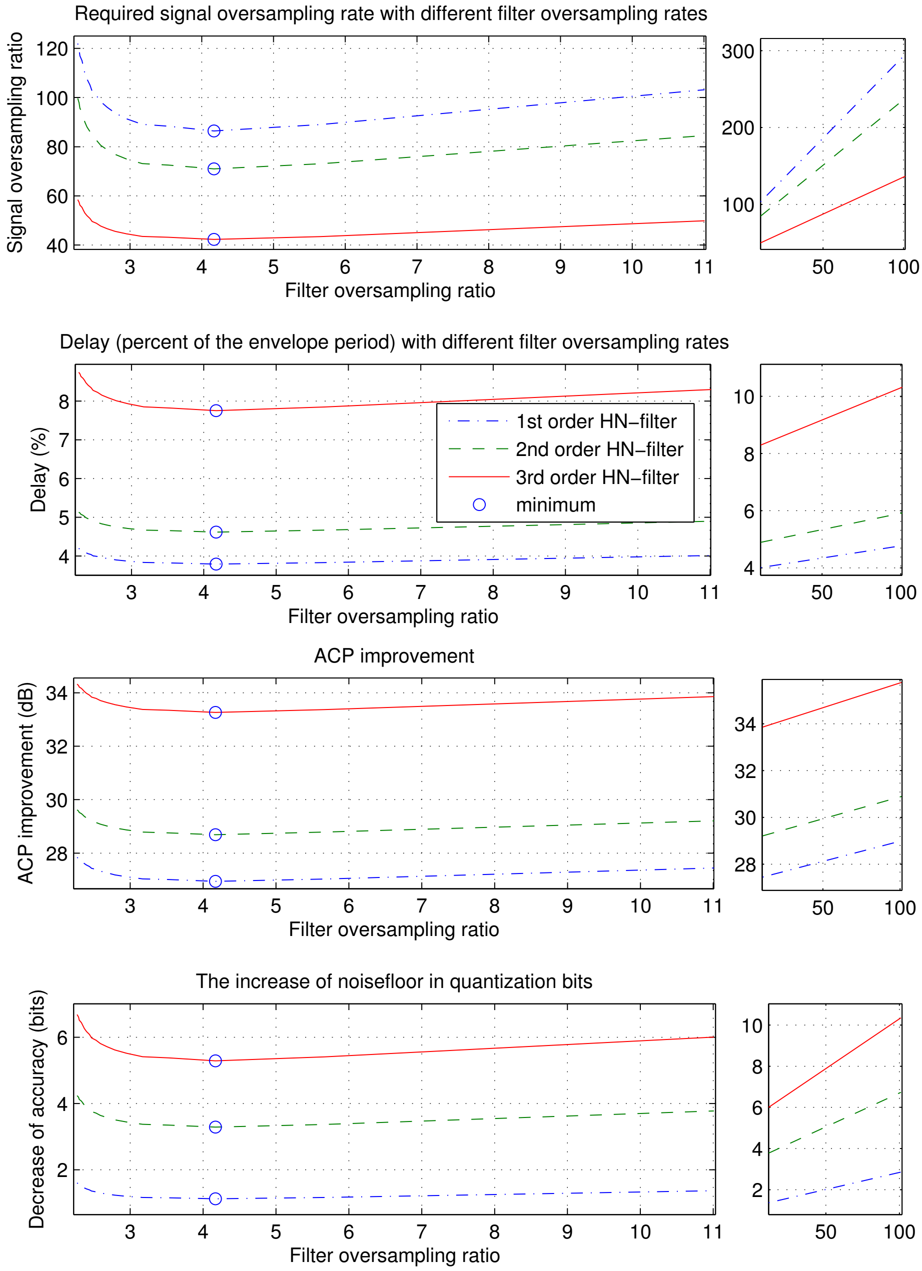


Figure 5.12: The required signal oversampling ratio for different filter bandwidths with noise attenuation (H-N filter)

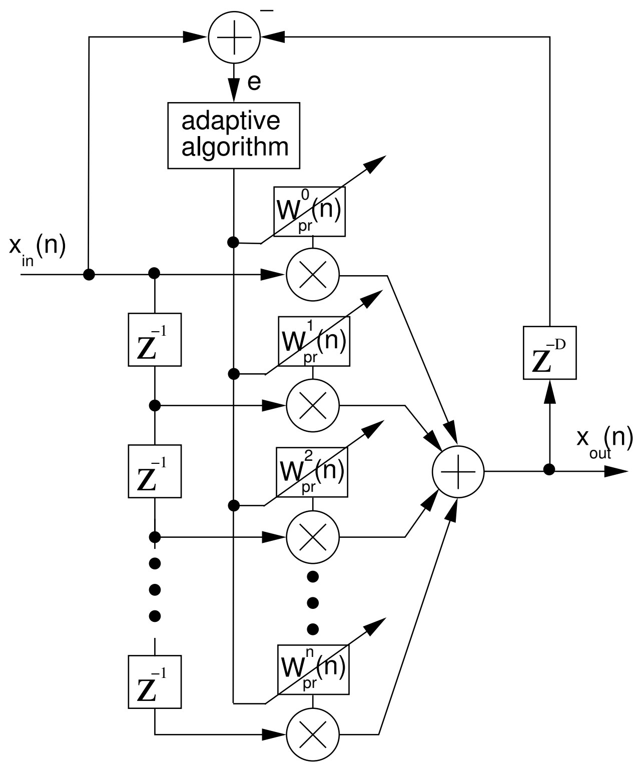


Figure 5.13: Adaptive filter structure [125]

$x_{in}(n)$ is the input signal value at time n, D is the number of clock cycles to be predicted, μ is a constant that defines the speed of convergence and stability.

A problem with the basic LMS prediction is that the optimal polynomial order and the instantaneous error varies quite a lot. This means that the required adaptation speed varies over time, which makes the environment non-stationary and further causes that the update to be either too slow or too unstable to be effective in improving the ACP. [125]

The situation can be remedied by using a non-constant adaptation coefficient μ so that the adaptation speed is altered according to the current situation. [125]

A normalized LMS (NLMS) algorithm is a fairly simple method to make the μ variable dependent on the signal value. In the NLMS algorithm, the μ in (5.25) is replaced with $\frac{\mu}{\|\overline{X}_{in}(n-D)\|}$ resulting in

$$\begin{aligned} x_{out}(n) =& \quad \overline{W}^T_{pr}(n)\overline{X}_{in}(n) \\ e =& \quad x_{in}(n) - x_{out}(n-D) \\ \overline{W}_{pr}(n) =& \quad \overline{W}_{pr}(n) + \frac{\mu}{\|\overline{X}_{in}(n-D)\|} e\overline{X}_{in}(n-D) \end{aligned} \quad . \tag{5.26}$$

This alteration maximizes the MSE reduction per iteration [125].

The RLS filter uses the previous signal values to calculate the filter coefficients that minimize the least squares error between the predicted value and the actual signal. The past signal values are used to calculate recursively the correlation matrix. The advantage of the RLS adaptation is that the convergence behavior is less dependent on the signal than with the LMS algorithm

[125] and it inherently contains a variable adaptation coefficient. However, the computational complexity is larger than with the LMS algorithm.

The RLS algorithm can be written in several forms from which the following will be used in this thesis [125]

$$\begin{aligned}
\mathbf{S}_{\mathbf{D}}^{\mathbf{NxN}}(n|n<D) =&\quad \delta\mathbf{I}^{\mathbf{NxN}} \\
\mathbf{p}_{\mathbf{D}}^{\mathbf{Nx1}}(n|n<D) =&\quad [0\,0...0]^T \\
\mathbf{X}_{\mathbf{in}}^{\mathbf{Nx1}}(n) =&\quad [x_{in}(n)\,x_{in}(n-1)...x_{in}(n-N)]^T \\
x_{in}(n|n<D) =&\quad 0 \\
\mathbf{S}_{\mathbf{D}}(n) =&\quad \tfrac{1}{\lambda}\left[\mathbf{S}_{\mathbf{D}}(n-1) - \tfrac{\mathbf{S}_{\mathbf{D}}(n-1)\mathbf{X}_{\mathbf{in}}(n-D)\mathbf{X}_{\mathbf{in}}^{\mathbf{T}}(n-D)\mathbf{S}_{\mathbf{D}}(n-1)}{\lambda+\mathbf{X}_{\mathbf{in}}^{\mathbf{T}}(n-D)\mathbf{S}_{\mathbf{D}}(n-1)\mathbf{X}_{\mathbf{in}}(n-D)}\right] \\
\mathbf{p}_{\mathbf{D}}(n) =&\quad \lambda\mathbf{p}_{\mathbf{D}}(n-1) + \mathbf{x}_{\mathbf{in}}(n)\mathbf{X}_{\mathbf{in}}(n-D) \\
\mathbf{W}_{\mathbf{pr}}(n) =&\quad \mathbf{S}_{\mathbf{D}}(n)\mathbf{p}_{\mathbf{D}}(n) \\
\mathbf{x}_{\mathbf{out}}(n) =&\quad \mathbf{W}(n)^T\mathbf{X}_{\mathbf{in}}(n)
\end{aligned} \tag{5.27}$$

The forgetting factor λ defines how much the past signal values affect the calculation and thus defines the adaptation speed. Values near zero mean shorter memory and faster adaptation and values near one mean long memory. Again, $x_{out}(n)$ is the predicted signal at time instant n, $\mathbf{W}_{\mathbf{pr}}(n)$ is a vector of length N containing the filter coefficients, $\mathbf{X}_{\mathbf{in}}(n)$ is a vector containing N most recent values of x_{in} at time n, $x_{in}(n)$ is the input signal value at time n and D is the number of clock cycles to be predicted.

In principle, the adaptive predictive filters are more flexible than polynomial filters as they can change the polynomial order according to the signal shape and even produce non-polynomial prediction functions. However, the predictor still suffers from the increased noise due to high pass filtering and, since the adaptation is not optimized to offer additional noise suppression, the excess filter order is in vain. Also, the reconfiguration of the filter to a higher or lower polynomial order takes time, as the feedback for the correctness of the prediction comes only after the actual signal value has arrived. This may even lead to instability as the filter tries to change the shape of the prediction function to something that is not anymore valid, further increasing the error. Furthermore, the hardware costs of the adaptive predictor exceed the polynomial predictor significantly. Thus an adaptive filter is not usually a recommended solution for prediction, but LMS and RLS filters will be used as reference in this thesis.

5.4.3 Simulation results

Delay compensation using prediction systems were simulated with Matlab using the RF -predistortion system based on linear LUT update described in Section 3.5.2. The circuit was simulated both by using a fixed LUT to find the effect of delay and predictors on a general LUT-based digitally-controlled RF predistorter and by using the linear adaptation to find the effect of delay in the case of the presented predistorter.

The test signal used in the simulations was a 16QAM signal with the symbol rate of 22 ksym/s and filtered with a 0.3 roll off factor root raised cosine (RRC) filter. The clock frequency was 10 MHz. PA2 was used to model the PA nonlinearity (Section 2.7). The delays were modeled by delaying the predistorter control signals by an integer number of clock cycles, no filtering was used. The delays varied from 0% to 3.5% of the envelope period. The delays between the PA input and output signals were assumed to be matched. The simulations were performed with polynomial predictors with polynomial orders from 1 to 3 (equations (5.19), (5.21) and (5.23)) and $N = polynomial\, order + 1$, a fourth order normalized LMS filter (Equation 5.26), with $\mu = \frac{1}{2^5}$ and a third order RLS filter (Equation 5.27), with $\mu = 0.9999$. A 256-entry LUT was used.

First, the prediction was simulated with a fixed LUT, in other words without adaptation. A 20-bit accuracy of calculation was used to reduce the effect of noise. The resulting first adjacent channel powers are presented in Figure 5.14. The second order H-N filter offers clearly the best results, being able to compensate fairly well for delays of over 3% from the clock period. This results from the wide bandwidth and low-enough amplification of the second order H-N predictor presented in Figure 5.10. The RLS filter and the third order H-N filter give good results and are able to compensate well over 2% delays. It should, however, be noted that, for simplicity, the internal signals of the RLS algorithm were not quantized and thus the results are optimistic. Also, the ACP curve for the third-order H-N predictor stays flat longer than the curve for the RLS predictor. The third order H-N filter already suffers from the large noise amplification. The results are worse than with the second order H-N filter and the transition from full compensation to poor compensation is very fast. The first order gives worse results than the other H-N filters and the RLS filters but it still is able to compensate 2% delays with good accuracy. Finally, the worst results are given by the LMS filter, which improves the ACP by only about 5dB and goes unstable at large delays.

Next, the word length was reduced to 16 bits to increase the noise level and thus the adverse effects of the out-of-band amplification of the higher order H-N filters. The other parameters were kept the same. The RLS and LMS filters were not used in simulations with word lengths below 20 bits. The results are shown in Figure 5.15a. The low out-of-band amplification makes the first order H-N filter the best predictor in this case, and it is able to achieve almost the same results as when implemented with 20-bit accuracy. The second- and third-order H-N filters clearly begin to suffer from the noise amplification. The compensation ability of the second order predictor has reduced to the level of the first-order predictor and the third-order filter has become quite useless. Figure 5.15b shows the effect of adding three filter orders to the H-N filters for noise reduction so that $N = polynomial\, order + 4$. The noise filtering reduces the bandwidth of the first order filter as expected. What is notable is the positive effect of noise reduction on the second and third order H-N filters. Due to the reduced noise, the second order filter almost achieves results as good as those achieved with a 20-bit word length without additional noise filtering; thus the second order filter again is the best performing predictor.

Next, the word length was reduced to 12 bits to investigate the operation of the prediction algorithm in short word length and high noise applications. No noise filtering was used. The

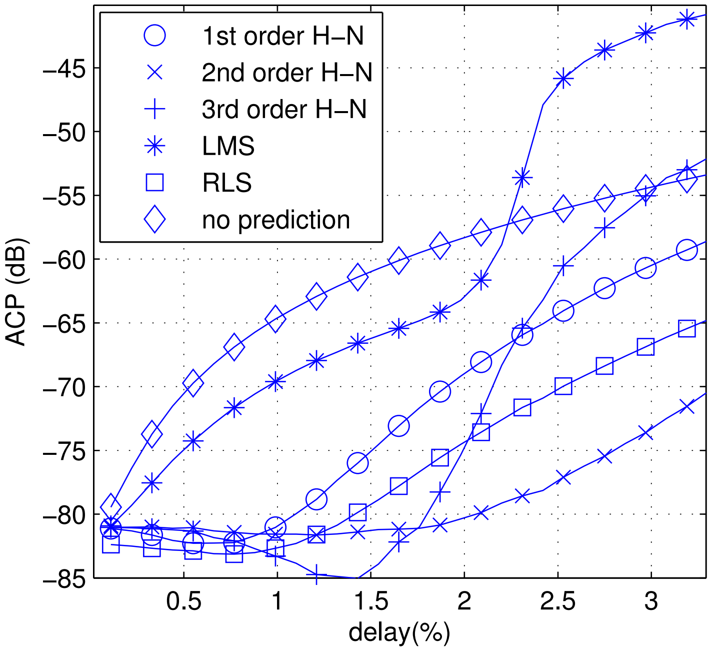
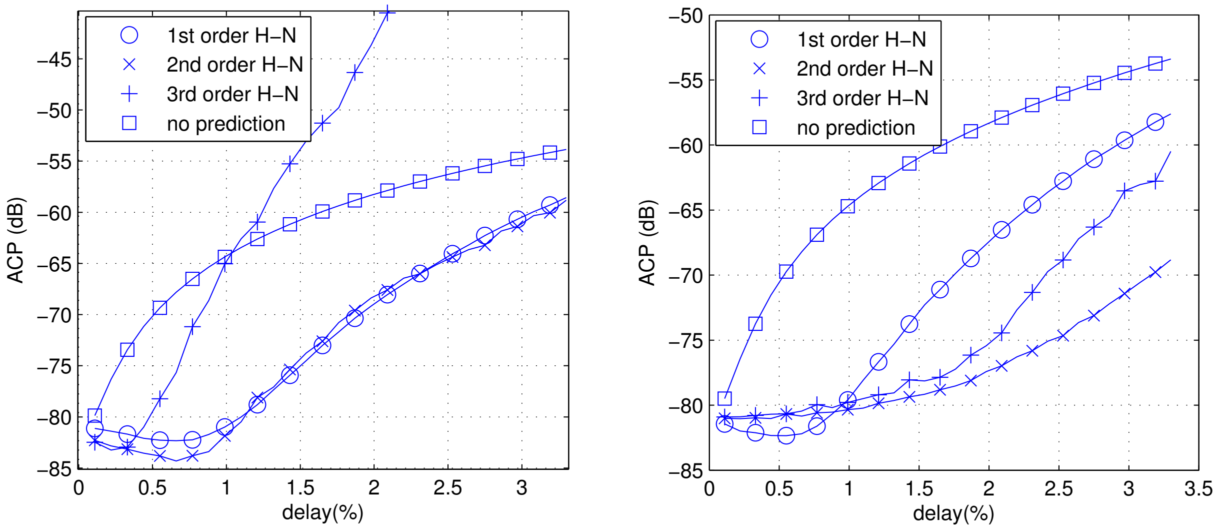


Figure 5.14: ACP as a function of the predistorter control signal delay with different prediction systems (Preloaded 256-entry LUT, no adaptation, 20-bit word length)



(a) no noise reduction

(b) 3 additional filter orders for noise reduction

Figure 5.15: ACP as a function of the predistorter control signal delay with different prediction systems (Preloaded 256-entry LUT, no adaptation, 16-bit word length)

results are shown in Figure 5.16. As can be seen, the second and third order H-N filters become useless. The low out-of-band amplification of the first order filter helps it to still give good results. Some fluctuation in the ACP, however, starts to be visible, indicating that the word-length limit is near. This is what could be expected, since according to Figure 5.10, the loss in accuracy due to the noise is 3 bits, and so the effective number of bits in the signal before the LUT address calculation is only 9 bits, being only one bit more than the required address word length.

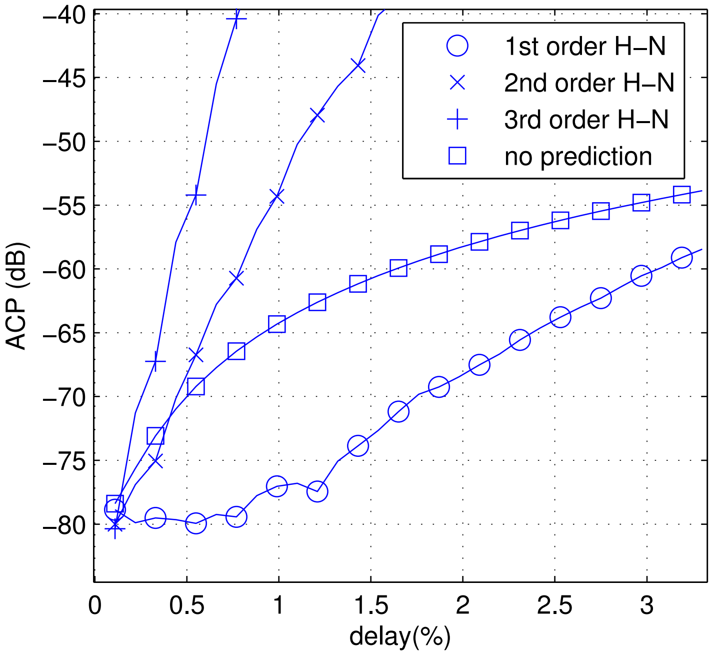


Figure 5.16: ACP as a function of the predistorter control signal delay with different prediction systems (Preloaded 256 entry LUT, no adaptation, 12 bit word length)

Finally, the circuit was simulated also using linear adaptation for the LUT update and 16-bit word length. The update algorithm was the basic linear update (Equation 3.12). The results are shown in Figure 5.17. The control signal delay clearly messes up the linear update very rapidly. Without prediction, the update becomes unstable below 0.5% delay and the ACP deteriorates quickly. The LMS, RLS and third order H-N prediction perform a little better, but still become unstable very quickly. The first order H-N is able to compensate almost 1% delays before becoming unstable. This led to the requirement of improving the basic linear update method to reduce the probability of instability in the case of inaccurate delay compensation. The improved update algorithm will be discussed in Chapter 7.

5.5 Conclusions

This chapter investigated the effect of the control signal delays and the delay difference between the predistorter input and PA output signals on the operation of an RF predistorter and methods to compensate the delays.

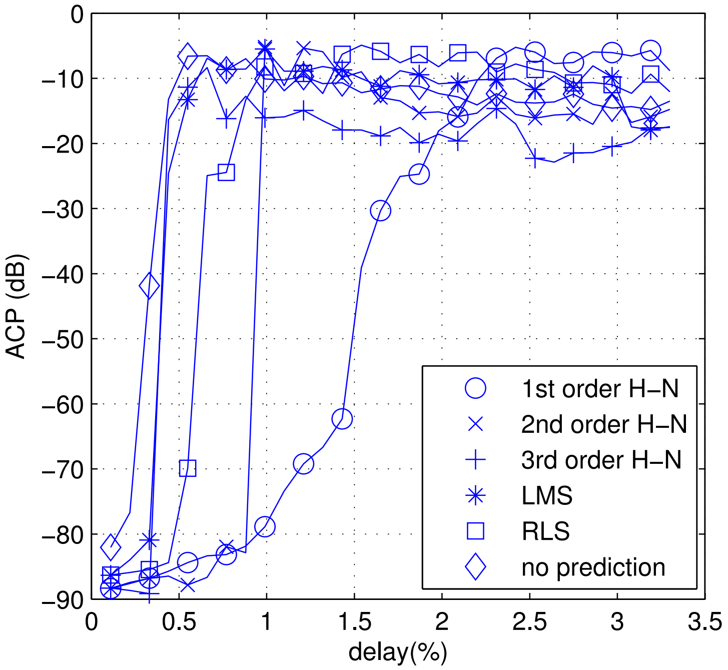


Figure 5.17: ACP as a function of the predistorter control signal delay with different prediction systems (Linear LUT adaptation, 256-entry LUT)

The residual error in the output and input signal comparison due to delay difference was calculated; it was seen that the error increases very rapidly as the delays increase, especially if the difference is calculated in analog domain using the RF signals. However, due to the fact that both signals to be compared usually have quite high frequency, the delays are fairly small and can be compensated with small analog delay elements. If the comparison is made after down conversion of the signals, the error caused by the delay decreases and the delays also can be compensated digitally.

The delay difference between the RF signal and the predistorter was seen to be very problematic for the RF predistorters. Delays of 2% of the envelope period were seen to be able to increase the ACP by almost 30dB. What makes this even more problematic is that, due to the large frequency difference between the baseband control signals and the RF signal, delays may differ by several orders of magnitude. Previously these delays have been compensated with analog delay elements, but, due to large delay differences, the required components are very large. This makes it very hard to implement a single-chip RF-predistortion module.

In this chapter, several digital predictive filters that can be used to compensate the control signal delay digitally before the LUT indexing are presented. First to third order polynomial filters were studied and it was found that, as the polynomial order increases, the maximum signal bandwidth (and prediction ability) increases, but the predictor becomes more sensitive to noise. It was seen that the first order H-N filter is the most useful of the three when using the more common word lengths, although the second order predictor gave very good results when very long word lengths were used. It was also seen that the first order H-N filter was able to compensate for over

1.5% delays efficiently. RLS and LMS filters were also considered, but, although RLS filter gave fairly good results, the complexity of these methods makes them less attractive solutions than a basic polynomial filter.

Chapter 6

Detectors and filtering in RF-predistortion systems

6.1 Introduction

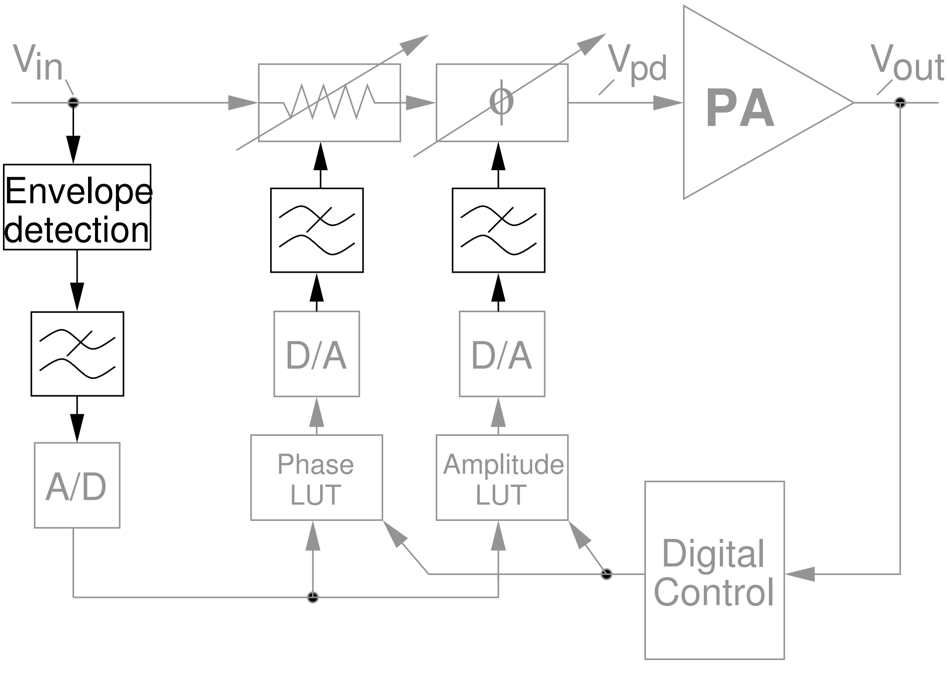


Figure 6.1: Envelope detectors and filters

The RF-predistortion systems usually use an envelope detector to generate the signals for the predistortion control. The generated signals are then digitalized and used in, for example, indexing the LUTs that contain the predistortion function or to calculate a predistortion polynomial based on the envelope. The transfer function of the detector and the consistence of the relation between the input and output signals is very important as this affects how the values in the look up tables are distributed and how accurately a detected envelope value corresponds to actual envelope value thus affecting the linearization ability. The goal of this chapter is to investigate how large the effects of the envelope detectors on the linearization ability of the predistorter are, and how the detectors should be selected to minimize their adverse effects on the predistortion.

There are three commonly used envelope detection methods, namely quadratic or power detector [134, 150, 151], linear (diode) detector [10, 114] (actually, not really linear, but only more linear than the power detector and therefore called a "linear" detector) [151] and logarithmic detector [81]. Each of these emphasize different signal amplitudes, thus giving different results on different types of signals. The different envelope detector types also spread the spectrum of the output signal differently, thus affecting the post-detection filtering requirements.

To remove the high-frequency signal components generated in the envelope detection, the predistorter input signals have to be filtered. Also, the predistorter output signals have to be filtered to remove the unwanted spectral components. In addition to delaying the control signals, the filters have other undesired effects. As the envelope detector is a very nonlinear device, it may generate a infinitely wide output signal spectrum, depending on the input signal. One therefore has to choose at which point the filter cut off should be, so as not to remove so much information from the signal that the predistortion starts to suffer. Similar effects also affect the predistorter output signals, which may be very broad band, depending on the predistortion system and the PA. Additionally, the filtering also causes frequency-dependent amplitude fluctuation to the filtered signals, which may affect the operation of the predistorter. This effect is also present in baseband predistorters.

The problems of selecting of the control signal generation method and the filtering of the output and feedback signals in baseband predistortion systems has been discussed in several publications [108, 139–141]. In baseband predistortion systems, the predistorter control signals are not filtered and the LUT index generation methods differ mainly in terms of the shape of the index generation function. In the RF-predistortion systems, the envelope detectors also differ with respect to the difficulty of biasing, sensitivity, speed and power consumption, and the filters affect the output signal of the detector. These effects have not been discussed in the literature and this chapter concentrates mainly on them.

First, commonly used envelope detector types will be discussed and evaluated by simulations, whereafter the effect of filtering is examined. Finally, a method to improve the performance of the detectors is presented.

6.2 Envelope detector models

The output of a power detector can be approximated as

$$y_{pow}(t) = |y_{bb}(t)|^2, \tag{6.1}$$

the output of a linear diode detector can be approximated as

$$y_d(t) = |y_{bb}(t)| \tag{6.2}$$

and the output of a logarithmic amplifier can be approximated as

$$y_l(t) = \log_{(a_{log}+1)}(a_{log}\,|y_{bb}(t)| + 1) \tag{6.3}$$

where $y_{bb}(t)$ is the input envelope value and a_{log} defines the shape of the logarithmic function.

The power detector can be implemented with a diode biased into square law operation condition or multiplying the signal by itself [150], the linear detector consists of a diode biased to linear operating condition [151] (actually, since the diode is a nonlinear component, the linear operation condition means that the envelope transfer function of the diode can be approximated fairly closely by a straight line, although there is deviation from the line shape, especially near the cut-off voltage) and the logarithmic amplifier can be implemented with an operational amplifier and a diode or a dedicated logarithmic amplifier chip (based on several successive amplification stages).

The selection of the detector affects the statistical distribution of the detected address values (how often the signal passes through the corresponding LUT entry), as well as the distribution of the LUT entries (how large an amplitude range each LUT entry spans). What we would like to have is a dense distribution of LUT entries at the amplitude values where the signal stays most of the time and that has the strongest nonlinearity.

The amplitude spans of the LUT entries and the amplitude concentration of a 16QAM signal are plotted in Figure 6.2 when using a power detector, linear diode and logarithmic detector. The actual transfer functions were extracted from a model for a successive amplification-based logarithmic amplifier and Schottky diode models using Agilent ADS Mentor simulator. What can be seen from the figure is that both the power detector and the logarithmic detector have the largest LUT entry span at the most probable amplitudes. This is especially a problem for the logarithmic amplifier. The linear detector has roughly the same entry span for all amplitudes. However, different shapes of the amplifier nonlinearity may benefit from different detector amplitude distributions. This will be examined in the following sections.

The power detector and the logarithmic detectors also have advantages, namely they are easier to bias than a diode detector, which requires large signal amplitudes and careful tuning [151].

6.3 The calculated effect of the envelope filtering

As the envelope detector operates by driving the signal through a nonlinear function and then removing the results that fall on the multiples of the carrier frequency instead of the DC, a filter is required after the nonlinear device to remove these unwanted components.

The effect of the reconstruction and anti-aliasing filters in baseband predistortion systems has

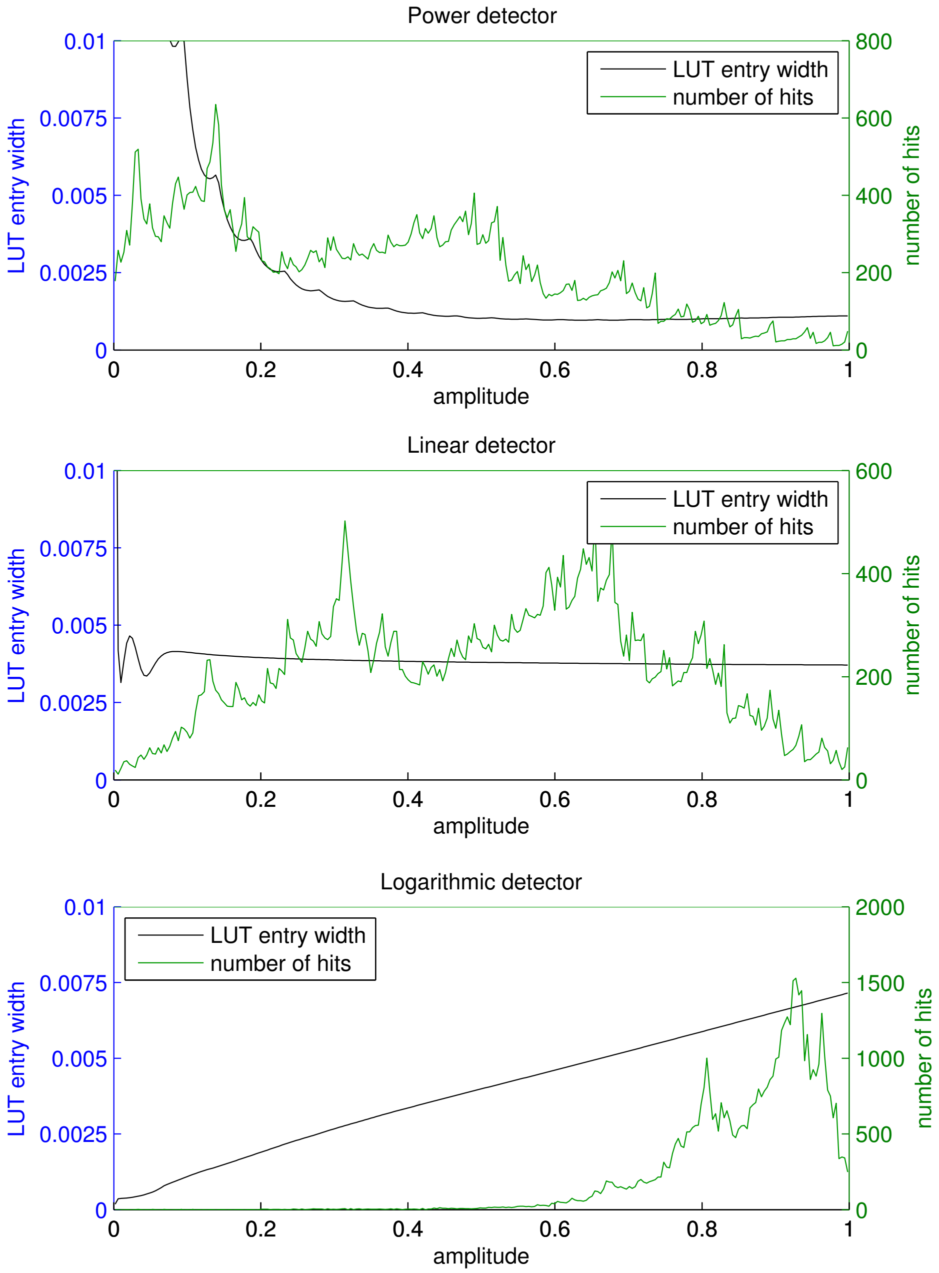


Figure 6.2: The concentration of the amplitude of a 16QAM signal and the LUT entry width in a 256-entry LUT with different detectors

previously been analyzed [140, 141]. As the envelope detection is a crucial part of the predistorter control signal generation, in RF predistorters the effect of the envelope filtering should be taken into consideration as well.

If a linear diode (Equation 6.2) that corresponds closely to the absolute value indexing [108] in a baseband predistorter is used, the spectrum of the envelope is considerably wider than the original signal bandwidth. This is due to the fact that the calculation of the absolute value requires a square root function that is quite nonlinear. A logarithmic detector spreads the spectrum even wider due to the additional logarithmic function. Both of these functions spread the spectrum infinitely, but the bandwidth can be reduced by filtering to a limit until the error caused by the truncation is negligible. The power detector has much more limited spectral spreading due to the fact that $|x|^2 = x^*x$, and thus the bandwidth is two times the bandwidth of the original signal. This leads us to the hypothesis that the power detector does not require as wide a band filter as the other detector types.

For the following consideration, the delay compensation between the RF- and control signals is assumed to be perfect and accomplished by analog means since the band limitations of the predictors and the delays of the filters affect each other, thus complicating the calculations.

The effect of envelope filtering can be analyzed theoretically by using a truncated Fourier series [152]:

$$F(x) = a_0 + \sum_{n=1}^{N} a_n cos(nx) + \sum_{n=1}^{N} b_n sin(nx) \tag{6.4}$$

where

$$\begin{array}{lll} a_0 & = \frac{1}{2\pi} & \int_{-\pi}^{\pi} f(x)dx \\ a_n & = \frac{1}{\pi} & \int_{-\pi}^{\pi} f(x)cos(nx)dx \ . \\ b_n & = \frac{1}{\pi} & \int_{-\pi}^{\pi} f(x)sin(nx)dx \end{array} \tag{6.5}$$

A four tone signal

$$\begin{array}{rcl} y_{RF}(t) & = & (sin((\omega_{rf} - 2\omega_{env})t) + sin((\omega_{rf} - \omega_{env})t) + \\ & & sin((\omega_{rf} + \omega_{env})t) + sin((\omega_{rf} + 2\omega_{env})t))/2 \end{array} \tag{6.6}$$

was selected to present a wide band variable amplitude signal. The ω_{rf} is the carrier frequency and the δ_ω is the tone frequency difference. This can be written in the baseband form

$$y_{bb}(t) = (sin(\omega_{env}t) + sin(2\omega_{env}t))/2. \tag{6.7}$$

In the following calculations two detector types are used. Namely, linear diode detector (Equation 6.2) and logarithmic detector (Equation 6.3). For the calculations we chose $a_{log} = 9$. The Fourier series of the detectors can be calculated using (6.4) and (6.5) by substituting (6.3) or (6.2) for $f(x)$. The value of N defines the truncation point of the series and can be understood as a brick wall filter that has cutoff frequency at $N\omega_{env}$.

A third-order nonlinearity (Equation 2.8) is used to model the power amplifier and a fifth-

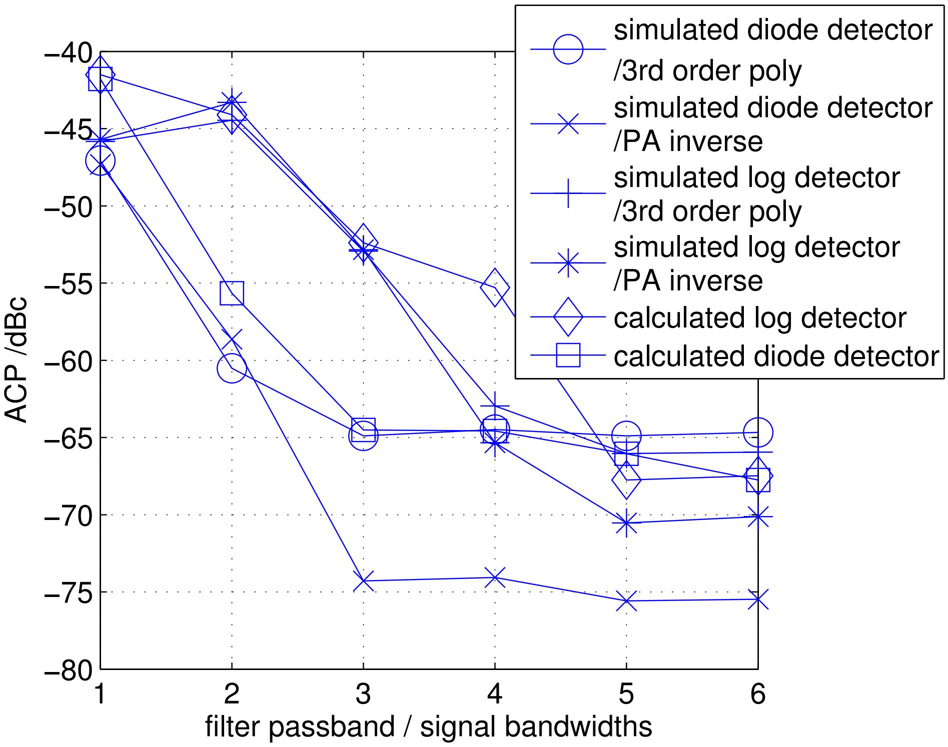


Figure 6.3: ACP as function of the filter cutoff frequency

order polynomial (Equation 3.3) is used to model the predistorter. By replacing $|V_{in}(t)|$ with $V_{df}(t)$ that is the truncated Fourier series approximation of the absolute value and use coefficients (3.4) and (3.5), we obtain

$$V_{pd}(t) = V_{in}(t)\left(1 - aV_{df}(t)^2 + 3a^2V_{df}(t)^4\right). \tag{6.8}$$

the detector output, $F(x)$, is obtained by inserting (6.7) into (6.2) and (6.3) and inserting the results into (6.4).

For the diode detector, we can simply use

$$V_{df}(t) = F(x) \tag{6.9}$$

but the logarithmic detector output after filtering has to be linearized using function

$$V_{df}(t) = (10^{F(x)} - 1)/a_{log} \tag{6.10}$$

to get an approximation for the absolute value.

Now, when we insert (6.8) to (2.8), we can calculate the generated distortion components at the power amplifier output depending on the filtering after the detector. The results of these calculations for $a = 0.07$ are presented in Figure 6.3. The figure also shows simulated results for predistorters that use a third-order polynomial stored into a 256-entry LUT for predistortion and the PA inverse transfer function stored into a 256-entry LUT. The distortion was calculated by adding first and second, third and fourth and so on Fourier coefficients together. This was done to take into account the four-tone nature of the signal. The filter passband is presented in multiples of the $2\omega_{env}$ for the same reason. The filters were modeled in the simulations with fourth order Butterworth filters using varying cut-off frequencies. ω_{env} was chosen to be $0.01 f_{clk}$.

As can be seen, the calculated curves follow the simulated. The simulations with a third order linearizer give somewhat worse results than the calculated due to the non ideal delay matching, quantization and phase offset. The simulations with the PA inverse, on the other hand, give better results due to better approximation of the nonlinearity and thus better correction than the polynomials can achieve (See Chapter 3). However, both the simulated and calculated curves clearly show that the linear diode detector requires the filter passband to be at least three times the signal bandwidth and that the log detector requires five times the bandwidth for the maximum correction.

6.4 Simulations of the envelope detector and filtering

As was seen in previous sections, a signal envelope detected with a logarithmic amplifier requires significantly wider bandwidth than a diode detector and gives the largest emphasis to the lowest amplitudes where the common digital communications signals usually spend least of their time.

When comparing the different detector types, it should be remembered that the power amplifiers nonlinearity is distributed differently in amplitude domain depending on the amplifier type. As was discussed in Chapter 2, in class-A amplifiers the nonlinearity is mostly concentrated on the high amplitudes whereas the class-B, -AB and -C amplifiers exhibit nonlinearities at both low and high amplitudes.

It would be expected that distribution of the LUT entries in such a way that the amplitude values with most nonlinearity are most densely spaced would be advantageous. Cavers [108] has presented results for different LUT indexing methods for a baseband predistorter. These methods include indexing in power, which corresponds closely to a power detector, indexing in amplitude which corresponds to linear diode and μ-law detector which is close to a logarithmic detector. However, as Cavers [108] considers only the baseband predistorter, he does not take into account the filtering. He also uses only one LUT size and does not examine the effect on the linearized ACP, so in the following analysis these aspects will also be examined.

To study these factors, a static RF-predistortion system using the three detector types described in Section 6.2 and the PA models PA1, PA2 and PA3 was simulated using Matlab. The detectors were modeled in Agilent ADS by implementing the diode detector using an Agilent HSMS2820 diode and the logarithmic amplifier with a nine-stage successive detection log amp with stage gain of 14.3 dB. The diode detector was biased to linear and quadratic operation points. The simulated AM-AM transfer functions were extracted and transferred into Matlab. The transfer functions are shown in Figure 6.4. The clock frequency in simulations was selected to be 10 MHz and the signal was a 16QAM signal with 400 kHz bandwidth with the signal maximum

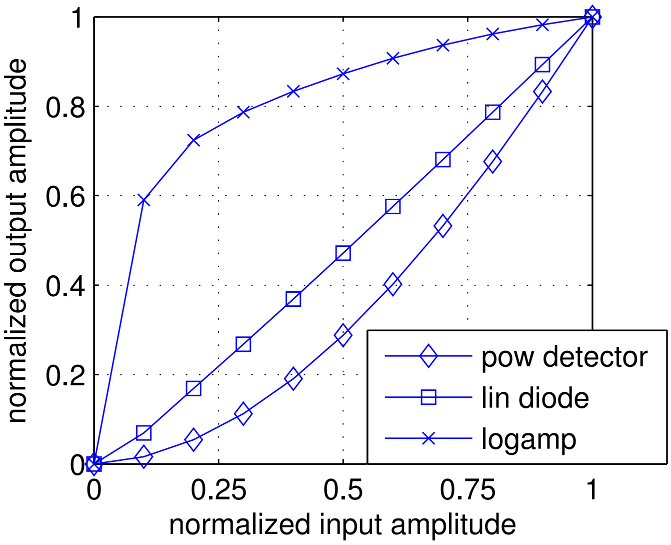


Figure 6.4: The AM-AM transfer functions of the detectors

normalized to one. The LUT was loaded with the inverse of the PA transfer function.

6.4.1 The simulated effect of envelope detector and power amplifier characteristics

To find the effect of the transfer function of the detectors to the linearization ability of an RF predistorter, the predistorter was simulated with ideal time matching and without filters. To see how the quantization affects the detected envelope, the circuit was simulated using LUT sizes from 4 to 4096 entries and the calculation word length was set to 20 bits to reduce the effect of output quantization. The signal was a 16QAM signal normalized to have maximum amplitude of one. The results are plotted in Figure 6.5.

It can be noticed from the figure, that, in every case, the maximum achievable ACP for each PA type is very nearly the same, although the address word length required to achieve this varies. It can be noted that different detectors achieve the maximum with different PA models. This means that, with proper selection of the detector, it is possible to either improve the ACP or reduce the number of bits required for the target ACP. As it can be seen, the differences can be as high as 30 dB. However, such large differences require a large number of LUT entries, which increases the hardware costs and slows the convergence down if the LUT is updated adaptively.

The power detector is advantageous when the PA has nonlinearities only at high amplitudes, but, even then, the improvement compared to the linear diode detector is small. The compression of low amplitudes clearly affects the linearization ability adversely when there are nonlinearities at the low amplitudes. The logarithmic detector is able to achieve the best linearity if there are nonlinearities at low amplitudes. However, the advantage is gained only when the required ACP is large and often other factors, such as delay matching, limit the linearity to lower ACP values. Also, the required number of LUT entries is quite large. Usually, the linear diode would offer the

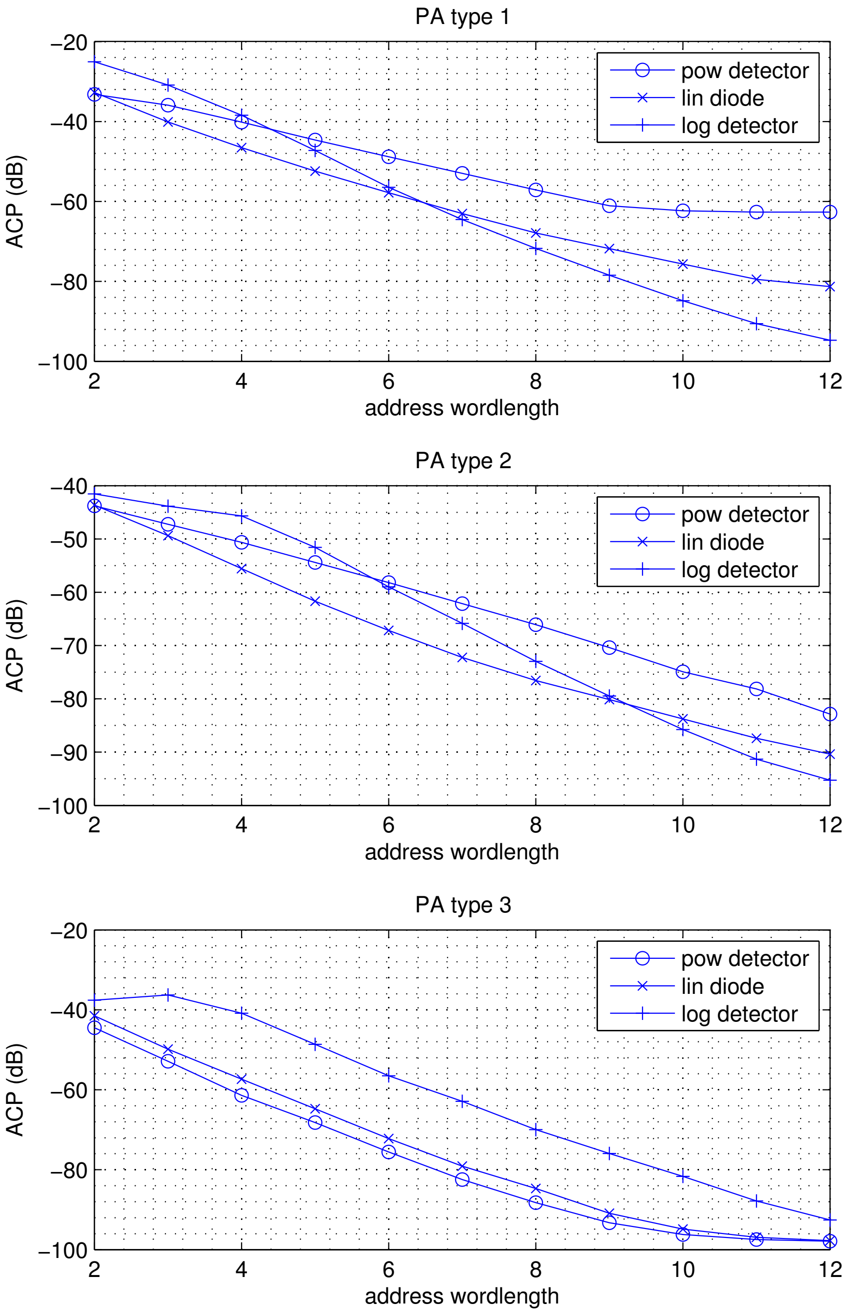


Figure 6.5: The achieved ACPs with different detectors, power amplifier types and LUT address word lengths

most general solution for envelope detection.

For an ACP of 70dBc, an 8-bit LUT address should be enough when using a linear diode or a logarithmic amplifier. In the case of PA3, even a 6-bit LUT should be enough. An 8-bit LUT will be used in most of the simulations throughout this thesis.

6.4.2 Simulations with envelope filtering

The calculations in Section 6.3 showed that the amount the signal spectrum spreads due to envelope detection varies significantly with the detector type. This affects the required bandwidth of the envelope filter. The filters also express amplitude and phase fluctuation in the passband, which may affect the linearization ability. This section will study these effects by simulations.

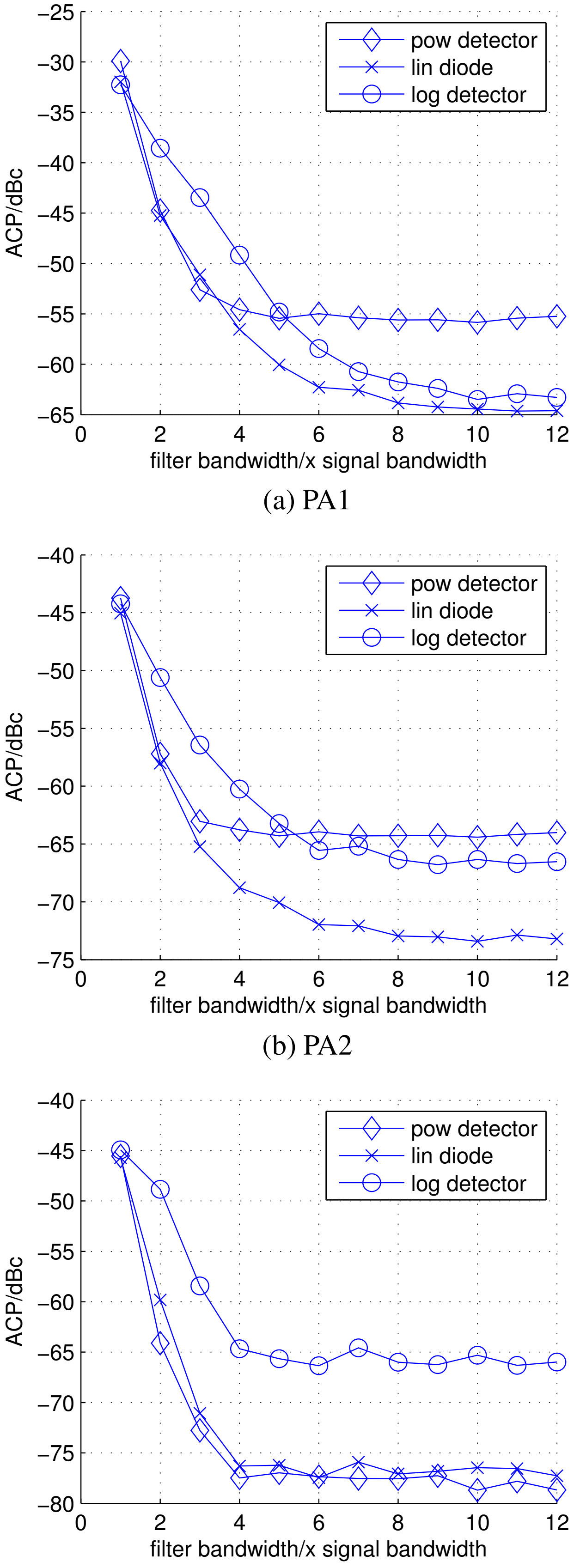
Fourth-order Butterworth filters are used to filter the envelope in the simulations. The 3-dB corner frequency was altered in 200 kHz steps, starting from 200 kHz up to 2.6 MHz. So the corner frequency varies from one signal band (SB) to 13 SB. The control signal delays caused by the filters were matched with an accuracy of 4 ns or 0.16% of t_{env} to minimize the effect of delays. The control signals for the phase and amplitude modulator were filtered with a fourth order Butterworth filter with the 3-dB frequency at 17xSB after the D/A conversion. A 256-entry LUT was used in the simulations. The input word length was 12 bit and internal word length 16 bits.

The simulated ACP with the three detectors and power amplifiers versus the filter corner frequency expressed in SBs is plotted in Figure 6.6. It can be seen that, after the filter bandwidth is wide enough, the results follow the results simulated without filters. When the plots are studied with respect to the filter bandwidth, it can be noted that the diode detector requires a bandwidth from 4 to 6 SB. The logarithmic-detector-based method requires for the same ACP from 1 to 2 SBs wider passband than the diode and power detector. These results are similar to those that were calculated in Section 6.3.

In the case of PA3, the curves follow quite well the calculated ones quite well, although the minimum is achieved with a 1 SB wider bandwidth. This, however, can be explained by the slightly better ACP minimum. In the case of PA1, the logarithmic detector gives a good minimum ACP, as expected; however, the linear diode still exceeds these results with narrower bandwidth. In the case of PA2, the results with the logarithmic amplifier are somewhat worse than expected. In none of the cases is the logarithmic detector able to give the best linearization results, due to delay mismatch and LUT size restriction.

The results support the view that the linear diode is the most general of the three detectors. It has the least bandwidth requirements in all cases in addition to having the best minimum ACP with all PA types.

The narrow bandwidth of the envelope detected by the power detector did not prove to be a real advantage in any of the cases. The bandwidth requirements are almost exactly the same as those for the linear diode detector. Even in the case of PA3, the advantage is insignificant.



(a) PA1

(b) PA2

(c) PA3

Figure 6.6: Simulation results with different PA and detector types

The fairly large bandwidth requirements of the power detector compared to the diode detector can be explained by inspecting the error caused by the filtering relative to the amplitude. Figure 6.7 shows the distribution of the filtering error expressed in terms of the number of LUT entries. DQPSK, 16QAM and two-tone signals filtered with a Butterworth filter with corner frequency at 3 SB and a 256-entry LUT were used.

It is noted that the filtering error is mainly concentrated at the low amplitudes except in the case of the power detector, which has very low error at all amplitudes. Because the power detector compresses the low amplitudes thus reducing the accuracy there, this advantage is partially lost. What is more, the transfer function of the power detector is most advantageous in the case where there are no nonlinearities at the low amplitudes. In this case, all the LUT entries in the linear part have almost the same value, and thus the error in the LUT indexing caused by the filtering does not have a large effect on the LUT output. The concentration of the filtering error on the low amplitudes hinders also the logarithmic detector, which would offer the most accurate correction at the low amplitudes.

When the filter bandwidth requirements for different PA types shown in Figure 6.6 are compared, it can be seen that the case with PA1 requires the widest bandwidth and the case with PA3 the narrowest. This agrees with the data presented in Figure 6.7, as PA1 requires the best information of the low amplitude values to be linearized properly and so the filter bandwidth needs to be wide to minimize the errors, whereas PA3 is linear at the low amplitudes.

To conclude, it seems that the linear diode is the most versatile solution ACP wise and bandwidth wise. However, as discussed earlier, the biasing of the diode to linear operation condition requires large signal amplitudes and also increases current consumption [151]. This may cause one to have to make the selection between the power and logarithmic detectors.

6.5 Linearized detectors

It would be beneficial to be able to have an envelope detector that would combine the very versatile transfer function of the linear diode detector while still retaining the easy biasing and lower power consumption of power and logarithmic detectors, especially since often the linear detector is not feasible, making it necessary to settle for the inferior transfer functions of the other detectors.

The use digital signal processing for generation of the predistortion control offers a solution for this. Since the detected signal is transferred to digital domain, an inverse transfer function of the detector can be implemented by, for example, using an LUT to approximate the transfer function of the linear detector. This section inspects the effectiveness of this solution to reduce the drawbacks of the power and logarithmic detectors and provide an easy-to-bias detector with a good linearization performance.

A linearisation function generates an inverse of the logarithmic or quadratic function so that the LUT index is a linear function of the RF signal envelope. The linearisation function for the log amplifier is of the form (6.10) and for the power detector $y_{lin} = \sqrt{y_{pow}}$. In the ideal case,

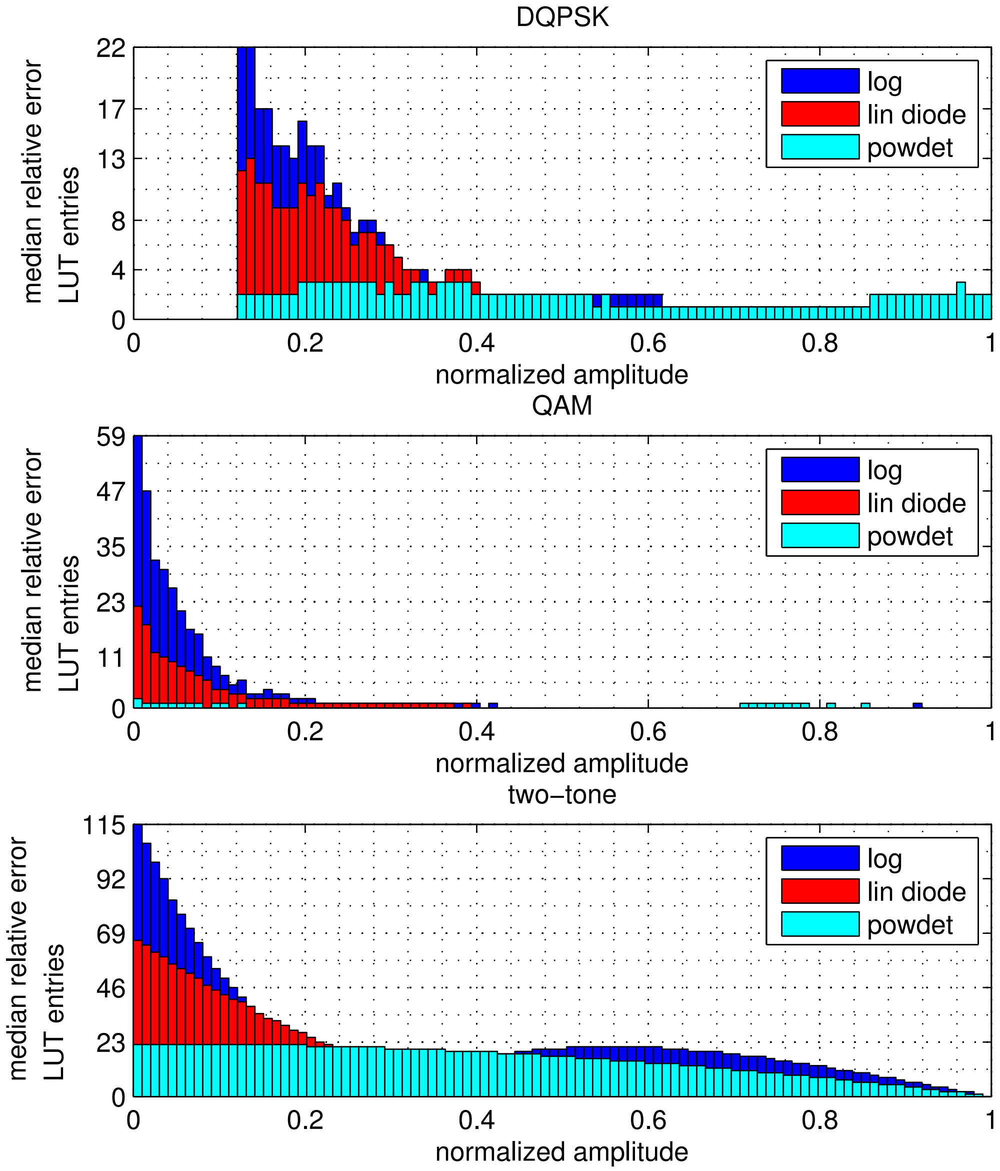


Figure 6.7: The distribution of error caused by filtering the detected envelope for different detectors and signals

these give $y_{lin} = |y_{bb}(t)|$. However, the filtering causes the actual results to deviate from these ideal results. The effectiveness of the linearized detection was investigated by simulations.

The predistortion system was simulated using the same parameters as in the previous section but with a 4096-segment linearization table added before the LUT indexing to find the actual effectiveness of the linearization. The results are collected in Figure 6.8. The most notable result is that all the detectors achieve as good an ACP as the linear diode. This means that the linearization is clearly able to remove the adverse effects due to the amplitude compression. This is also visible in the operation of the power detector. As the actual detected signal is fairly narrowband compared to the other detectors, and as the adverse effects of the low-amplitude compression are compensated by the linearization, the linearized power detector , when used in conjunction of PA1 and PA2, is able to achieve with a narrower filter bandwidth the same results as the linear detector. These PA types benefit from the removal of the low amplitude compression of the power detector.

The results clearly show that the linearization of the envelope detector works well as a method to generate a general easy-to-bias envelope detection method that still retains the linearization ability of the linear detector. The disadvantage, however, is the increased size of the digital hardware due to the additional LUT.

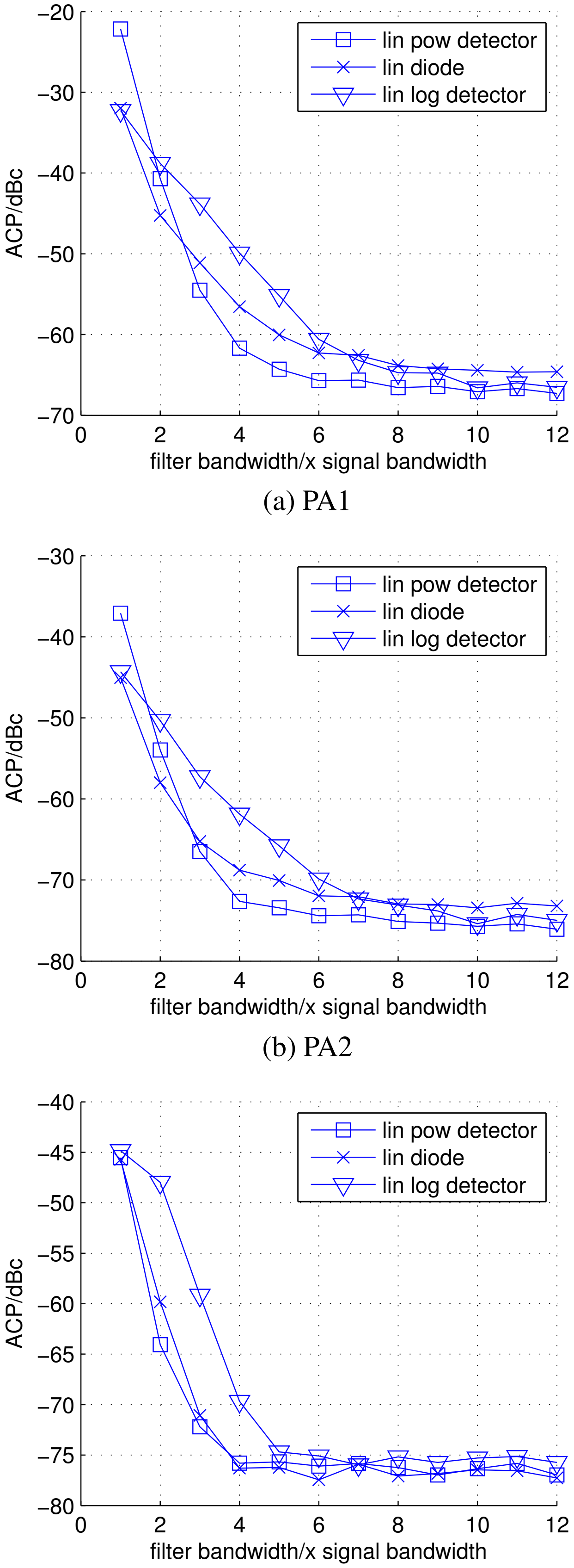
6.6 Conclusions

This chapter investigated the effect the envelope detectors and filters have on the operation of an RF predistorter and how the envelope detector should be selected to minimize its adverse effects on the predistorter.

The simulations showed how the power detector tends to compress the signal into lower amplitudes and at the same time generates the sparsest LUT entry spacing at these amplitude values increasing the quantization errors. The logarithmic detector has the same effect on the high amplitudes. Furthermore the simulation results showed how the logarithmic detector requires, due to its more nonlinear nature, over 50% more bandwidth from the envelope filter than the diode detector.

When the detectors were used without filters, the simulations showed how the logarithm detector gave better linearization results than the other detector types when the nonlinearities were concentrated on the low amplitudes and the LUT was large, thanks to its emphasis on the low amplitude values and how the power detector performed the best when the nonlinearities were concentrated on the high amplitudes, again thanks to its emphasis on these amplitudes. However in all cases the results achieved with the linear detector with small LUT sizes were the best or very close to the best.

When the filtering was added, the logarithmic detector lost its advantage and did not perform well with any of the PA models. This was shown by simulations to be result of the filters tendency to affect especially the detectors lower amplitude output values where the advantage of the loga-



(a) PA1

(b) PA2

(c) PA3

Figure 6.8: Simulation results with different PA and linear detector types

rithmic detector would be. The power detector was neither able to outperform the linear detector with any of the PA models.

The envelope detector based on diode biased to the linear operation condition, was thus shown to be the best detection method, but it may not always be a possible solution depending on the signal level. In these cases, a logarithmic amplifier or a quadratic diode has to be used.

To compensate the disadvantages of the logarithmic and power detectors, a method of linearizing the outputs of the power and logarithmic detectors digitally was introduced and tested through simulations. The method was shown to be able to improve the operation of the logarithmic and power detectors so that the results, given wide enough bandwidth were practically the same as for the linear detector.

Chapter 7

Implemented RF-predistortion system

7.1 Introduction

During the thesis work an RF predistortion system was implemented to investigate the operation of an RF predistorter and evaluate its linearization performance through measurements. This system was also used to verify the effect of the digital prediction algorithm on the predistortion.

In this chapter, the design and implementation of the RF-predistortion system is presented. The predistortion system was implemented to linearize different class-AB multi-stage amplifiers. The signal bandwidth of 18 kHz used in most measurements and carrier frequency range from 400 MHz to 420 MHz were chosen according to the TETRA standard [6]. The implementation uses an analog phase modulator and amplitude modulator on the RF path that are controlled by a digital LUT-based algorithm in an FPGA. The system is the first one to implement a phase/amplitude-modulator-based predistortion with a simple time-domain feedback and adaptive phase LUT. The previous systems have used either a static-phase LUT [10], quadrature LUT [86, 114] or more complex LUT update methods [9].

Also, measurement results are presented for the system with and without a prediction algorithm. The results of the first measurements led to development of an improved LUT update algorithm. The measurement results for the system using this updated algorithm are presented in the end of this chapter.

7.2 Hardware implementation of the RF predistorter

A block diagram of the complete system, including the measurement and control setup, is shown in Figure 7.1. The signal was divided with an on-board discrete Wilkinson power divider (block A) between the detectors and the PA. The PA signal is predistorted using an analog predistorter that consists of a SV-Microwave VP451 voltage-controlled phase shifter (block B) and a PIN diode T-attenuator (block C). A class-AB PA implemented with discrete components is situated after the predistorter (block D). Both two- and three- stage amplifiers with various gains were used in the measurements. The output is sampled using a -20 dB directional coupler (block E) for the feedback path. The feedback output is further attenuated with an adjustable attenuator to match the power level at the PA input.

After block A directs part of the input signal power to the detectors, the signal is further divided between the phase and amplitude detectors with the second Wilkinson divider (block F). The PA output signal is also divided between the phase and amplitude detectors with a Wilkinson divider (block J). The phase and amplitude detector inputs are matched to 50 Ω using matching circuits (blocks G, I, L and M).

The phase detector was implemented using an Analog Devices AD8302 phase detector chip (block H). The two envelope detectors (blocks N and O) were implemented with Analog Devices AD8313 logarithmic detectors. The outputs of the phase and amplitude detectors are amplified and filtered with active filters constructed with OPAMPs (blocks P, Q and R). The filtered output is transferred to digital domain with three 12-bit A/D converters operating at 10MHz (blocks S, T and U). The digital data is transferred to an ALTERA Cyclone FPGA operating at the same clock frequency. The FPGA (block V) contains the digital algorithm presented in Section 7.3.

The predistorter control signals from the FPGA are fed to 10 MHz 12-bit D/A converters (blocks X and Y). The outputs are filtered and amplified with active filters constructed with OPAMPs (blocks Z and a). Finally the outputs are fed to the predistorter.

The input RF signal for the system was generated with a Rohde & Schwartz SMIQ vector signal generator (block b). After the PA, the amplified signal is driven to the measurement devices (block c). The output of the PA was measured using a spectrum analyzer, oscilloscope and a network analyzer. The parameters of the digital algorithm can be altered using a Python-based interface on a PC connected to the FPGA through a serial port (block e). Full-speed digital data from the FPGA can be read with a logic analyzer connected to the FPGA (block d).

7.3 The first version of the digital algorithm

The design of the digital predistortion algorithm was started by the selection of the LUT update algorithm and the word lengths for the calculation. As can be seen from the formulas in Section 3.5.2 the secant method is significantly more complex than the linear iteration method. Also, on the basis of simulations (Section 3.5.2), the convergence speed of the linear method was seen to be good enough and the stability was better than with secant method. Thus the first version of the predistorter used a basic linear update algorithm (eq. 3.12). The implementation of the algorithm is shown in Figure 7.2.

Env_out and Env_in are the measured 12-bit PA output and input envelopes, respectively, and phase diff is the measured 12-bit phase difference signal, phase_corr and env_corr are the

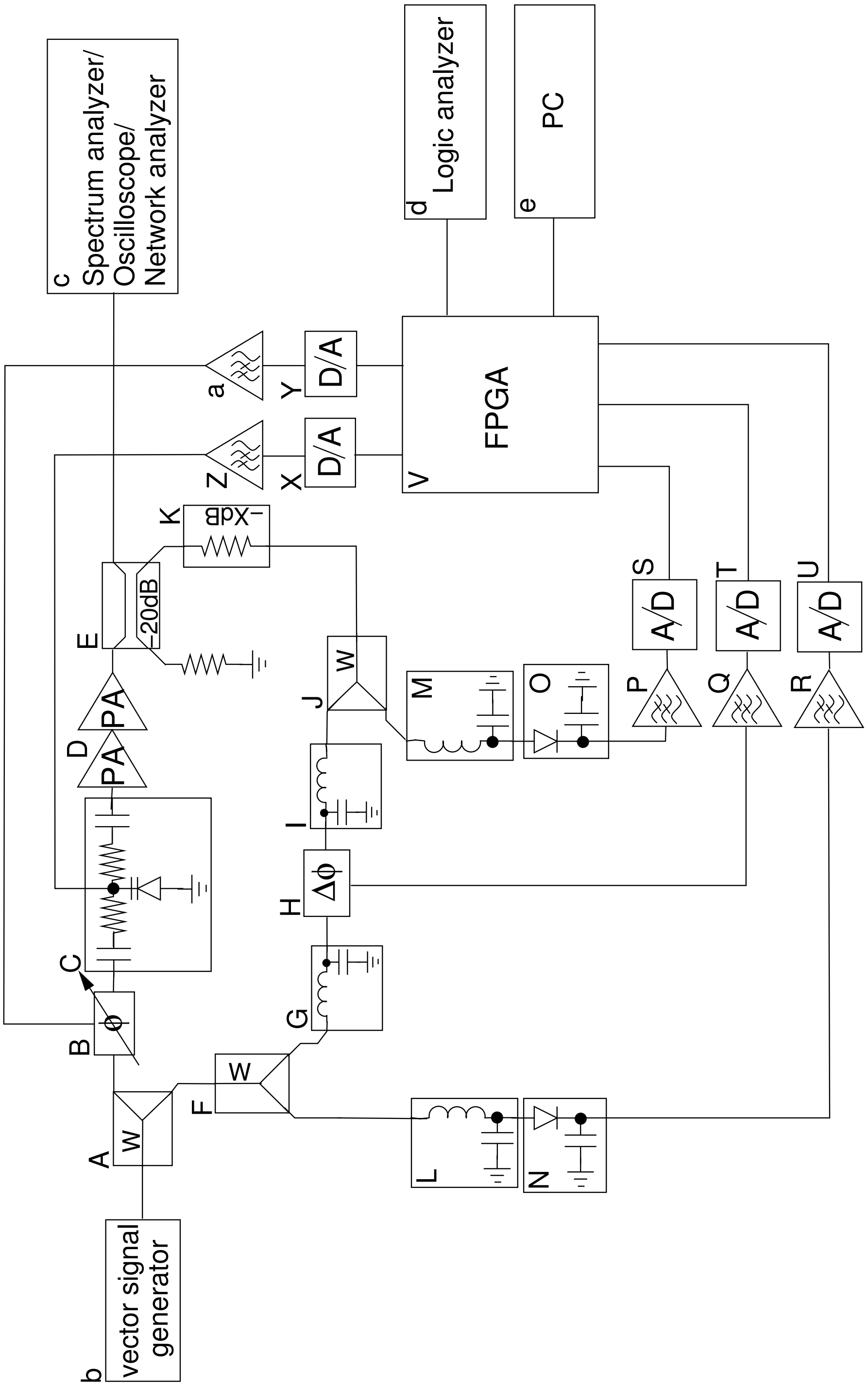


Figure 7.1: The hardware implementation of the RF predistorter and the measurement setup

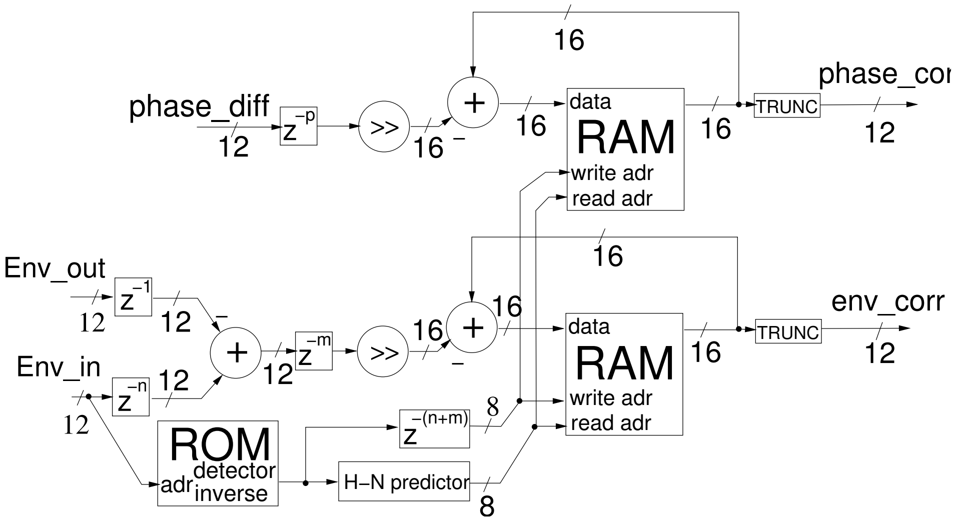


Figure 7.2: The block diagram of the basic linear update algorithm

12-bit phase and envelope correction signal outputs. The convergence coefficient (a in (3.12)) was implemented with an adjustable right shift. The word length of the internal calculations was chosen as 16 bits to accommodate the shifted bits without truncating the words.

The delay of the PA was compensated by adding an adjustable delay to Env_in before the subtraction from Env_out. There are also adjustable delays after the phase_diff input and the subtraction of the envelope signals to compensate for the delay difference between the phase and amplitude detection. The address for the LUT update is delayed to match the delays of the envelope difference signal.

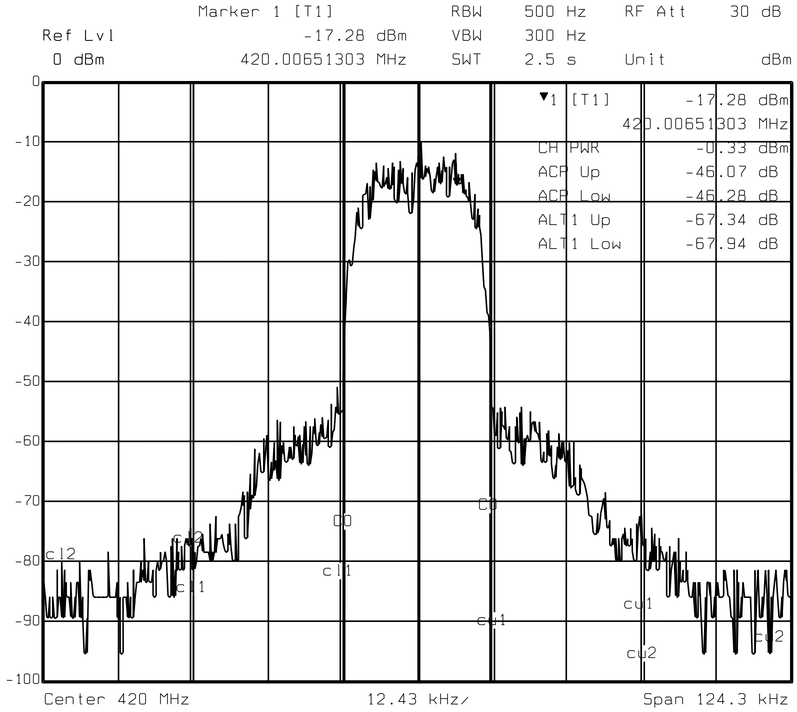
The two RAM blocks are the phase and amplitude LUTs that each contain 256 entries with 16-bit word lengths. The ROM contains the inverse of the logarithmic amplifier or diode transfer function. The ROM contents can be changed at the system startup using the PC interface. The TRUNC block truncates the RAM output to 12 bits before sending it to the D/A converters.

7.3.1 Implementation of the predictor

To reduce the control-signal delay, an H-N predictor was added to the implementation. The predictor is located in the digital part after the envelope linearization LUT (Figure 7.2). The predictor was implemented with a basic FIR with 5 coefficients. The coefficients were implemented so that they could be reprogrammed through a serial connection using a PC. The order of the predictor could be selected between first to third and the filter order between second to fifth. The predictor used 16-bit internal accuracy with 12-bit inputs and outputs.

7.4 Measurement results without predictor with the original algorithm

Measurements were performed using a 16QAM signal with an 18 kHz bandwidth and filtered with a root raised cosine filter with a 0.3 roll off factor. The carrier frequency was 420 MHz. The DC level of the phase shifter's control was adjusted to half way of the phase shifter's control range and the variable attenuator was set to 2 dB attenuation when the control signals were zero. This allowed the signal to be linearized by amplifying the signal. A two-stage amplifier with 12 dB gain was used. Figure 7.3 shows the spectrum of a 30 dBm 16QAM signal at the PA output without linearization.



```
                Marker 1 [T1]          RBW    500 Hz    RF Att     30 dB
Ref Lvl                  -17.28 dBm    VBW    300 Hz
 0 dBm          420.00651303 MHz       SWT    2.5 s     Unit         dBm

▼1 [T1]              -17.28 dBm
               420.00651303 MHz
CH PWR                -0.33 dBm
ACP Up               -46.07 dB
ACP Low              -46.28 dB
ALT1 Up              -67.34 dB
ALT1 Low             -67.94 dB

Center 420 MHz             12.43 kHz/            Span 124.3 kHz
```

Figure 7.3: The measured PA output without linearization and using an 18 kHz 16QAM signal. The output is attenuated by 20 dB.

Figure 7.4 shows the measured gain and phase shift of the amplifier as a function of the input power before and after correction. The phase curves were minimized so that the minimum was 0° and the gain curves were normalized so that the maximum gain was 0 dB. It can be seen that due to linearization, the envelope variation decreases from 1.39 dB to 0.3 dB and phase variation from 1.74^{o} to 0.37^{o} over a 20 dB range.

Although the narrow band AM-AM and AM-PM results were promising, it was noted that especially the phase LUT update was very unstable when using wider band signals. The reason for this was found to be the unacceptably large delay between the RF and control signals (Section 5.2). Figure 7.5 shows a measured spectrum of the 18 kHz 16QAM at the PA output when the phase and amplitude correction are applied and the LUT is constantly updated. A mild improvement at the upper sideband can be noted but, due to the instability, there are visible error spikes

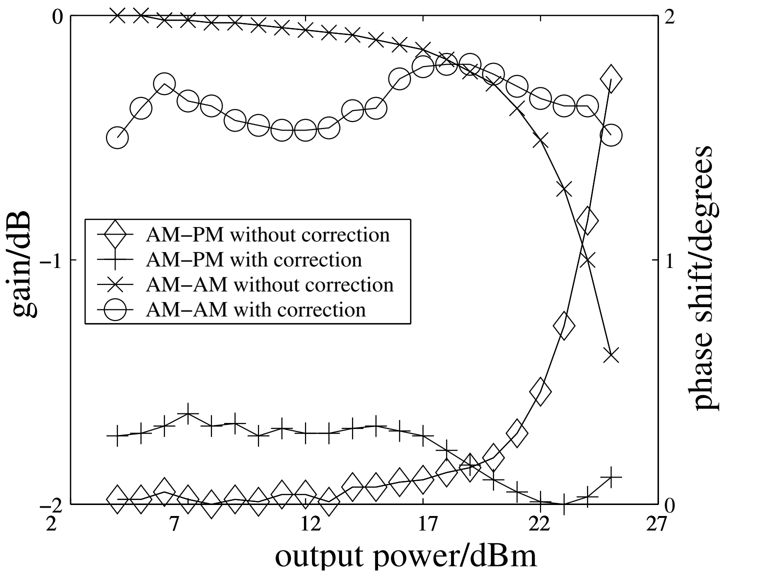


Figure 7.4: The measured AM-AM and AM-PM effects of the PA before and after correction

at the spectrum.

The problem was averted by using a 500 kHz triangular wave as a training signal to load the LUT. This signal was slow enough to be stable. The phase and amplitude LUTs were updated separately. The corrected spectrum of an 18-kHz 16QAM signal at 420 MHz carrier measured from the output of the two-stage amplifier is presented in Figure 7.6. According to the measurements, the predistorter is able to reduce the first ACP by over 10 dB and the second sideband power by 6 dB. The unbalanced sidebands signify that there are memory effects present in the system. The sources are the PA and different delays in the control signal.

The power amplifier was later updated to a three-stage amplifier to increase the gain and to test the system with a more nonlinear amplifier. The gain of the three-stage amplifier was 20 dB. The LUT update still used the separate training signal. Figure 7.7 shows the measured uncorrected and linearized spectra of an 18 kHz 16QAM signal for this configuration. The third amplifier stage increases the uncorrected ACP by 6 dB. The linearization improves the lower ACP by 8 dB and the upper ACP by 17 dB. The memory effects are significantly increased by the addition of the third stage, leading to a 9 dB difference between the sidebands.

7.5 Measurements with the Heinonen-Neuvo predictor

For the following measurements the predictive filter was programmed to be a first order H-N predictor with 5 coefficients and prediction ability of 8 clock cycles, which seemed to give the best results. The prediction ability was selected to compensate the A/D and D/A converter delay (1 and 3 clock cycles or 100 ns and 300 ns respectively), the DSP delay (1 clock cycle or 100 ns) and analog filter and detector delays (over 200 ns).

The LUT was not loaded using a test signal. Instead, the LUTs were initially empty and

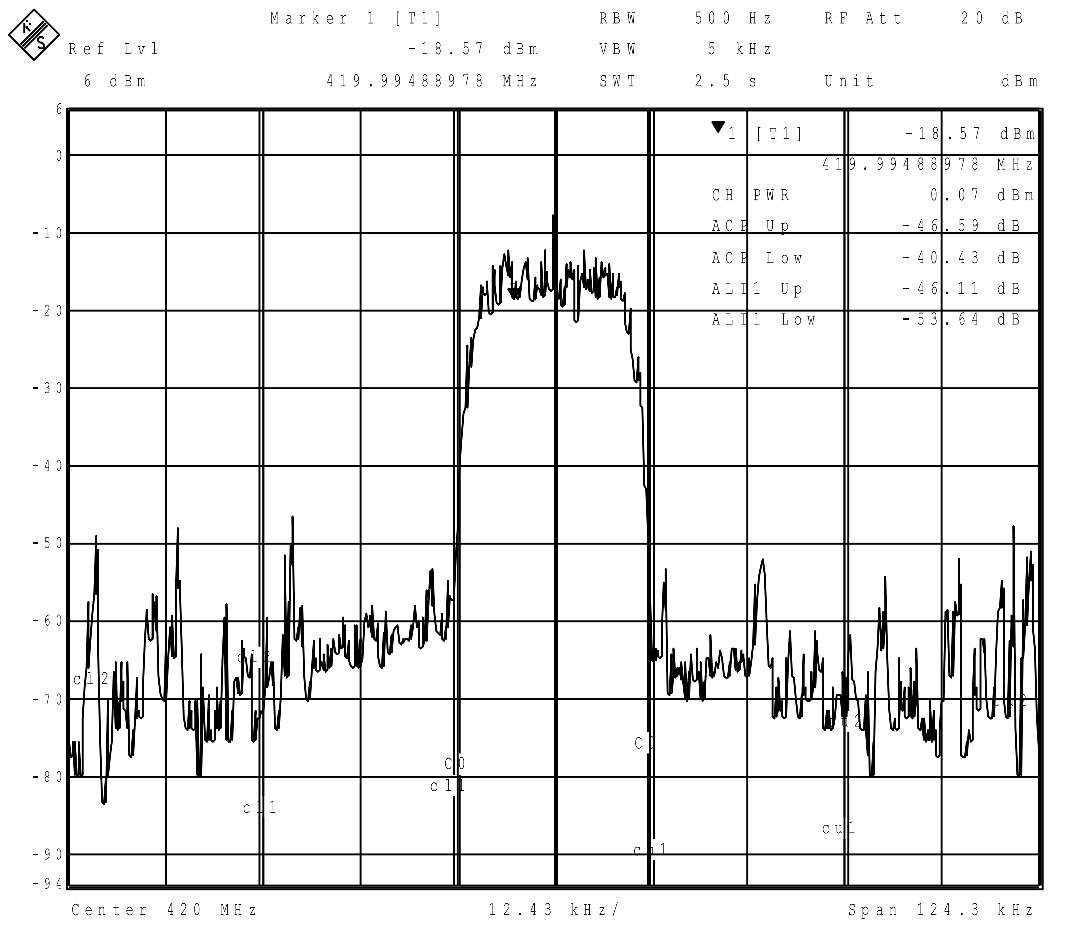


Figure 7.5: The measured spectrum using an 18kHz 16QAM signal and continuous LUT update

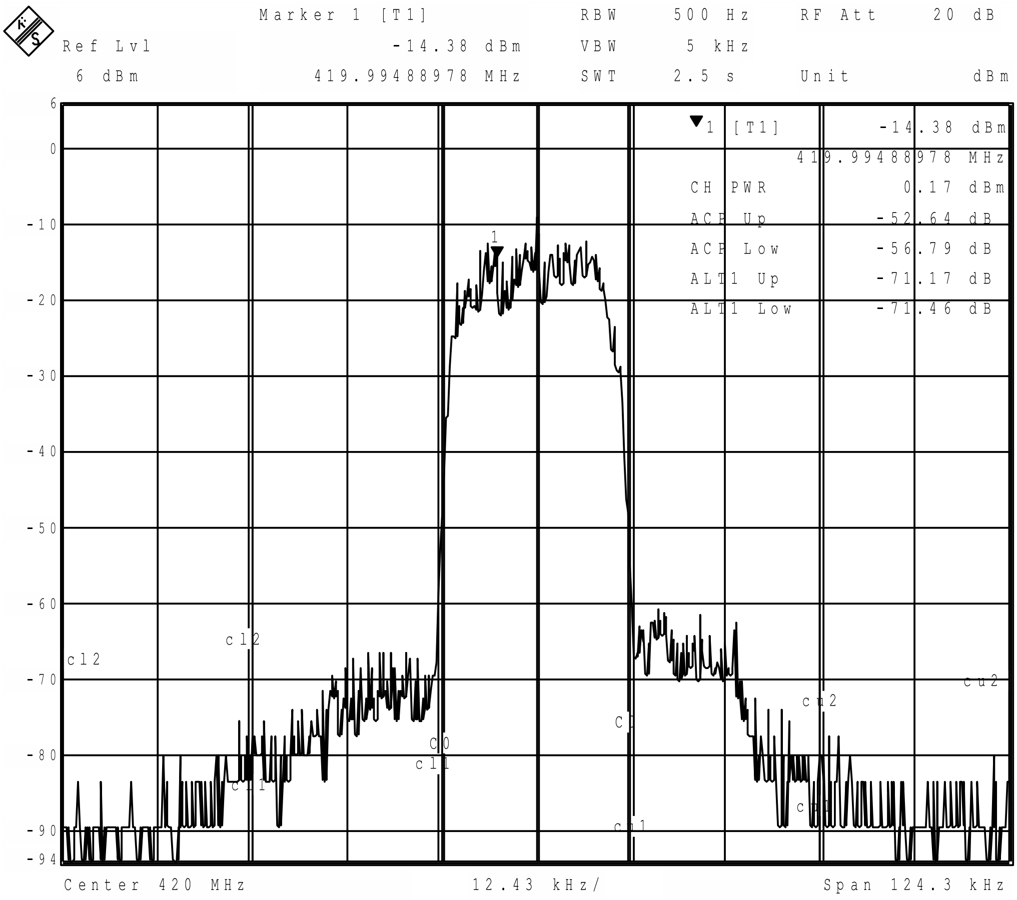
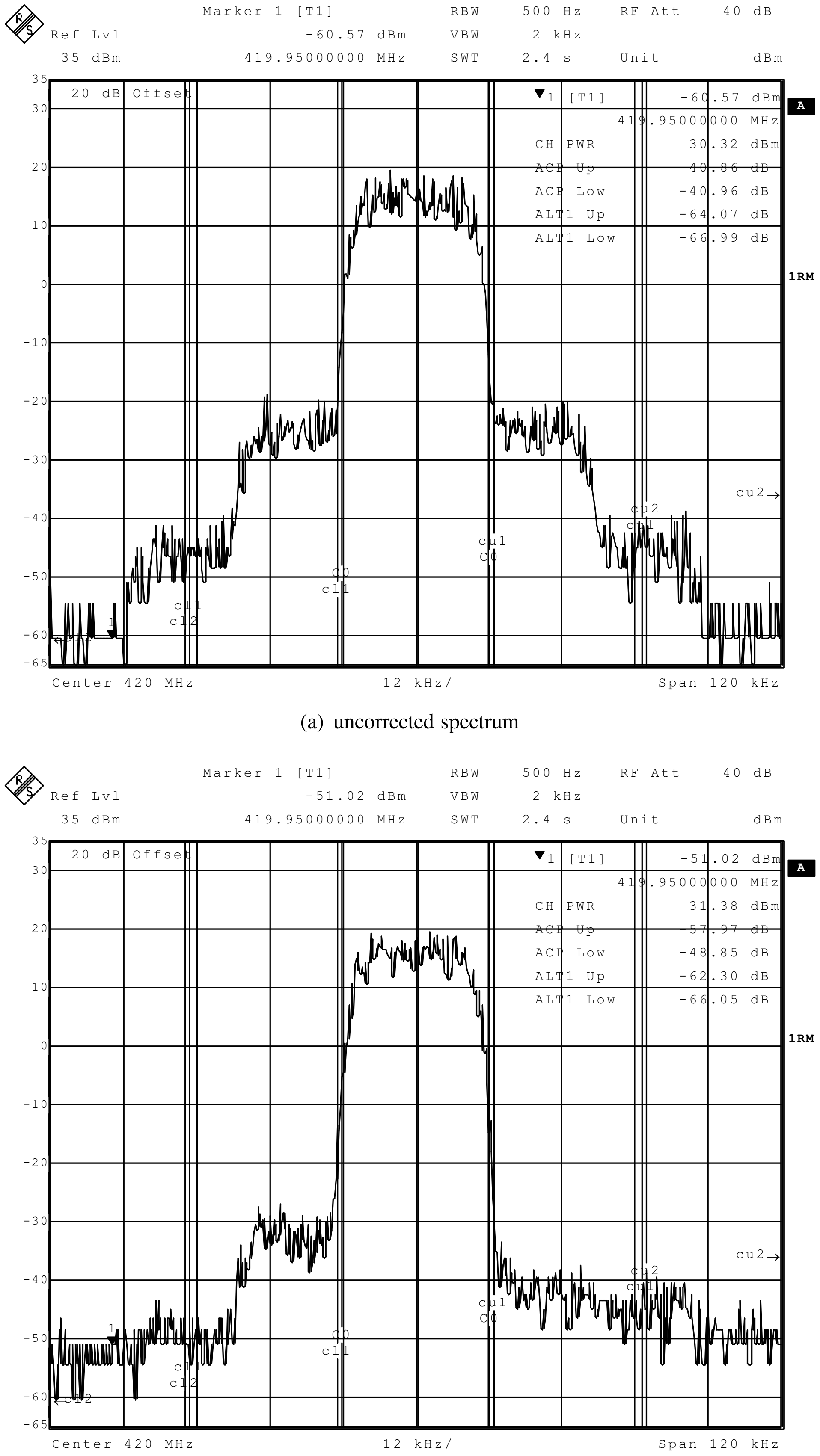


Figure 7.6: Measured 18 kHz 16QAM signal. Fixed LUT .



(a) uncorrected spectrum

(b) linearized spectrum

Figure 7.7: The measured spectrum using an 18kHz 16QAM signal and static LUT update with a three stage PA

were constantly updated with the linear update algorithm. The measurement results are shown in Figure 7.9 and in Figure 7.8. During the measurements presented in Figure 7.8, both the phase and amplitude predistortion were in use. An 8.5 kHz $\frac{\pi}{4}$-DQPSK signal was amplified. As can be seen, without the prediction the predistortion has not much effect except that of making the results worse due to the stability problems. When the predictor is switched on, the predistorter is able to reduce the instability and improve the first ACP. However, the adaptation is still not fully stable and thus the second and higher adjacent channel powers remain unacceptably high. The measurements still show that the predictor is clearly able to reduce the instability of the update caused by the delayed control signals. Unfortunately, increasing the prediction ability of the predictor was not able to improve the results due to increasing prediction error and due to larger delays than expected.

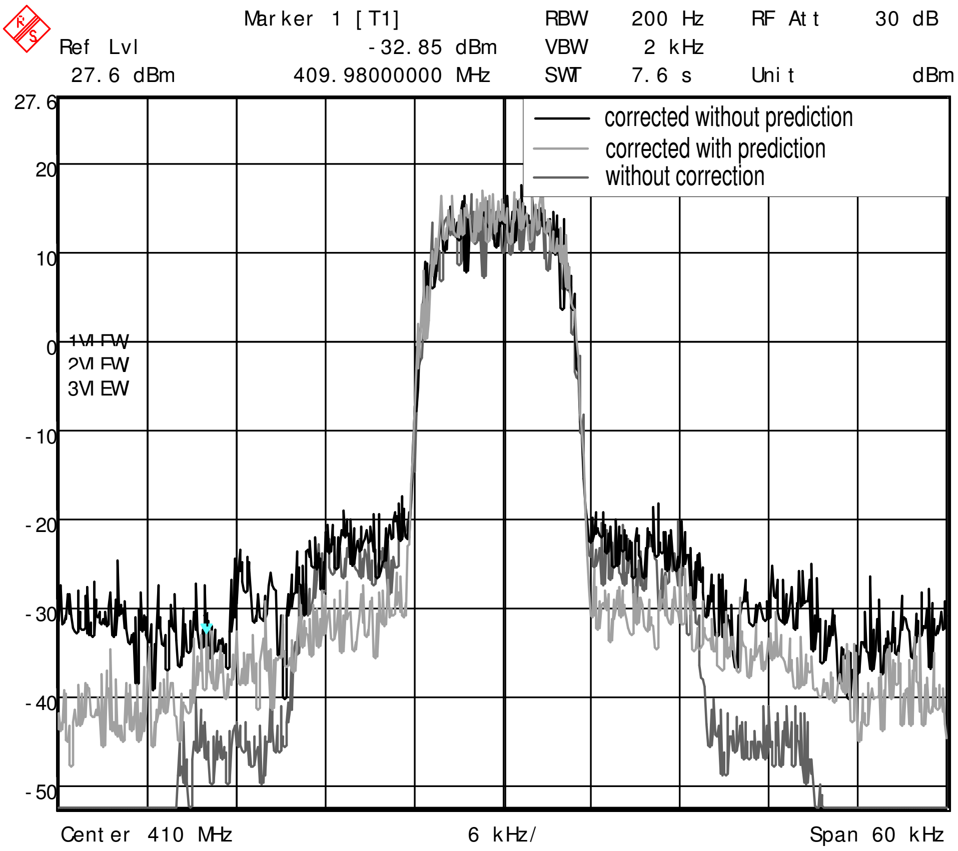


Figure 7.8: The effect of prediction on the spectrum of an 8.5 kHz $\frac{\pi}{4}$-DQPSK. Phase and amplitude adaptation on.

For wider bandwidth signals, the phase update is still too unstable, even with the predictor. Thus for the wider band measurements only the amplitude update was on. Measurement results for a 14 kHz $\frac{\pi}{4}$-DQPSK signal are shown in Figure 7.9. It can be seen that the results are even better than in the previous case due to the removal of the more sensitive phase update. However, errors that increase the ACP at higher adjacent channels still remain.

In conclusion, the predictor is able to improve the linearization results in the measurements due to increased stability.

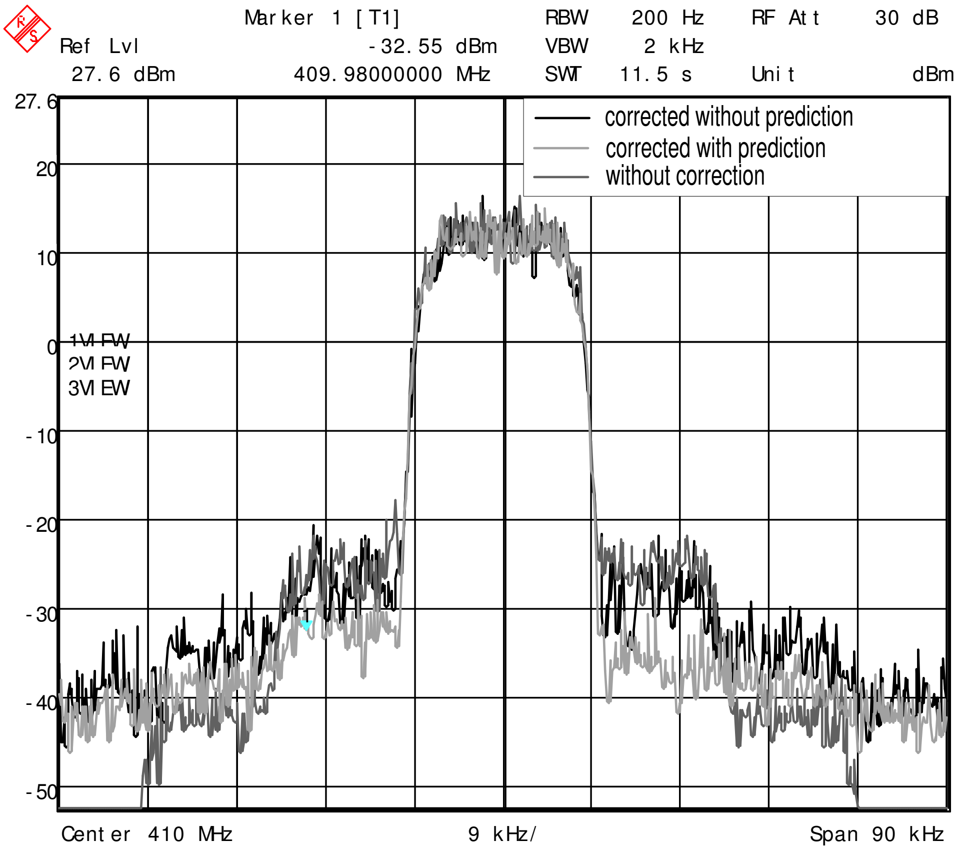
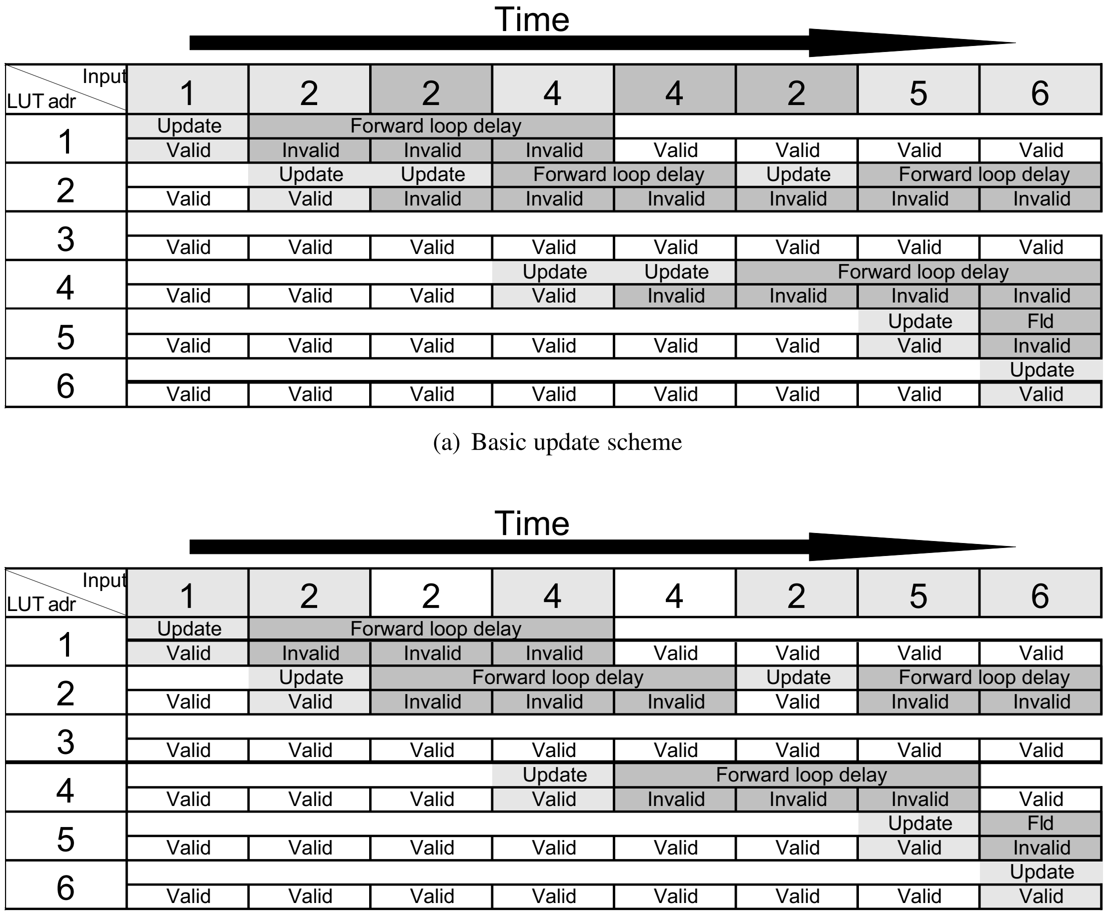


Figure 7.9: The effect of prediction on the spectrum of a 14 kHz $\frac{\pi}{4}$-DQPSK spectrum. No phase adaptation.

7.6 Improved DSP algorithm

As the results in previous sections show, it was noticed that the original digital algorithm became unstable when the errors in the envelope and more importantly phase detection errors, became too large due to noise, delays, offsets etc. Also the delays in the circuit were larger than expected. Since the instability of the update is due to the incorrect and noisy values of the detected signals, it would be beneficial to collect the signal values over a larger time span. Another problem is that, due to delay, the same error affects the update several times before the updated LUT value starts to affect the fed back signal. Thus there is undesirable overcorrection that may, in the worst case, make the update unstable and so the suppression of the update until the new LUT value is in effect is desirable.

Figure 7.10a shows the situation with the constant update. When there are several equal values in rapid succession, the LUT values are updated again before the previous update has taken any effect. As can be seen, three of the updates are invalid. In Figure 7.10b, the update of an LUT entry is disabled until the new value has taken effect. We can note that, now there are no invalid updates, there are two clock cycles without update and now the third and last update is valid.



Input / LUT adr	1	2	2	4	4	2	5	6
1	Update	Forward loop delay						
	Valid	Invalid	Invalid	Invalid	Valid	Valid	Valid	Valid
2		Update	Update	Forward loop delay		Update	Forward loop delay	
	Valid	Valid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
3								
	Valid	Valid	Valid	Valid	Valid	Valid	Valid	Valid
4				Update	Update	Forward loop delay		
	Valid	Valid	Valid	Valid	Invalid	Invalid	Invalid	Invalid
5							Update	Fld
	Valid	Valid	Valid	Valid	Valid	Valid	Valid	Invalid
6								Update
	Valid	Valid	Valid	Valid	Valid	Valid	Valid	Valid

(a) Basic update scheme

Input / LUT adr	1	2	2	4	4	2	5	6
1	Update	Forward loop delay						
	Valid	Invalid	Invalid	Invalid	Valid	Valid	Valid	Valid
2		Update	Forward loop delay			Update	Forward loop delay	
	Valid	Valid	Invalid	Invalid	Invalid	Valid	Invalid	Invalid
3								
	Valid	Valid	Valid	Valid	Valid	Valid	Valid	Valid
4				Update	Forward loop delay			
	Valid	Valid	Valid	Valid	Invalid	Invalid	Invalid	Valid
5							Update	Fld
	Valid	Valid	Valid	Valid	Valid	Valid	Valid	Invalid
6								Update
	Valid	Valid	Valid	Valid	Valid	Valid	Valid	Valid

(b) update scheme with update suppressed during the loop delay period

Figure 7.10: The effect of too a frequent update on LUT update

Naskas et al. [138] suggest a batch update for a baseband predistortion system, that can remove the invalid updates caused by the forward loop delay. This batch update method uses a training signal that is driven through the PA and the output values are collected into memory. After the training signal has ended, the new LUT values are calculated according to the measured signal. After that, the LUT is updated and the normal operation commences using a fixed LUT. However, the requirement for a training signal makes the method complicated. Also, the predistorter is actually a static predistorter, since the update is performed only when the predistorter is started and not during the actual operation.

It was decided to add some properties of the batch update to the basic linear LUT update algorithm but to still retain the use of the actual data signal for LUT update and the continuous LUT update. To accomplish this, a memory that contains one entry for each LUT entry was added. The averages of the phase and amplitude differences over fixed number of samples that correspond to the respective LUT entries are stored into these entries. When enough samples are collected, the LUT is updated according to the stored values using the basic linear update algorithm. Then the update is suppressed for a fixed number of clock cycles or until the LUT entry changes. This adds to the algorithm the averaging property and also reduces the possibility of instability due to updating invalid LUT entries. On the whole, the algorithm operates as follows:

1. Set the averaging counters and the update-suppression counters to zero and clear the LUT and the memory
2. Calculate the LUT address and the difference of the input and output envelopes
3. If suppression counter is zero, or the LUT address $\neq$ last update address, add the differences to the averaging memory and add the corresponding averaging counter by 1
4. Subtract suppression counter by 1
5. If averaging counter > averaging limit calculate the new LUT values with (3.12), update the LUTs and set the averaging counter to zero and the suppression counter to 10
6. Go to step 2

The improved algorithm is shown in Figure 7.11. The additions performed to the original algorithm did not increase the hardware cost unacceptably much and the whole algorithm could be still implemented fully on the FPGA.

To verify the improvement, the modified circuit was simulated with MATLAB. The system setup was the same as that was used for Figure 5.17, with the algorithm changed to the new one and with a value of a in (3.12) set to 0.5 and the averaging limit to 16 clock cycles. Figure 7.12 shows the simulated results. What can be seen is that the simulations without the linearization stay stable for much larger delays than before modification. This is important, as the predictor may not compensate the delay exactly and the remaining delay affects the system just as an equally large uncompensated delay (or the effect can be even worse as the delay error is accumulated

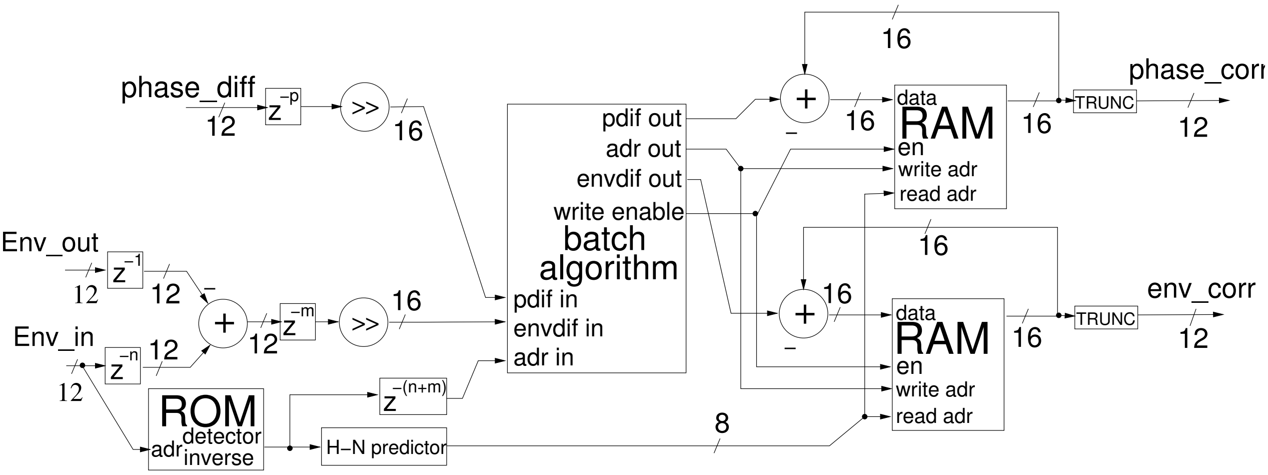


Figure 7.11: The DSP algorithm with the batch update

with the errors due to the predictive algorithm) and the circuit should remain stable even if the residual delay errors are present. The improved algorithm can be seen to improve the stability of the prediction methods, especially with the first order H-N filter.

7.7 Changes in the measurement setup

In addition to the changes in the DSP algorithm, the predistorter hardware was slightly modified. Due to the simulation results presented in Chapter 6, the logarithmic detectors (blocks N and O) were replaced with Agilent HSMS2820 Schottky diodes biased to nearly linear operating point. Furthermore, the ROM containing the detector inverse was loaded with the inverse of the diode detector to make the detector more linear, while the amplifier to be linearized was again changed to a two stage class-AB amplifier with 22 dB gain due to the fact that the three stage amplifier implementation proved to be too fragile for measurements and the number of component breakdowns became too high. The carrier frequency was also reduced to 400 MHz to better match to the measurements of the baseband predistorter.

7.8 Measurements with the improved algorithm without teaching signal

The implemented improvements were tested with measurements using the measurement setup described in the previous sections. Several bandwidths were tested to test the limits of the system and also to see how it operates with less demanding narrow bandwidth signals. Two tone signals were to investigate the contribution of different orders of distortion to the results.

The operation of the algorithm was first tested with a 1 dBm 3 kHz two tone signal input, with only amplitude correction and with both amplitude and phase correction. The results are shown in Figure 7.13. As can be seen, the linearization now remains clearly stable and the noise floor does

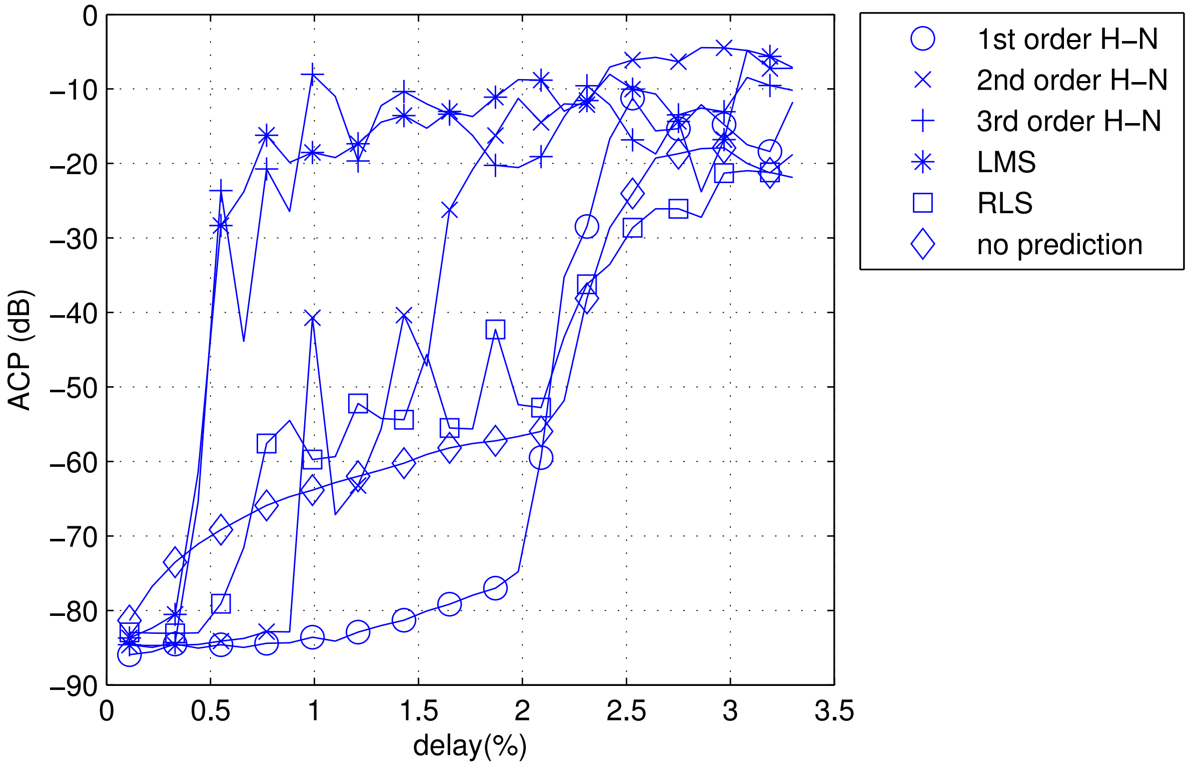


Figure 7.12: The simulated ACP of the predistorter using the improved linear adaptive algorithm

not increase due to linearization. What can be seen also is that the higher order distortion components do not increase due to the linearization. The addition, the phase correction clearly improves the lower third-order distortion component and the upper fifth-order distortion component.

The bandwidth was next increased to 18 kHz. Figure 7.14 shows the measurement results for the 18 kHz 23 dBm AM signal. The update is still stable, but, with the wider band signal, memory effects become more significant. When there is only amplitude correction present, the lower third order distortion component is reduced by over 20 dB, whereas the upper third order component remains the same as without correction. However, the improvement in the fifth-order component is not large.

When also the phase correction is operational, it can be seen that the memory effects change shape and the lower third order distortion component returns to the original level and the higher order component is the stronger one. This would either imply that there is different kind of memory on the phase and amplitude correction branches, or that the imbalance of the lower and upper distortion coefficients is actually due to the minor memory effects that the phase correction can reduce. If the amplitude correction is tuned to improve the other distortion coefficient, and now that the phase correction works, the tuning is lost and the sidebands return closer to each other, thus removing the gained linearity improvement also.

To find the performance of the updated linearizer in a more real-life situation, the PA was used to amplify a 32QAM signal. First a signal with 3 kHz and 23 dBm output power was

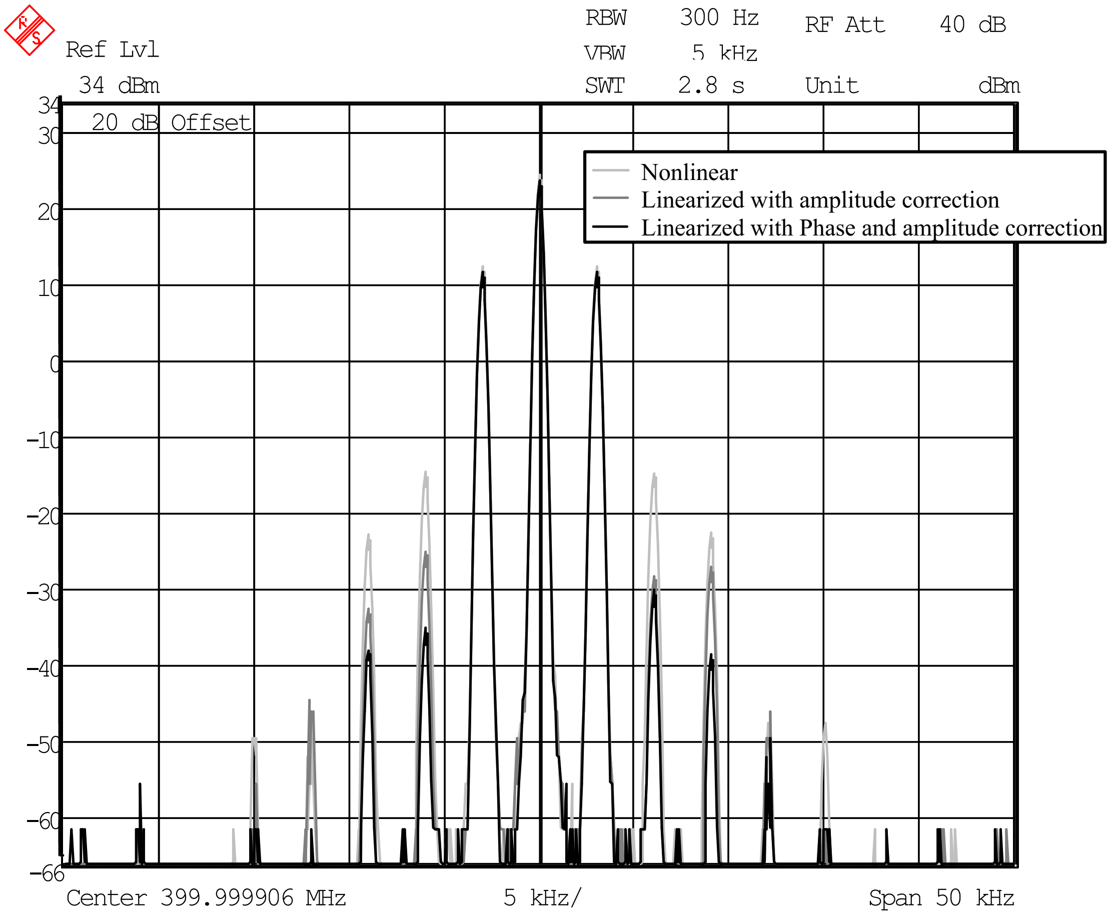


Figure 7.13: Spectrum of 3 kHz AM signal amplified with and without RF-predistortion and with both amplitude and phase correction in operation and with only amplitude correction.

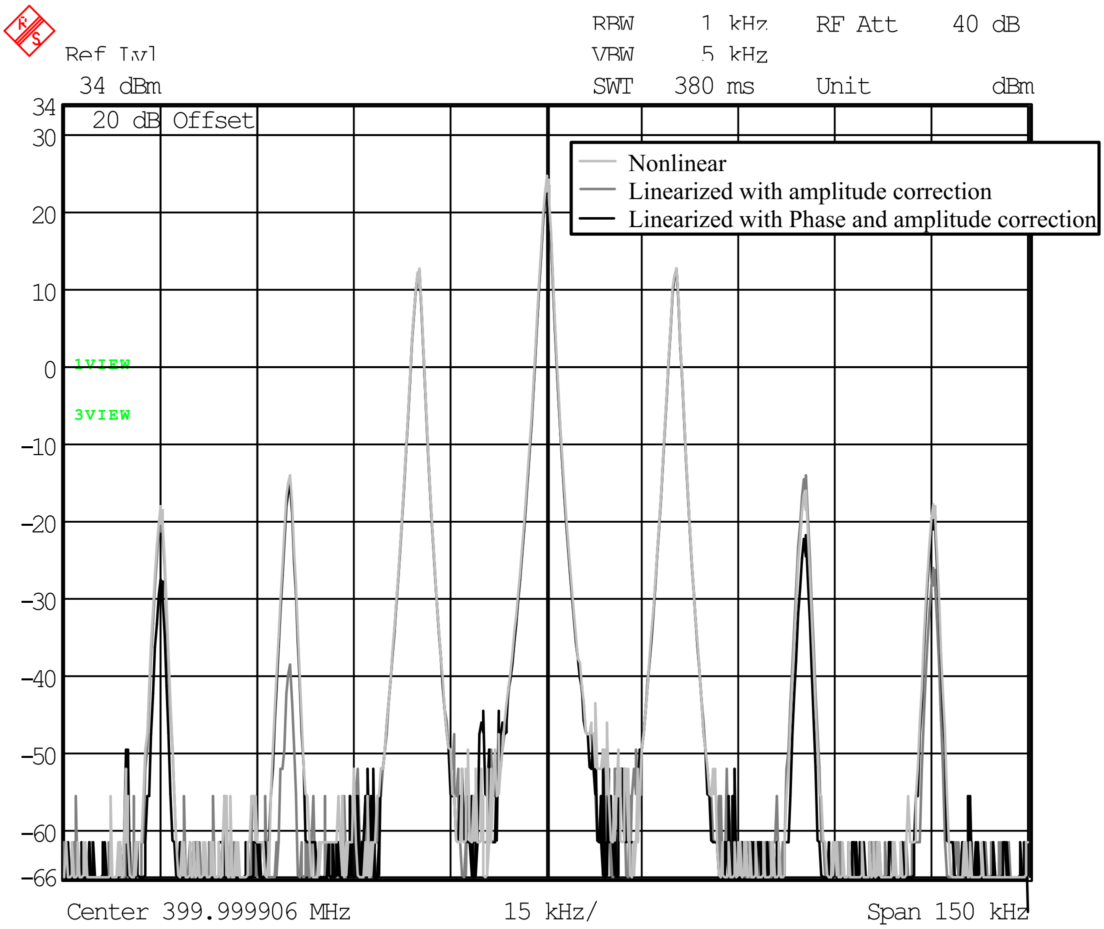
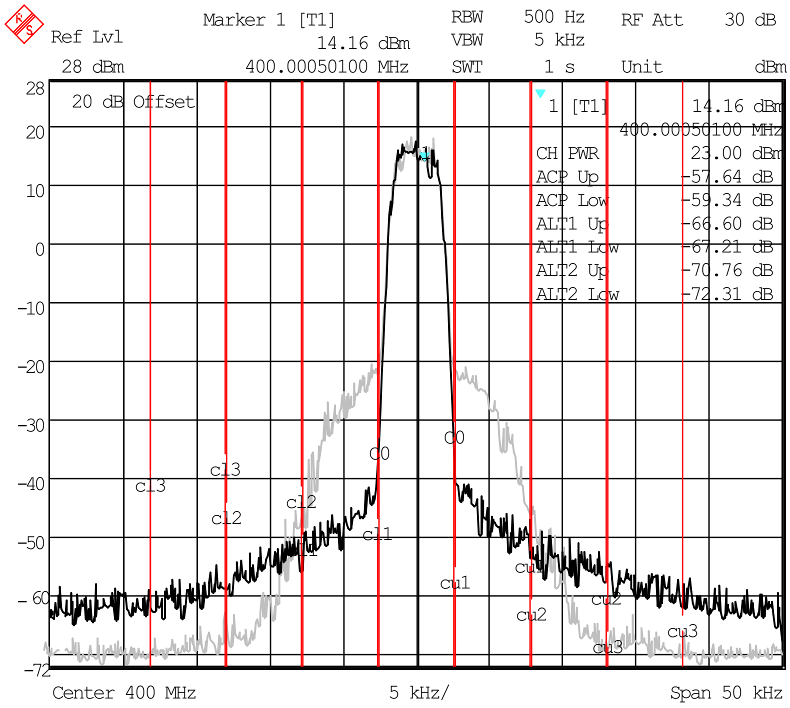


Figure 7.14: Spectrum of 18 kHz AM signal amplified with and without RF-predistortion and with both amplitude and phase correction in operation and with only amplitude correction.

used. The results are shown in Figure 7.15. The linearization improves the first upper and lower sidebands by 20 dBc but the ACP of the higher sidebands increases due to the remaining nonlinear components. The worse results compared to the AM sine are due to the greater dependency of QAM signals from the phase linearity.



```
Ref Lvl        Marker 1 [T1]          RBW    500 Hz   RF Att    30 dB
                       14.16 dBm      VBW      5 kHz
28 dBm           400.00050100 MHz     SWT      1 s    Unit        dBm

20 dB Offset                          1 [T1]              14.16 dBm
                                                   400.00050100 MHz
                                      CH PWR              23.00 dBm
                                      ACP Up             -57.64 dB
                                      ACP Low            -59.34 dB
                                      ALT1 Up            -66.60 dB
                                      ALT1 Low           -67.21 dB
                                      ALT2 Up            -70.76 dB
                                      ALT2 Low           -72.31 dB
```

Figure 7.15: Spectrum of a 3kHz 32QAM signal with and without linearization (phase and amplitude linearization)

Figure 7.16 shows the measured phase rotation and gain of the power amplifier plotted against input amplitude. Both the phase and gain become significantly flatter after the correction. The best results are achieved at the high amplitudes where the noise and other error sources are least significant. At the low amplitudes, the errors start to hinder the performance and the measurements spread over wider phase and gain ranges. The removal of the sources of these errors would be very crucial to the development of the circuit.

Next the bandwidth of the signal was increased to 18 kHz, keeping the output power the same. The results are shown in Figure 7.17. As the envelope frequency increases, the errors in the system start to affect the linearization ability more. For example, the memory effects become stronger, the control signal delays are more significant etc. The linearity improvement therefore reduces to 10 dB. However, when compared to Figure 7.5, the improvement in stability due to the new algorithm is clear.

Finally a 50 kHz 32QAM with 17.5 dBm output was linearized. The phase update could not keep up with the signal anymore and had to be shut down. As the phase update is shut down and the bandwidth becomes wider, the memory effects also start to show and result in unbalanced sidebands.

The operation of the linearizer on different power levels and the achieved efficiency improve-

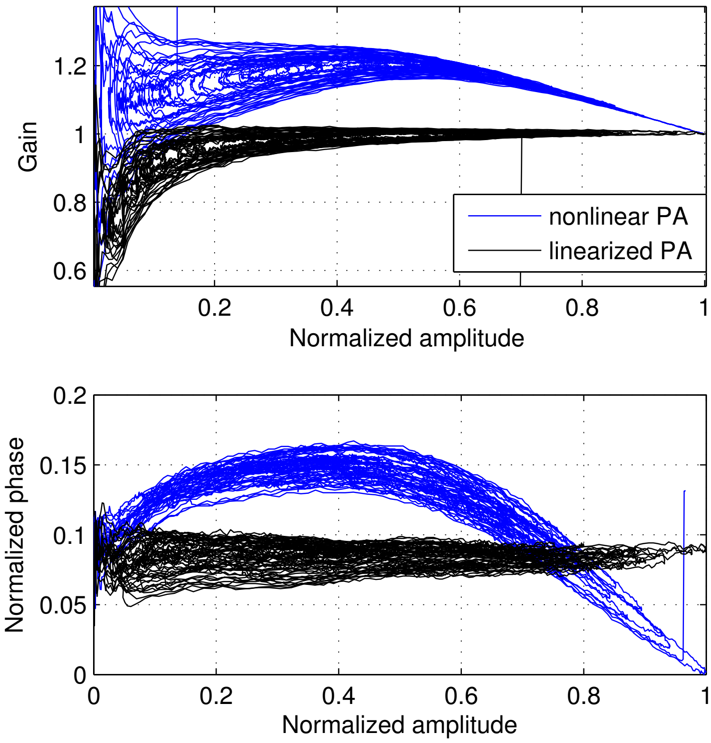


Figure 7.16: The nonlinear and linear phase and amplitude transfer functions with a 500kHz 32QAM signal

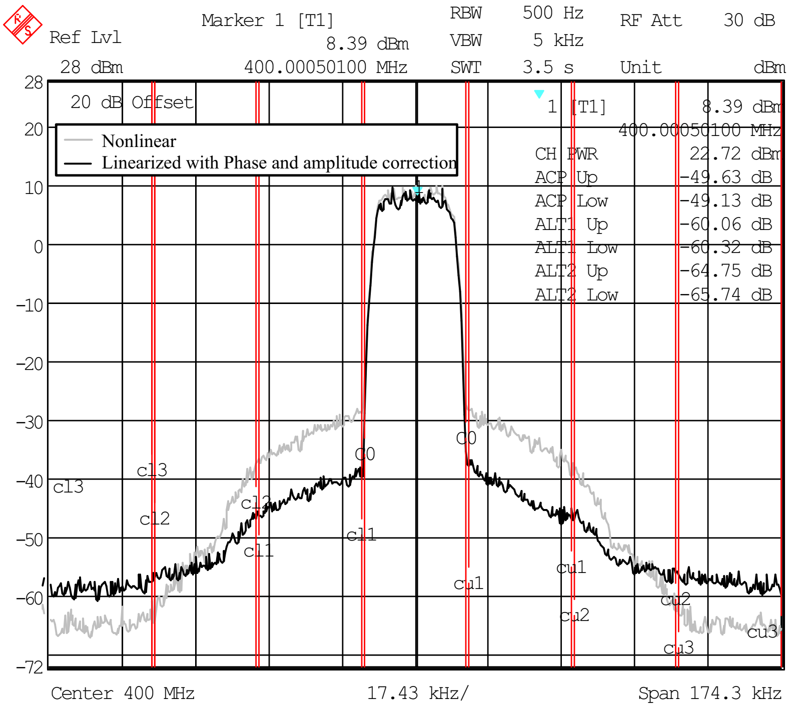


Figure 7.17: Spectrum of a 18kHz 32QAM signal with and without linearization (phase and amplitude linearization)

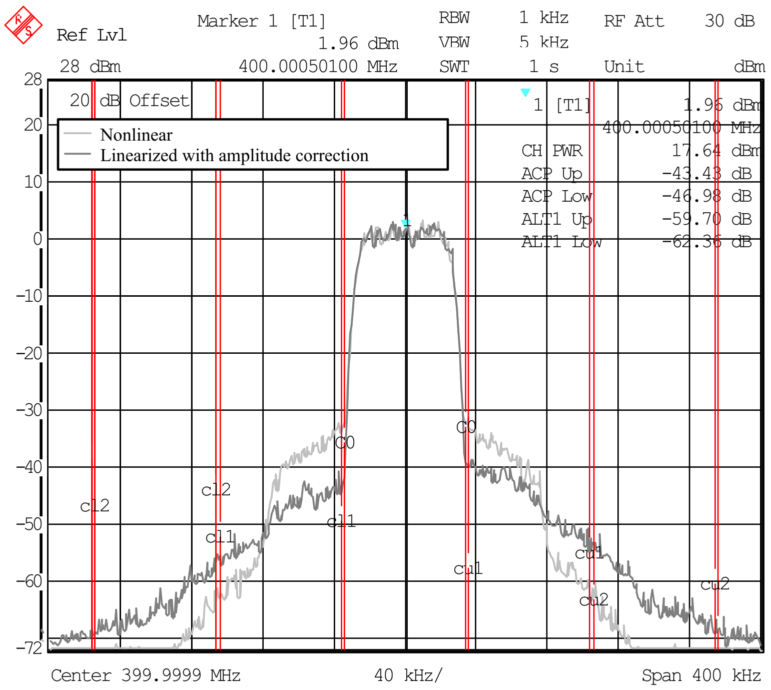


Figure 7.18: Spectrum of a 50kHz 32QAM signal with and without linearization (only amplitude linearization)

ment of the PA was tested using the 18 kHz 32QAM signal optimized to maximum linearity at the 1 dBm input level. The optimization means the adjustment of the detector biases and the D/A and A/D converter analog in put and output voltage ranges to maximize the linearity. Figure 7.19 shows the measured efficiency of the PA with and without linearization. To achieve over 50 dBc ACP the nonlinearized amplifier requires a backoff of 15 dB, which, on the other hand, reduces the PAE to less than 1%. When the linearization is switched on, the 50 dBc ACP is achieved with no backoff and the PAE at 10%. Unfortunately, as the predistorter is optimized at 1dBm power, the linearization ability deteriorates, when the backoff is increased. To compensate for this, the predistorter should be optimized again for a lower power level if this is required.

Figure 7.20 shows the ACP versus efficiency for both the linearized and nonlinear amplifier. The nonlinear amplifier has an ACP plateau between 3% and 18% efficiencies close to 40 dBc. The linearization is able to improve this plateau between 4% and 12% efficiencies to 50 dBc, thus improving the linearity of the system.

7.9 Conclusions

This chapter described the design of the RF-predistortion system that was implemented during this thesis work. The system included a PA chain suitable for a TETRA transmitter and analog phase and amplitude modulators were used as the predistortion elements. The system used both logarithmic and diode-based envelope detectors. The predistortion control and the adaptation algorithm was implemented on an FPGA. The carrier frequency range used in the measurements

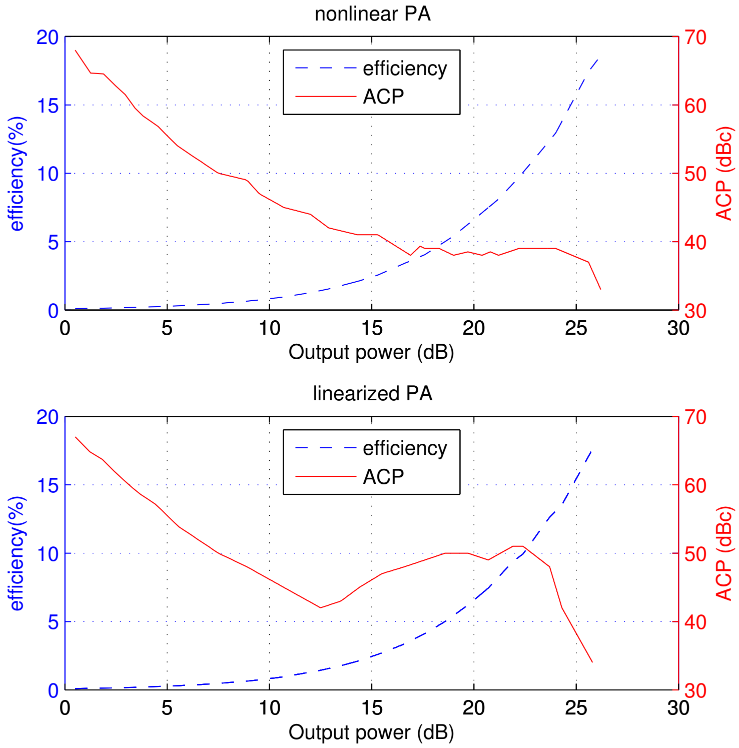


Figure 7.19: The linearized and nonlinearized ACP and efficiency of the measured PA with an 18 kHz 32QAM signal versus output power when the predistorter is optimized for 1 dBm input power

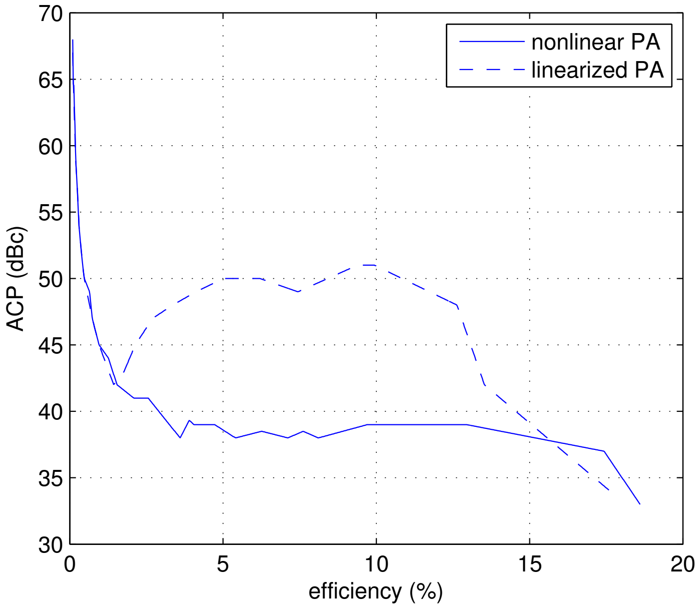


Figure 7.20: The linearized and nonlinearised ACP of the measured PA versus efficiency with an 18kHz 32QAM signal with the predistorter optimized at 1 dBm input power

was from 400 MHz to 420 MHz and the signal bandwidth varied from 3 kHz to 50 kHz.

Measurement results for this system were presented, first without digital prediction. These results proved that the original LUT update algorithm was unstable when used with wide band signals. One contributing factor for this was the control signal delay which caused errors at the measured feedback signals. When the LUT was loaded using a narrow bandwidth teaching signal, the predistorter was able to linearize the 18 kHz bandwidth signal and reduce improve the ACP by 8 dB on the other adjacent channel and 17 dB on the other.

The addition of the digital predictor was able to reduce the instability of the system, but not remove it completely. Therefore a new adaptation algorithm was developed and implemented to the system. The algorithm was based on collecting more measurements for the update and adding a guard period for the update to reduce the risk of contaminating the update by the changing of the LUT value. The algorithm improved the performance of the system and the system was able to achieve a 10 dB reduction in the ACP on both adjacent channels, without the use of separate training signals.

Chapter 8

Baseband predistortion

8.1 Introduction

Another predistorter that was considered for the linearization of the TETRA band transmitter is the baseband predistorter. It was not seen to have the same development potential, being more studied and used solution than RF predistorter. It is a more complex system than the RF predistorter. However, it has some advantages and therefore also a baseband predistortion system, with specification similar to the RF predistorter, was implemented during this thesis work to compare its performance to the RF predistorter. During this implementation, some new issues relating to the baseband predistorters, were studied, such as the nonlinear quadrature modulator errors (section 8.6) and improved frequency domain LUT update methods (section 9.6) related also to the RF predistorters.

Since the baseband signals have much lower frequency than the final RF signal, when using analog predistortion element, it would be easier to implement the predistortion function at the baseband instead of RF [9]. As the modern communication systems usually use baseband signals in quadrature form, the analog implementation of the baseband predistorter would require a predistorter on both of the branches and the two predistorters should be identical which complicates the design. However, as the baseband quadrature signals are usually generated in DSP, it is beneficial to transfer the implementation of the predistorter into the DSP [9, 11, 79, 107]. This approach, however, forfeits some of the benefits of the RF-predistortion. Namely, the fact that RF and baseband circuitry are not any more independent from each other and the baseband predistorter also requires down mixers on the feedback path for the adaptation.

Figure 8.1 presents a basic block diagram of a baseband predistorter. The circuit uses correction values stored into a lookup table or, in some cases, directly as a polynomial function[153] to alter the complex baseband signal in such a way that the PA nonlinearity is compensated. After applying the correction, the signal is digital-to-analog converted, quadrature modulated, possibly up-converted and amplified with the PA.

This chapter presents the design of a complex gain baseband predistorter and compares dif-

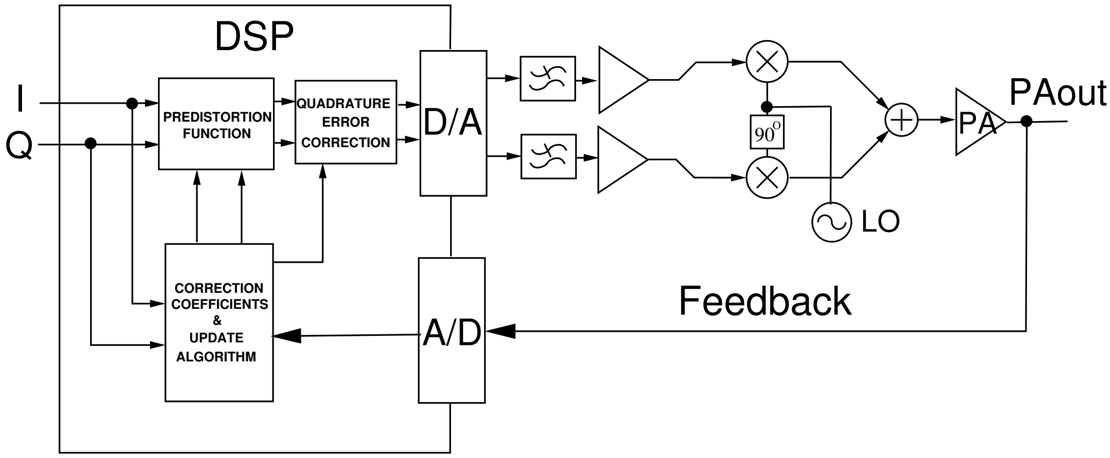

DSP
I
Q
PREDISTORTION FUNCTION
QUADRATURE ERROR CORRECTION
D/A
CORRECTION COEFFICIENTS & UPDATE ALGORITHM
A/D
90°
LO
PA
PAout
Feedback

Figure 8.1: Block diagram of a digital PA predistorter.

ferent quadrature error correction systems required for the system. Also, new results of the effect of quadrature modulator nonlinearity are presented. Finally, simulation and measurement results of the baseband predistortion system are presented.

8.2 Building blocks of the baseband predistorter

An adaptive baseband predistorter consists of two main building blocks: the predistortion function, which defines the maximal linearization ability of the predistorter and the adaptation algorithm, which defines how well the predistorter can follow changes in the PA to be linearized. These blocks will be discussed in this section.

8.2.1 Predistortion Function

Several possibilities to implement the predistortion function of a baseband predistorter have been presented in the literature. The first presented digital baseband predistorters were actually data predistorters [87], which means that the actual baseband data signals are shifted to reduce the distortion in the data constellation. As the predistortion is performed at the symbol frequency instead of the D/A converter sampling frequency the requirements for the digital circuitry are relaxed. However, the problem is that this system corrects only the errors in the received symbols and not the actual distortion on the envelope. This means that only in-band distortion is actively reduced, although the ACP can improve inadvertently.

The idea of a predistorter operating at the D/A output frequency and thus being able to correct the out-of-band distortion was presented by Bateman [90]. The first implementation was presented by Nagata [79]. The system uses a very straightforward method involving a two-dimensional LUT containing the complex predistortion function. The LUT is directly indexed with the complex baseband signal. The predistortion function can be written as

$$v_{pd} = v_{in} + F_{pd}(v_{in}) \tag{8.1}$$

This kind of system is called the mapping predistorter. A block diagram of the mapping predistortion function is shown in Figure 8.2. Although the algorithm is computationally simple, this method has very high hardware costs due to the two-dimensional LUT. This large LUT also converges very slowly to the final value [79] in adaptive implementations. Modifications to the mapping predistorter system to reduce the size of the LUT have been suggested [130] but these modifications also affect the linearity adversely.

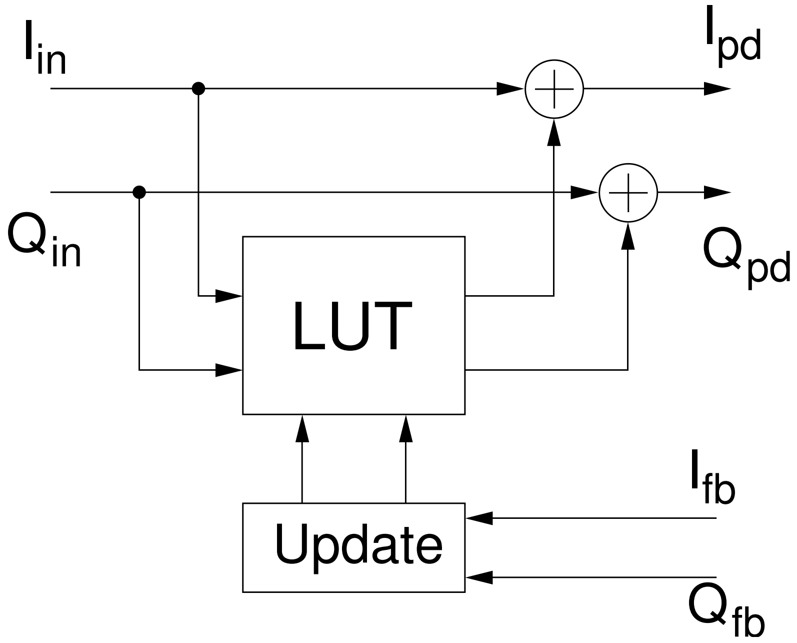


Figure 8.2: Mapping predistortion [79]

However, since the PM-PM and PM-AM effects are negligible in regular power amplifiers, the number of LUT entries can be significantly reduced by changing the indexing so that it depend eds only on the amplitude of the envelope [11] and applying the predistortion function to the original signal as a complex gain function. This reduces the size of the LUT by the square root of the size of the mapping predistorter, thus reducing the hardware costs and the LUT adaptation time. The cost of this is the increased computational complexity, as the amplitude calculation requires at least calculation of the signal power and the generation of the complex gain requires complex multiplication [11]. The predistortion function of the complex gain predistorter can be written as

$$v_{pd} = v_{in} F_{pd} \left(|v_{in}| \right) \tag{8.2}$$

A block diagram of the complex gain predistortion function is shown in Figure 8.3.

The third common baseband predistortion method, the polynomial predistorter, can be seen as a variation of the complex gain predistorter. The relationship of the polynomial baseband predistorter and complex gain predistorter can be seen when (3.3) is written in the following form:

$$v_{pd} = v_{in} \left(\sum_{n=1}^{N} a_{PDn} \left| v_{in} \right|^{n-1} \right) = v_{in} F_{pd} \left(|v_{in}| \right). \tag{8.3}$$

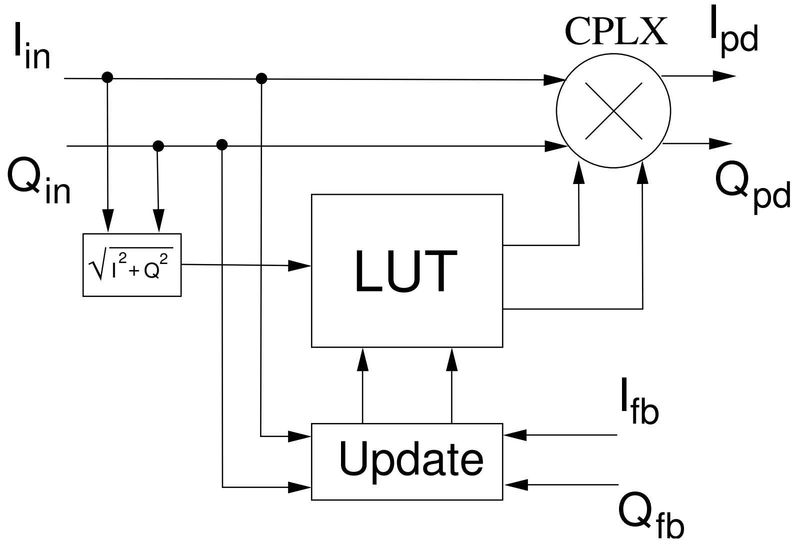


Figure 8.3: Complex gain predistortion[11]

which is the same as (8.2). A block diagram for a third order polynomial predistortion function is shown in Figure 8.4.

The polynomial predistortion function has especially been used in relation to the compensation of memory effects of power amplifiers (Chapter 2.8) [54, 120, 154], although memoryless implementations have been presented as well [155, 156].

The actual calculation of the polynomial every time instant is not very feasible, as the polynomial order increases. Instead, the values of the polynomial corresponding to some predefined input signal values are often calculated beforehand and inserted into a LUT [29, 54]. Thereafter, the operation of the predistorter is exactly the same as for a normal complex gain predistorter, with the exception that the function is limited to polynomial form. The advantage of the polynomial predistorter is that it requires only a low number of parameters, which is beneficial in some adaptive update methods (Chapter 9).

The fourth common baseband predistortion method is the polar predistorter[91, 157]. The polar predistorter requires the baseband signal to be in polar form instead of rectangular form and the predistortion is achieved by altering the phase and amplitude of the signal as follows:

$$v_{pd} = |v_{in}|\, F_A\left(|v_{in}|\right) e^{\arg(|v_{in}|)+F_\phi(|v_{in}|)}. \tag{8.4}$$

This is closely reminiscent of the operation of the phase-amplitude modulator based RF predistorter (Section 4.4.1). Block diagram of the complex gain predistortion function is shown in Figure 8.5. The polar predistorter allows the inspection of the phase and amplitude errors of the PA separately, which is useful in behavioral studies [91], but requires the transformation to polar form and back during the predistortion, which is very costly hardware-wise [91]. If the predistorter is implemented adaptively, another rectangular to polar converter is required for the feedback. However, if the baseband signal is originally in polar form, as is the case with EE&R transmitters, the polar predistorter becomes a significantly more efficient method than the com-

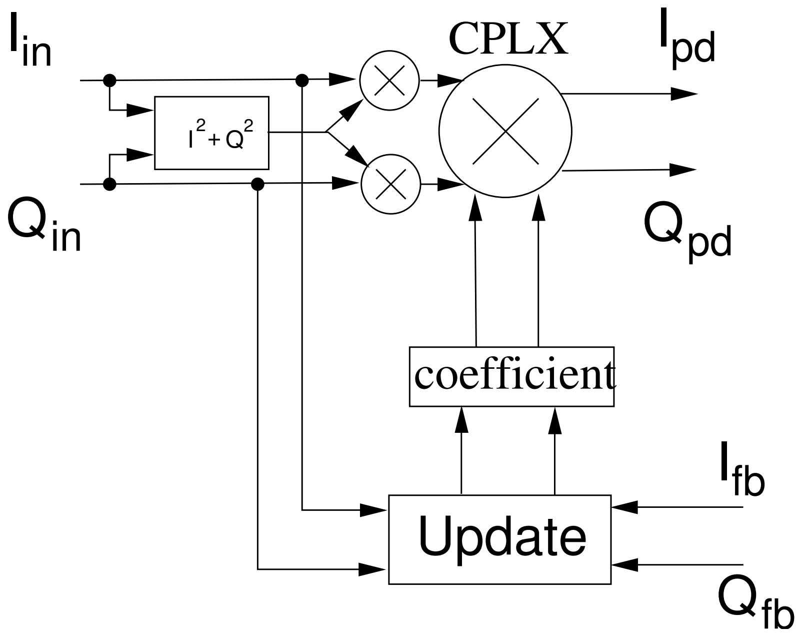


Figure 8.4: Third order polynomial predistortion function implemented using multipliers

plex gain predistorter [91].

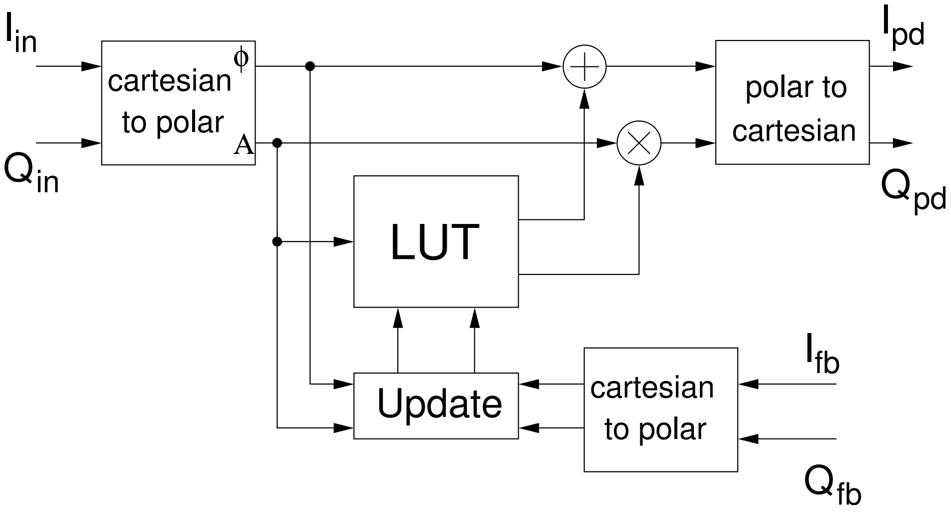


Figure 8.5: Polar predistortion[91]

8.2.2 The adaptation

As the power amplifier transfer function changes with temperature and age and from device to device, the linearization ability achieved by constant correction is limited. As is the case with the RF-predistortion, the baseband predistortion function can also be implemented adaptively.

Just as with the RF-predistorter, there are two main methods to implement the feedback: frequency and time-domain sampling.

The frequency-domain sampling can be implemented with similar ACP feedback as presented for RF-predistortion in figure 4.6. The ACP feedback allows the implementation to be performed with narrow bandwidth, as only the average distortion power level has to be known [56, 105]. The ACP feedback is suitable for predistortion systems using, for example the Nelder-Mead algorithm [158], genetic algorithm [159] or simulated annealing [160] for the LUT update. These use an aggregate measure of error for calculation of the update.

The time-domain feedback is the most common feedback method [11, 54, 79, 161–163] for baseband predistorters. It allows for several different adaptation methods, which can be further divided into those that update only one LUT entry at a time and those that update the whole LUT at once.

The update of one LUT entry at a time is usually computationally less complex and does not restrict the shape of the predistortion function. As examples of the update methods for single entry update, Nagata et al. use the modified linear update method (Equation 3.12) [79], Cavers et al. use the secant method (Equation 3.13) [11] and the basic linear update method (Equation 3.11) [108]. As is the case with the RF-predistortion systems, the time-domain update requires good delay matching between the system input and output signals. This is discussed more thoroughly in Section 5.3 and in references [79, 115].

The update of the whole LUT at once requires much more computation power than the update of a single LUT entry and the complexity increases rapidly as the number of the LUT entries increases. However, it is more suitable for compensation of PA memory effects which require several codependent functions to be updated simultaneously. Usually, this update method is used in conjunction with polynomial LUTs to reduce the number of parameters [56, 105, 120, 121, 124, 153, 154]. Common update methods are RLS [120, 124, 125, 154] and LMS [121, 125, 153]. Also a genetic algorithm has been proposed for both memoryless predistorters and memory predistorters [56, 105]. The polynomial function, however, limits the linearization ability of the predistorter. Chapter 9 discusses a possibility to reduce the number of parameters with an interpolation function.

8.3 Linear quadrature modulator and demodulator errors

The quadrature modulator and demodulator required on the forward and feedback paths of the baseband predistorter are significant error sources in baseband predistortion systems [164]. Although linear quadrature modulator and demodulator errors cause linear transformation of the modulated signal, by themselves they do not cause spectral spreading of the signal [164]. However, when combined with a nonlinear amplifier, new phase dependent spectral components are generated.

Common errors are DC offsets, which cause the signal constellation to shift, phase offsets, which cause the I and Q channels to feed through to each other and squash the constellation in the direction of 45^o from I or Q axis and gain errors, which squash the constellation in the direction of I or Q axis [165]. Figure 8.6 depicts the effect of the quadrature modulator errors on a 16QAM signal constellation.

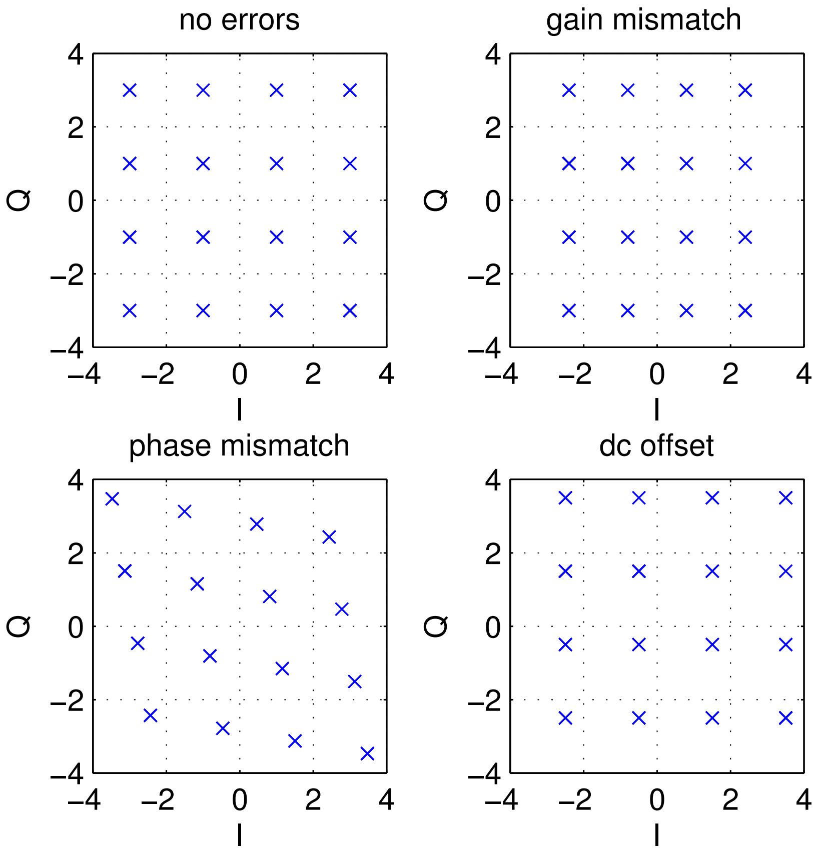


Figure 8.6: The effect of quadrature modulator errors on 16QAM signal constellation

The linear quadrature errors can be written in the following matrix form: [165, 166]

$$\begin{bmatrix} I_{qm} \\ Q_{qm} \end{bmatrix} = M \begin{bmatrix} I \\ Q \end{bmatrix} + \begin{bmatrix} dI \\ dQ \end{bmatrix} = \begin{bmatrix} a\cos(\phi/2) & b\sin(\phi/2) \\ a\sin(\phi/2) & b\cos(\phi/2) \end{bmatrix} \begin{bmatrix} v_I \\ v_Q \end{bmatrix} + \begin{bmatrix} dI \\ dQ \end{bmatrix}, \tag{8.5}$$

where a and b are the gains of the I and Q channels respectively, ϕ is the phase error and dI and dQ are the I and Q channel offsets. Using this format eases the development of a compensation method for the modulator errors using inverse matrices.

Another possibility is to use a complex conjugate format of the modulated signal [116].

$$v_{qm} = Av_{pd} + Bv_{pd}^{*} + C, \tag{8.6}$$

where A, B and C are coefficients depending on the phase, gain and DC errors, as presented by Cavers [116]. The conjugate format allows for simpler calculations than the matrix format as the calculations can be performed directly with complex numbers without having to resort to matrices.

All the linear QM errors depend on both the phase and the amplitude of the baseband signal (or the individual I and Q signals), i.e. they cannot be expressed in the form of

$$v_{qm} = H(|v_{pd}|). \tag{8.7}$$

The mapping predistortion is fairly insensitive to the QM errors due to its two-dimensional indexing, which makes it possible to have separate correction values for each I and Q signal combination [130].

However, due to the reduced hardware costs of the method, most baseband predistortion systems implement the linearization based on the amplitude of the signal. Unfortunately, amplitude-based predistorters, such as the complex gain predistorter, are vulnerable to the quadrature demodulator errors [91, 116, 164]. They are able to compensate only errors of the form of (8.7) and the linearization ability is significantly reduced due to QM errors.

Figure 8.7 shows the spectrum of a quadrature sine wave,

$$f(t) = \sin(\omega t) + i\,\cos(\omega t), \tag{8.8}$$

when driven through a non-ideal quadrature modulator. The DC offsets cause an unwanted carrier signal to emerge and the phase and gain offsets generate an unwanted image signal on the opposite side of the carrier to the wanted signal. All of these signals cause misadjustment to the predistorter adaptation. The figure shows shows second and third order harmonic components caused by independent nonlinearities in the I and Q channels. The effect of these nonlinear components on the predistortion systems will be discussed more thoroughly in Section 8.6.

In adaptive predistortion systems the errors in the quadrature demodulators also cause error in the adaptation [167]. Usually, compensation circuits for the quadrature modulator and demodulator errors are implemented in amplitude-based baseband predistorters [165].

The effects of quadrature modulator and demodulator errors on nonlinear linearized power amplifiers are discussed in more detail in references [116, 164, 167].

8.4 The predistorter architecture selected for implementation

It was decided to concentrate in this work on the complex gain predistorter, (Equation 8.2), due to its fairly low hardware requirements and relative simplicity. Also a hardware implementation of the complex gain predistorter was achieved. This made it necessary to also implement a quadrature error correction circuit, which will be discussed later in this chapter.

As is the case with RF-predistortion systems (Chapter 6), the LUT addressing significantly affects the efficiency and complexity of the baseband predistortion. However, unlike in RF-predistortion systems, all the indexing methods are usually based on indexing by power. For example, the amplitude of a complex signal is the square root of the instantaneous power. Thus, indexing by amplitude requires calculation of the power and calculation of the square root. There-

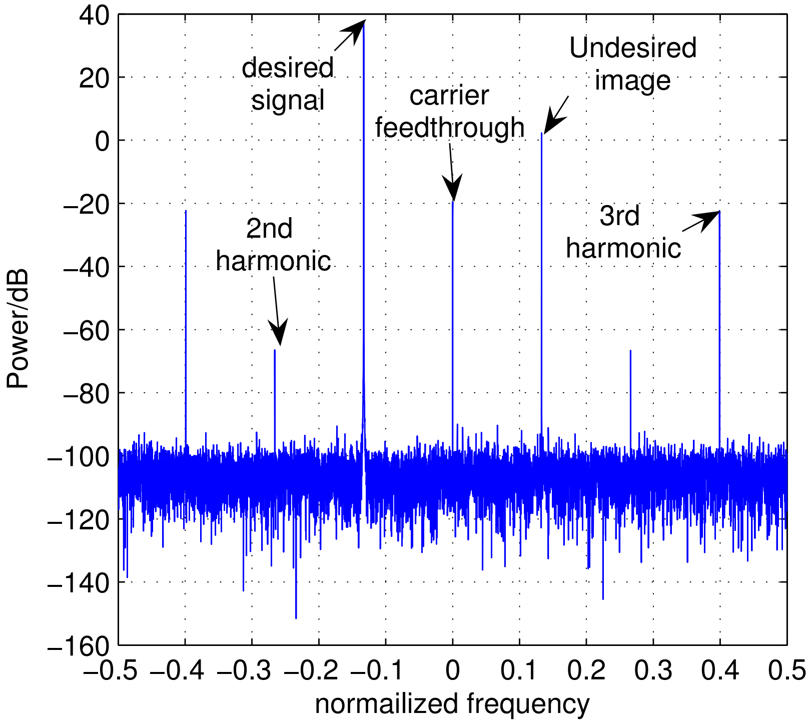


Figure 8.7: The effects of the quadrature modulator nonidealities on a quadrature sine wave signal, visualized in frequency-domain

fore, using any indexing method other than power indexing directly increases the complexity of the system. This relation is much more clear in baseband predistortion systems than in RF-predistortion systems.

Unlike in a mapping predistorter, which can use the complex baseband signal to directly index the LUT [79], in amplitude-based predistorters, an indexing signal related to the real amplitude of the complex baseband signal has to be calculated. The simplest way to implement the addressing is to use the squared absolute value or power of the complex baseband signal. This is also the basis for other LUT addressing methods. The squared absolute value is calculated with

$$v_p = real(v_{in})^2 + imag(v_{in})^2 \tag{8.9}$$

which means that two multipliers and an adder are required. This method, however, over-emphasizes the large amplitudes and affects the linearization ability adversely [108]. This is similar to the power detector arrangement in the RF-predistortion systems (Chapter 6).

Several methods have been presented to change the indexing so that the emphasis of the amplitude values can be changed to a more efficient one, such as direct amplitude indexing and indexing with a logarithmic function [78, 108–110, 168]. All of these methods, however, use (8.9) as the basis function and alter the result to calculate the new address. This means that they are actually methods for only rearranging the LUT entries. These methods shall be discussed in more detail in Chapter 9. However, since the use of the non-squared envelope significantly improves the operation of a baseband predistorter [108] and is a fairly commonly used method [107, 169–171], and since it was used in the implemented baseband predistorter, it will be discussed in this chapter.

The use of amplitude-based indexing requires the calculation of the square root of (8.9). The calculation of the square root with a high degree of accuracy is a complicated operation [172] that usually requires several iterations, so it is not feasible. However, the square root can be approximated with sufficient accuracy using small ROMs with an interpolated output [107, 169]. In the implemented predistorter, only one ROM was used in the implementation instead of a square root ROM and a square root difference ROM, due to availability of dual-port ROMs.

The implementation is presented in Figure 8.8 and operates as follows: The 16-bit input word is divided into an “integer” part, which consists of the six most significant bits of the input word, and into a “fractional” part, which consists of the 10 least significant bits of the input word. The integer part is used to index a ROM that contains the 64 square root values of the 6-bit input signal and the outputs of the ROM are the square root of the integer part and the next larger value in the ROM. The difference of these values is multiplied with the fractional part and added to the smaller ROM value to interpolate the output.

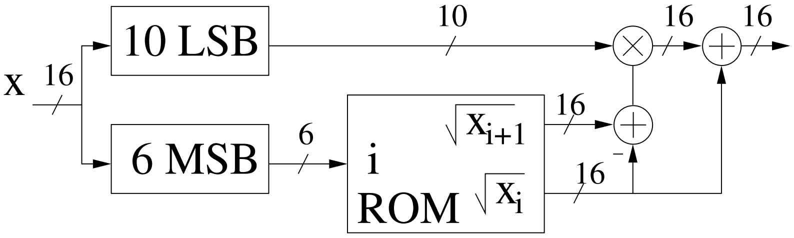
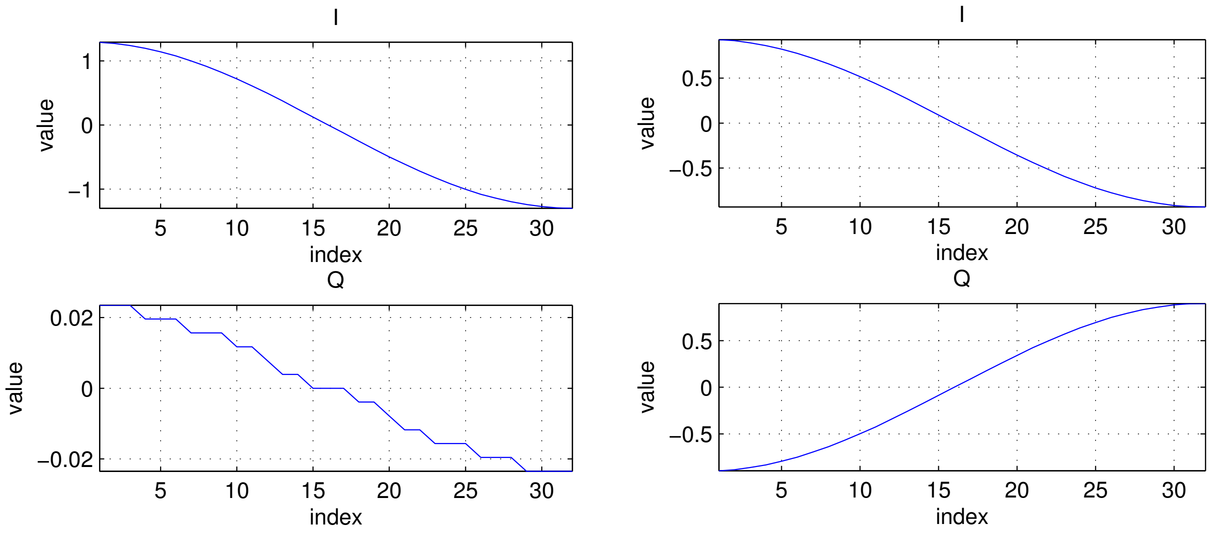


Figure 8.8: Square root algorithm.[107, 169]

It was noted that, since the phase distortion in the PA is usually fairly small, the LUT containing the imaginary part of the predistortion function contains only very small values and most of the value range of the quadrature LUT is unused. By adding a phase rotation at the predistorter output it is possible to create a constant phase rotation to the predistortion LUTs also. Part of the real part of the predistortion function can be moved into the imaginary LUT. This means that the unused value space in the quadrature LUT can be used to store part of the in-phase distortion. This makes it possible to have gain larger than one without clipping the predistortion function. Figure 8.9a shows an example of a 32-entry LUT quantized with 8 bits; the phase error is 1^o. As can be seen, the original in-phase LUT has entries with values larger than one and these values would be clipped to one when quantized. Figure 8.9b shows the same function rotated by 45^o. Now the function is more evenly distributed between the LUTs and the maximum value is reduced to one, thus there is no clipping although the gain of the signal is still more than one.

The block diagram of the basic complex gain predistorter algorithm with amplitude indexing and linear LUT update, (Equation 3.11), is shown in Figure 8.10. Block A calculates the absolute value of the complex signal, Block B implements (8.2), Block C does the rotation by 45^o, Block D calculates the new LUT value with the linear update and Block E stores the values to the LUTs.



(a) LUT entries before rotation

(b) LUT entries after rotation

Figure 8.9: The effect of the rotation of the predistorted signal on the LUT entry distribution

Lower case letters indicate signals that originate in one part of the diagram and terminate at the other. μ is the linear adaptation convergence factor, *Iin* and *Qin* are the input signals, *Ifb* and *Qfb* the feedback signals and *Iout* and *Qout* the output signals. CPLX MULT blocks implement a complex multiplication and the CPLX DIV divides a real number by a complex number.

It was decided to implement an interpolation scheme at the LUT output to improve the accuracy of the predistortion function. The different interpolation methods and their effects are discussed more thoroughly in Section 9.4.3. Due to its simplicity, the linear interpolation scheme was chosen. The linear interpolation scheme can be described with the formulas [91]

$$LUT_{intp}(n) = LUT_k(n) + frac(2^{bit}\,|V_{in}(n)|) \cdot (LUT_{k+1}(n) - LUT_k(n)) \tag{8.10}$$

and

$$V_{pred}(n) = V_{in}(n) \cdot LUT_{intp}(n). \tag{8.11}$$

where $V_{in}(n)$ is the original complex input signal normalized to have absolute value between 0 and 1, $LUT_k(n)$ is the k^{th} value stored in the LUT, $k = int(2^{bit}\,|V_{in}(n)|)$, int function returns the integer part of a number, $frac$ returns the fractional part, bit is the LUT address word length, n the time instant and $V_{pred}(n)$ the predistorted signal.

There is one thing that should be noted when using an interpolation scheme with an adaptive LUT: the predistortion signal consists of a weighted sum of two or more LUT entries, thus the LUT update value should be “deinterpolated” between two or more LUT entries to take into account their different weights [91]. If the linear update method (Equation 3.11) is used the LUT update with deinterpolation becomes [91]

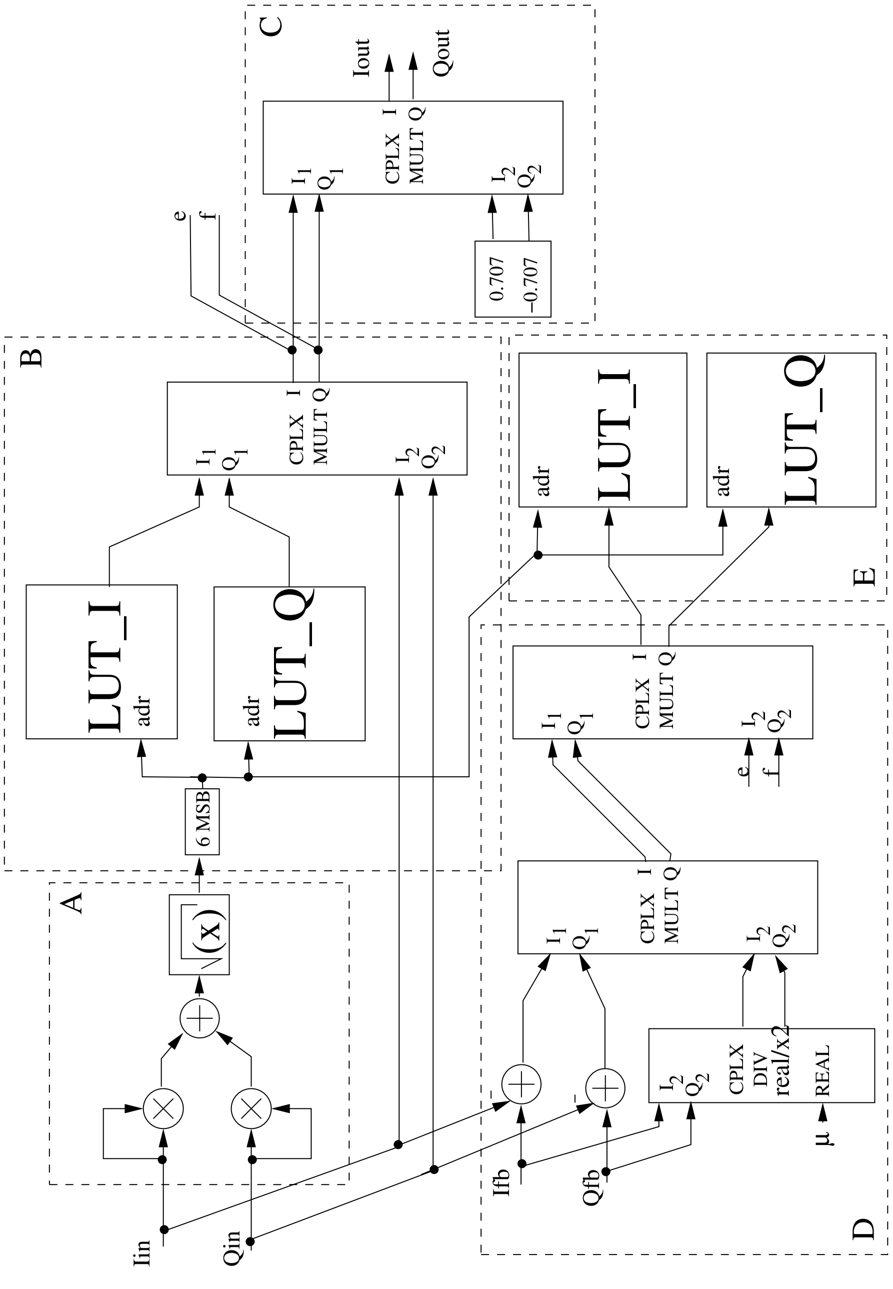


Figure 8.10: Block diagram of a complex gain predistorter with linear update

$$\begin{gathered} LUT_k(n+1) = LUT_k(n) + \\ (1 - frac(2^{bit}\,|V_{in}(n)|)) \cdot a\frac{(V_{out}(n) - V_{in}(n))}{V_{out}(n)} LUT_{intp}(n) \end{gathered} \tag{8.12}$$

and

$$\begin{gathered} LUT_{k+1}(n+1) = LUT_{k+1}(n) + \\ frac(2^{bit}\,|V_{in}(n)|) \cdot a\frac{(V_{out}(n) - V_{in}(n))}{V_{out}(n)} LUT_{intp}(n). \end{gathered} \tag{8.13}$$

where the notation is the same as in (8.10), (8.11) and (3.11).

The block diagram of the predistortion algorithm with linear interpolation and deinterpolation is shown in Figure 8.11. Blocks from A to E are the same as in Figure 8.10, Block F implements the interpolation and Block G implements the deinterpolation. CPLX REAL MULT multiplies a complex number with a real number. The other symbols are the same as in Figure 8.11.

8.5 Correction of linear quadrature modulator and demodulator errors.

If an analog quadrature modulator and demodulator are used in a baseband predistortion system, a quadrature error compensation circuit is almost a necessity [116, 164]. The following section discusses the implementation of the quadrature error correction circuit used in the design and methods for adaptive update of the correction function.

8.5.1 Correction of the linear quadrature errors

Most straightforward way to derive the quadrature error correction function is to generate the inverse formula of (8.5). Thus, the compensation can be performed by multiplying the predistorted baseband signal with the inverse of the matrix M in (8.5) and subtracting a DC compensation coefficient [173]:

$$\begin{bmatrix} I \\ Q \end{bmatrix} = \boldsymbol{C} \begin{bmatrix} I_{pd} \\ Q_{pd} \end{bmatrix} - \boldsymbol{C} \begin{bmatrix} dI \\ dQ \end{bmatrix}, \tag{8.14}$$

where I_{pd} and Q_{pd} are the predistorted baseband signals. The DC correction coefficient, $\boldsymbol{C} \begin{bmatrix} dI \\ dQ \end{bmatrix}$, can be written as a constant vector $\begin{bmatrix} dI_{pd} \\ dQ_{pd} \end{bmatrix}$ and as the multiplication matrix $\boldsymbol{C}$ we get:

$$C = M^{-1} = \frac{1}{\cos^2(\phi)\mathrm{a}b^2} \begin{bmatrix} 1 & -\tan(\frac{\phi}{2}) \\ -a/b\ \tan(\frac{o}{2}) & a/b \end{bmatrix}. \tag{8.15}$$

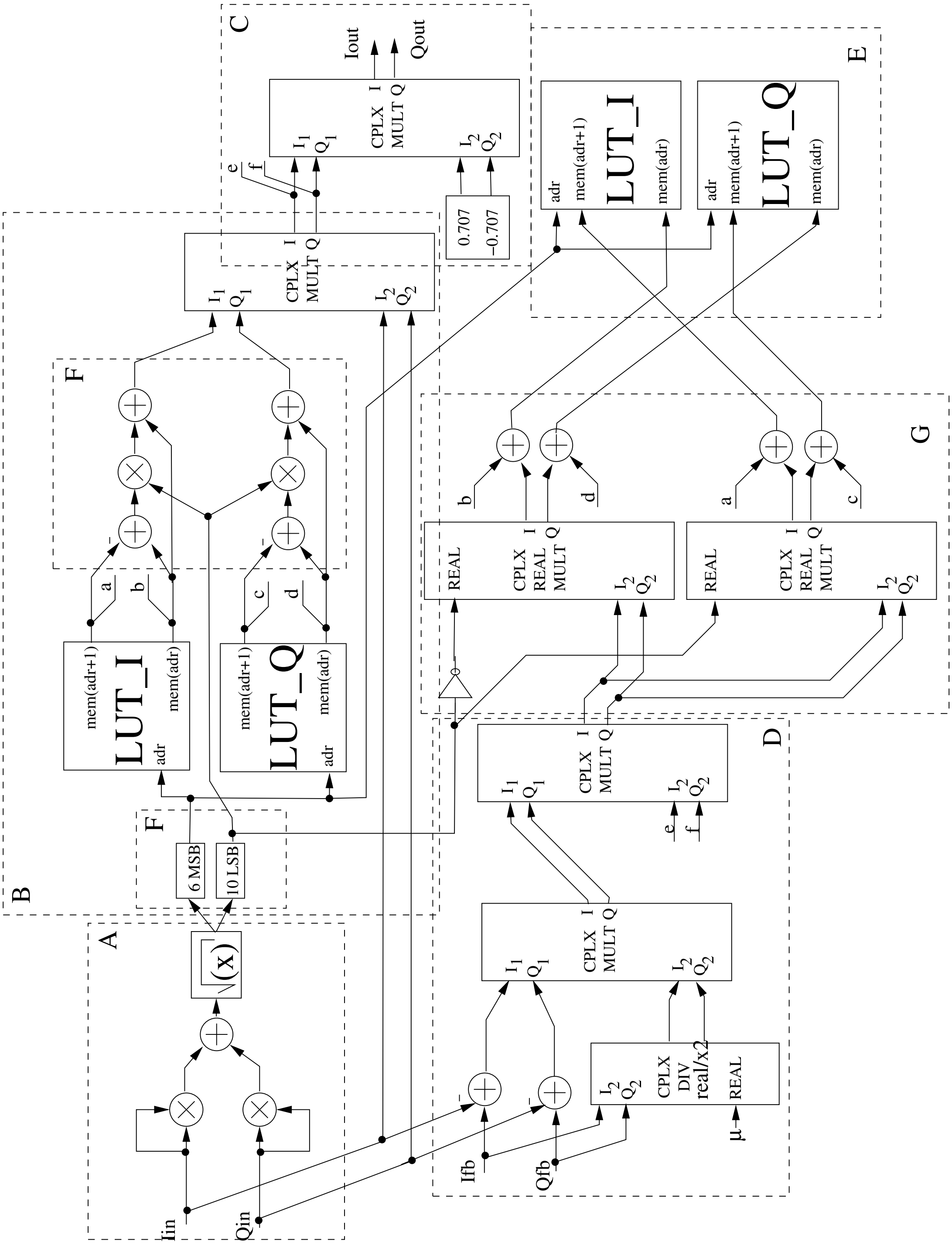


Figure 8.11: Block diagram of a complex gain predistorter with interpolation and linear update.

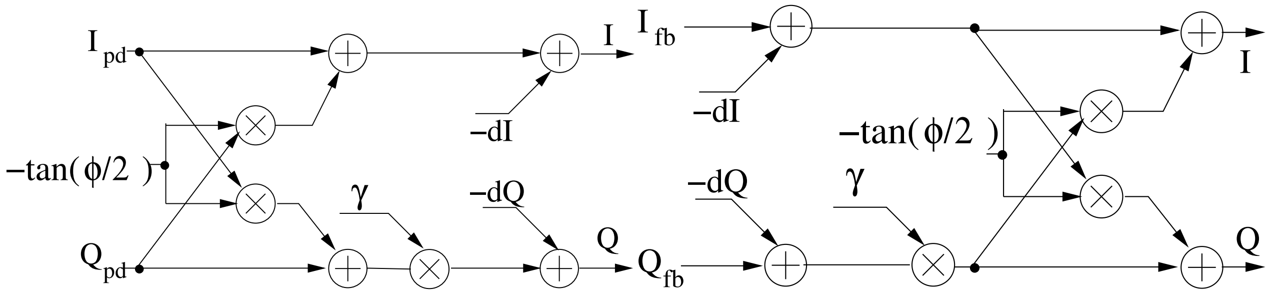
Because the term $\frac{1}{\cos^2(\phi)ab^2}$ corresponds to constant gain, it can be compensated with a complex gain predistorter and thus can be omitted. Now the C matrix can be further simplified to

$$C = \begin{bmatrix} 1 & -\tan(\frac{\phi}{2}) \\ -\gamma\ \tan(\frac{\phi}{2}) & \gamma \end{bmatrix}, \tag{8.16}$$

where $\gamma = a/b$. Thus the final quadrature compensation formula is

$$\begin{bmatrix} I \\ Q \end{bmatrix} = \begin{bmatrix} 1 & -\tan(\frac{\phi}{2}) \\ -\gamma\ \tan(\frac{\phi}{2}) & \gamma \end{bmatrix} \begin{bmatrix} I \\ Q \end{bmatrix} + \begin{bmatrix} dI_{pd} \\ dQ_{pd} \end{bmatrix}. \tag{8.17}$$

This matrix can be implemented with a simple digital circuit [20]. However, Faulkner et al. [20] use $\boldsymbol{C}$ multiplied by b, thus having gain correction multipliers in both I and Q paths. Figure 8.12 shows the implementation of (8.17) using a circuit similar to one presented by Faulkner et al. [20], but with the gain multiplier only on Q branch. The circuit elements in the quadrature modulator correction are ordered in such way that the DC errors are compensated last, so that they do not interfere with gain and phase error correction and the gain errors are compensated second so as not to interfere with the phase error correction [20]. The demodulator correction circuit is ordered in the opposite way for the same reason. The implementations of both circuits use four adders and three multipliers.



(a) Quadrature modulator correction

(b) Quadrature demodulator correction

Figure 8.12: Quadrature modulator and demodulator correction

Cavers et al. [165] have shown that, by assigning the phase error correction to only either I or Q branch, the circuit requires only two multipliers and three adders. This is results from the fact that $\boldsymbol{C}$ can be written as

$$\begin{bmatrix} 1 & -\tan(\phi/2) \\ -\gamma\ \tan(\phi/2) & \gamma \end{bmatrix} = \begin{bmatrix} \cos\phi & \frac{\sin\phi\cos\phi}{1+\cos\phi} \\ -\frac{\sin\phi\cos\phi}{1+\cos\phi} & \cos\phi \end{bmatrix} \begin{bmatrix} 1 & -\tan(\phi) \\ 0 & \frac{\gamma}{\cos\phi} \end{bmatrix}, \tag{8.18}$$

where the first matrix on the right side of the equality corresponds to a constant phase rotation and thus it can be compensated by the predistorter. Now, we can use as the compensation matrix

$$\boldsymbol{C} = \begin{bmatrix} 1 & -\tan(\phi) \\ 0 & \frac{\gamma}{\cos\phi} \end{bmatrix}. \tag{8.19}$$

However, it was noticed during simulations that the quadrature error correction without the removal of the constant phase shift (8.17) seemed to have better convergence properties than the reduced version, and thus was selected for the implementation.

Usually, the coefficients of $\boldsymbol{C}$ are found through an iterative process. Several methods for iteration have been suggested, some using a training signal to find the coefficients [20, 165, 173] and others using the actual transmitted signal [161, 173–175]. The use of a training signal often enables simpler hardware implementation than the actual-signal-based methods but it also requires stopping the transmission during the adaptation of the coefficients.

During the development of the baseband predistortion circuit presented in this thesis, four different adaptive methods were compared as a way to find the quadrature error correction coefficients. Two of the methods were time-domain envelope-based and two based on spectral measurements. All but one method use training signals to find the coefficients and thus coefficient update during the actual transmission is not possible in these cases. The methods are described in the following paragraphs.

The first method: This method was presented by Faulkner et al. [20] and uses different test signals for DC offset, phase mismatch, and gain-mismatch compensation. The output of the quadrature modulator is measured using an envelope detector. The adaptation consists of three phases. First the I and Q channels are set to zero and the DC correction values are adjusted until the measured envelope is minimized. Next, the gain compensation is performed by first setting the I channel to an arbitrary value, A, and Q channel to 0 and then I channel to 0 and Q channel to A. The measured envelopes are compared and the gain coefficients adjusted to minimize the difference. This is repeated and the gain correction adjusted until the measured envelopes in both cases are the same. Finally the phase compensation is done by alternating the I and Q channel values between (A, A) and $(A, -A)$. The phase coefficient is then adjusted until the measured envelopes in both cases are the same.

The second method: In this method, we use a quadrature sine wave (Equation 8.8) as the input signal and measure the power of the carrier leakage and the image signal shown in Figure 8.7. The coefficients can be found using, for example a modified binary search algorithm to minimize the measured powers. The I and Q channel DC values have to be adjusted together and the amplitude and phase have to be adjusted together as their effects are intertwined.

The third method: This method can be used to adapt the quadrature demodulator. It uses the fed back quadrature data to adjust correction values. The adaptation can be performed using either of the previous two methods, i.e. the absolute value of the complex feedback signal or the spectral components of the feedback signal can be calculated.

The fourth method: This method uses measurements of the adjacent channel power of the PA output signal when the predistorter is in operation. The coefficients are updated to minimize the adjacent channel power (ACP) using the modified binary search. However, now all of the coefficients have to be adjusted simultaneously as they all affect the ACP. The advantage of this method is that the adjustment can be made during the actual operation of the circuit. However, either high-order filters to separate the adjacent channel powers or heavy signal-processing operations are required.

8.5.2 Simulations

The complex gain predistortion system presented in Section 8.4 was simulated with Matlab using a full-scale 16QAM 18 ksym/s signal as the input. a 6 bit LUT address and 16 bit word length was used in the simulations. The sampling frequency was 1 MHz. The PA was modeled with the model PA2 from Section 2.7. A third order Butterworth filter with the knee frequency of 300 kHz was used at the predistorter output and feedback input.

The quadrature compensation circuit described in Section 8.5 was also used in the simulations and the first, second and third methods in Section 8.5 were used for the adaptation. 3.4^o phase offset and 4% gain mismatch between the I and Q branches and a DC offset of 3% of the maximum signal amplitude were used to model the quadrature modulator or demodulator nonidealities. First the adaptation of the predistorter was turned off to study only the effect of the quadrature modulator error compensation circuitry. The simulated corrected quadrature errors for first and second method are listed in Table 8.1, the third method is not listed as it can not compensate the quadrature modulator errors and the fourth method was tested only by measurements due to long simulation times. The table shows the gain error in percents between the I and Q channels, phase error in degrees between the channels and the absolute value of the DC offset in percents of the amplitude of the signal. Both methods are able to reduce the modulator errors significantly and the second method can reduce the gain and phase errors to half of the remaining errors using the first method.

Table 8.1: Quadrature errors with different coefficient finding methods (see Section 8.5).

Method	Gain Error	Phase Error	DC Offset
No quadrature correction	4.0%	3.4^o	3.0%
first method(simulated)	0.12%	0.7^o	0.45%
second method(simulated)	0.05%	0.6^o	0.2%

Figure 8.13 shows the original, corrected and uncorrected constellations of the 16QAM signal. The adaptation used was the first method in Section 8.5. After the correction, the constellation coincides with the original constellation.

As the quadrature modulator errors affect the linearization ability of the predistorter, the spectra of the signal at the output of the amplifier with and without predistortion and with and without quadrature compensation were also examined. The adaptation was now in use and thus quadra-

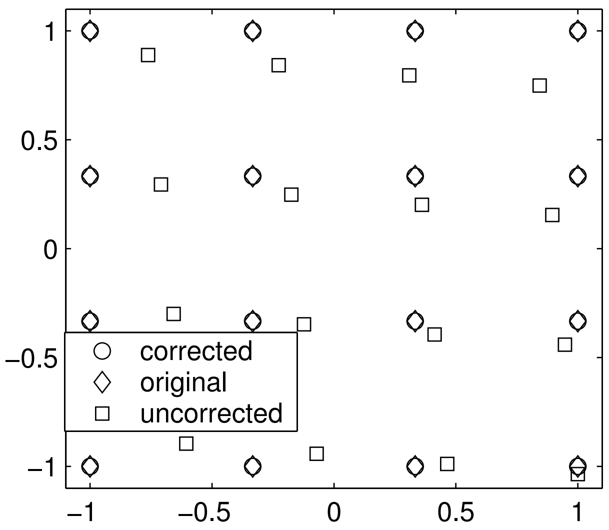


Figure 8.13: Signal constellation with and without quadrature compensation

ture demodulator errors were also added to the system. The demodulator used the same phase, gain and DC error values as the modulator. The third method was used for the QDM compensation. Figure 8.14 shows the spectra of the 16QAM signal when no linearization is in use, when no modulator or demodulator errors are corrected, and when only quadrature modulator errors are corrected and when both the quadrature modulator and demodulator errors are corrected. The adaptation algorithm used for the quadrature modulator coefficient adaptation was the first method. What can be seen is that without the modulator error correction the result is much worse than without linearization; the noise floor rises and there is no improvement in the ACP. Addition of the quadrature modulator error correction improves the ACP somewhat, but the errors in the demodulator still drive the adaptive algorithm into an unstable state and the noise floor remains high. When both the quadrature modulator and demodulator errors are corrected, the adaptation of the predistortion LUTs starts to work correctly and the ACP improves over 10 dB as the approximation of the inverse nonlinearity gets closer to the optimal.

The nonlinearised first ACP was 37 dBc. Simulated ACPs using the first, second and third QM and QDM error compensation methods are shown in Table 8.2. The third method was tested separately to investigate how the ACP improves when only the quadrature demodulator errors were corrected. After linearization, the first ACP improved to 58 dBc when the QM errors were compensated using the first adaptation method. An interesting result is that the second method gave worse results than the first regardless of the better performance when measured in gain, phase and DC offsets. This was assumed to be due to the fact that the use of the DC training signals gave better results in the quadrature demodulator adaptation than the sine training signals when the correction was finally applied to a 16QAM signal. This assumption was supported by the measurements presented in Section 8.8.

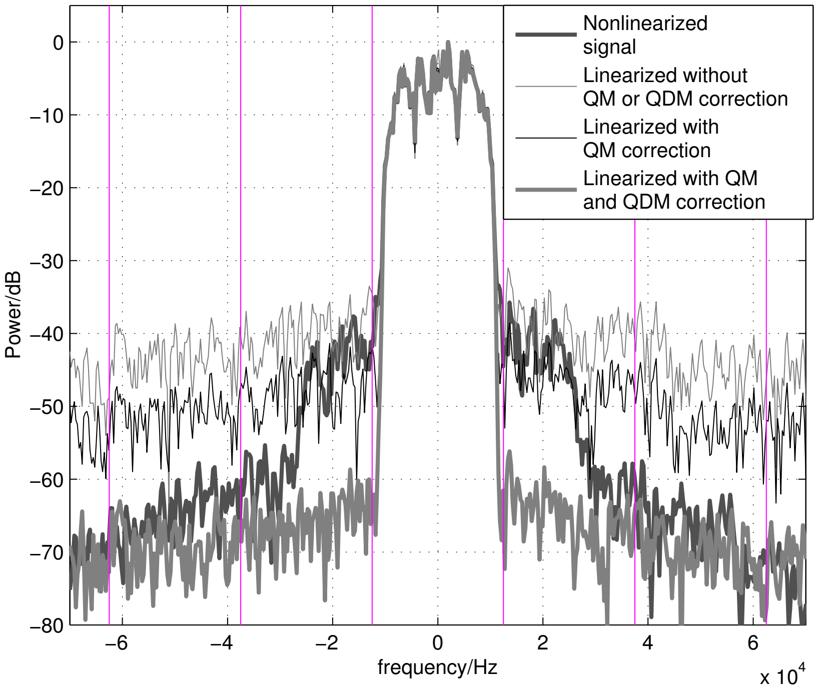


Figure 8.14: Simulated linearized and not linearized spectra.

Table 8.2: Simulated first and second ACP using different quadrature correction methods (see section 8.5).

Method	first ACP	second ACP
No predistortion (simulated)	37 dBc	57 dBc
predistortion with the first method (simulated)	58 dBc	63 dBc
predistortion with the second method (simulated)	55 dBc	58 dBc
predistortion with the third method (simulated)	44 dBc	50 dBc

8.6 Nonlinear quadrature modulator errors

In addition to the linear errors described in Section 8.3, quadrature modulators (and the amplifiers and converters on the baseband path) exhibit nonlinear distortion also and, as the linear quadrature modulator error correction increases the signal levels, the nonlinearities are stronger than before the correction[176]. Figure 8.15 shows some of the nonlinearity sources in a baseband predistortion system.

There are several sources of errors that generate nonlinearities that an amplitude-based predistorter is not able to compensate, such as even-order harmonics and phase dependent distortion.

Firstly, the DC offsets in the baseband signals give rise to even-order harmonics in the spectrum [176]. These offsets can actually be caused by the linear quadrature error compensation as the DC offset compensation values offset the baseband signals before they compensate the DC offset in the modulator [176]. This can be analyzed as follows. If the distortion is assumed to be

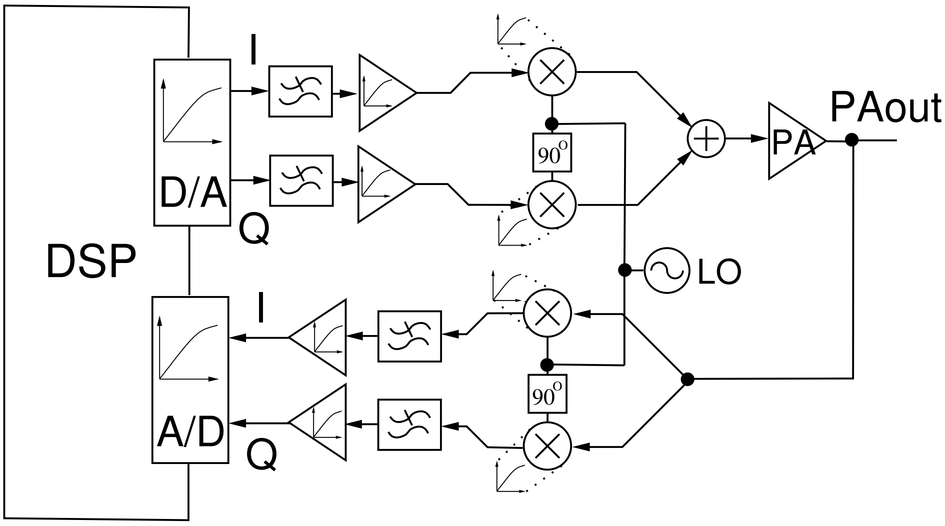


Figure 8.15: Sources of nonlinearity in a baseband predistortion system

purely third order,

$$v_{dist} = v_{in} + av_{in}^3, \tag{8.20}$$

where v_{in} is the baseband input signal in either I or Q branch, a is a constant defining the distortion and v_{dist} the distorted signal, the QM DC compensation causes the v_{in} to be replaced with $v_{in} + v_{dc}$. This transforms the formula to the following form:

$$\begin{aligned} v_{dist} &= av_{in}^3 + 3av_{in}^2 v_{dc} + (3av_{dc}^2 + 1)v_{in} + av_{dc}^3 \\ &= a_3 v_{in}^3 + a_2 v_{in}^2 + a_1 v_{in} + a_0, \end{aligned} \tag{8.21}$$

thus second order distortion components are generated.

Another problem is that the amplitude-dependent nonlinearities affect the baseband signal branches separately. Therefore the total nonlinearity seen by the complex baseband signal becomes dependent on the signal phase. This means that the predistorter is not able to compensate these errors. The following section investigates the effect of this distortion.

8.6.1 Analysis of the quadrature modulator nonlinearity

This section will investigate the effect of the quadrature modulator nonlinearity to the output signal of the modulator assuming simple third-order polynomial distortion on the modulator branches. The results also can be used to illustrate the effect of unequal nonlinearities on the modulator branches on the output.

A second-order distortion component was also included in the calculations since the common mode levels of the D/A converters easily cause asymmetric distortion in the following amplifiers and the modulator. The second order coefficients a_{I2} and a_{Q2} consist of the DC offsets v_{dcI} and

v_{dcQ} and the original second order coefficients of the amplifiers, a_{I2orig} and a_{Q2orig}, so that

$$a_{I2} = 3a_{I3}v_{dcI} + a_{I2orig} \tag{8.22}$$

and

$$a_{Q2} = 3a_{Q3}v_{dcQ} + a_{Q2orig}. \tag{8.23}$$

As the distortion separately alters both the baseband quadrature signals, the total effect of the distortion can be written in the following form:

$$v_{distIQ} = \sum_{n=0}^{3} a_{In}\Re(v_{pd})^n + i\sum_{n=0}^{3} a_{Qn}\Im(v_{pd})^n, \tag{8.24}$$

where $\Re(.)$takes the real part and $\Im(.)$ takes the imaginary part of the argument, $a_{In}$and a_{Qn} are the nth order distortion coefficients of the I channel and the Q channel, respectively, and v_{pd} is the complex output signal of the predistorter. Equation 8.24 can also be written in conjugate form [116, 164]

$$v_{distIQ} = \sum_{n=0}^{3}\sum_{m=0}^{3-n} g_{nm}v_{in}^n\overline{v_{in}}^m, \tag{8.25}$$

where $\overline{v_{in}}$ is the complex conjugate of the input signal and the g_{nm} are complex coefficients. This form simplifies the calculations as normal complex arithmetic can be used.

By equating (8.24) with (8.25), the coefficients g_{nm}can be solved. The calculated coefficients are listed in Table 8.3. The only coefficients that can be compensated by the complex gain predistorter are g_{10} and g_{21}; the rest would interfere with the predistorter update. By using exactly identical amplifiers in both branches, coefficients g_{01}, g_{20}, g_{02}, g_{12} and g_{30} can be canceled. However, there is still distortion remaining dependent on the squared absolute value of the signal and the cube of the complex conjugate of the signal. What is more, the amplitude of these components is doubled when the amplifiers are identical.

The next section inspects the results when these distorted signals are applied to a nonlinear power amplifier after up-mixing.

8.6.2 PA output with static predistortion

It can be expected that if the predistorter is designed to compensate for the premeasured nonlinearity of the power amplifier, it cannot compensate for the new distortion components generated to the baseband quadrature signals. Thus, the nonlinearities are amplified by the power amplifier and new ones are further generated by the PA nonlinearity. The following calculations investigate the distortion components at the PA output.

To simplify the calculations, the PA nonlinearity is assumed to be of third order and memory-

Table 8.3: Coefficients in (8.25)

Coefficient	Value
g_{00}	$a_{I0} + ja_{Q0}$
g_{10}	$(1/2 * a_{I1} + 1/2 * a_{Q1}),$
g_{01}	$(1/2 * a_{I1} - 1/2 * a_{Q1}),$
g_{20}	$(1/4 * a_{I2} - 1/4 * I * a_{Q2})$
g_{11}	$(1/2 * a_{I2} + 1/2 * I * a_{Q2})$
g_{02}	$(1/4 * a_{I2} - 1/4 * I * a_{Q2})$
g_{21}	$(3/8 * a_{I3} + 3/8 * a_{Q3})$
g_{12}	$(3/8 * a_{I3} - 3/8 * a_{Q3})$
g_{03}	$(1/8 * a_{I3} + 1/8 * a_{Q3})$
g_{30}	$(1/8 * a_{I3} - 1/8 * a_{Q3})$

less. Thus the PA output signal is

$$v_{out} = v_{pd} + a_{PA} v_{pd} \left| v_{pd} \right|^2, \tag{8.26}$$

where v_{pd} is the PA input signal and a_{PA} is the PA distortion coefficient. According to (8.25) the nonlinear distortion of the baseband signals also generates amplitude-dependent distortion of the complex conjugate of the input signal and phase dependent distortion components.

If we assume that the predistorter generates a fifth order polynomial to compensate for the PA nonlinearity, the predistorter output function becomes

$$\begin{aligned} v_{pd} = \quad & v_{in} + a_{PD3} v_{in} \left| v_{in} \right|^2 \\ & + a_{PD5} v_{in} \left| v_{in} \right|^4 \end{aligned} . \tag{8.27}$$

The coefficients a_{PD3} and a_{PD5} are calculated by inserting (8.27) into (8.26) and setting the third and fifth order coefficients to zero. This gives us

$$a_{PD3} = -a_{PA} \tag{8.28}$$

and

$$a_{PD5} = a_{PA}(2a_{PA} + \overline{a_{PA}}). \tag{8.29}$$

For the following calculations, we write the third-order coefficient of the imaginary part in the baseband amplifiers and the QMs as the coefficient of the real part, a_{IQ3}, multiplied with a_{dQ3}. The second order coefficients are also the same except for the multiplier, a_{dQ2}, which represents the combined effect of the quadrature DC offset on the second order distortion (Equation 8.22) and the actual difference between the distortion coefficients. The first-order amplitude error and the DC offsets are assumed to have been compensated by the predistorter and the linear quadrature modulator error compensation. Thus, we can write $a_{I3} = a_{IQ3}$, $a_{Q3} = a_{dQ3} a_{IQ3}$, $a_{I2} = a_{IQ2}$, $a_{Q2} = a_{dQ2} a_{IQ2}$, $a_{Q0} = a_{I0} = 1$ and $a_{Q0} = a_{I0} = 0$. Now, when we insert (8.27) into (8.25)

Table 8.4: Distortion components in (8.25)

component	value	component	value
C_{10}	1	C_{31}	$\frac{1}{2}a_{PA}(1 - ia_{dQ2})a_{IQ2}$
C_{01}	0	C_{22}	$(\frac{5}{4}a_{PA} - \frac{1}{2}\Re(a_{PA}))(1 + ia_{dQ2})a_{IQ2}$
C_{20}	$\frac{1}{4}(1 - ia_{dQ2})a_{IQ2}$	C_{13}	$(a_{PA} - \frac{1}{2}\Re(a_{PA}))(1 - ia_{dQ2})a_{IQ2}$
C_{11}	$\frac{1}{2}(1 + ia_{dQ2})a_{IQ2}$	C_{04}	0
C_{02}	$\frac{1}{4}(1 - ia_{dQ2})a_{IQ2}$	C_{50}	$\frac{1}{8}a_{PA}((1 + a_{dQ2}^2)a_{IQ2}^2 + (1 + a_{dQ3})a_{IQ3})$
C_{30}	$\frac{1}{8}(1 - a_{dQ3})a_{IQ3}$	C_{41}	$((\frac{9}{16}(1 - ia_{dQ2})^2 + ia_{dQ2})a_{IQ2}^2 + \frac{1}{4}(1 - a_{dQ3})a_{IQ3})a_{PA}$
C_{21}	$\frac{3}{8}(1 + a_{dQ3})a_{IQ3}$	C_{32}	$(a_{dQ2}^2 + 1)a_{IQ2}^2 a_{PA} + (a_{dQ3} + 1)(\frac{3}{4}a_{PA} - \frac{3}{8}\Re(a_{PA}))a_{IQ3}$
C_{12}	$\frac{3}{8}(1 - a_{dQ3})a_{IQ3}$	C_{23}	$(\frac{7}{8}(1 - ia_{dQ2})^2 - \frac{3}{4}ia_{dQ2})a_{IQ2}^2 a_{PA} + (a_{dQ3} - 1)(\frac{5}{4}a_{PA} - \frac{3}{4}\Re(a_{PA}))a_{IQ3}$
C_{03}	$\frac{1}{8}(1 + a_{dQ3})a_{IQ3}$	C_{14}	$(a_{dQ2}^2 + 1)a_{IQ2}^2 a_{PA} + (a_{dQ3} + 1)(\frac{5}{8}a_{PA} - \frac{3}{8}\Re(a_{PA}))a_{IQ3}$
C_{40}	$\frac{1}{4}a_{PA}(1 + ia_{dQ2})a_{IQ2}$	C_{05}	$\frac{1}{16}(1 - ia_{dQ2})^2 a_{IQ2}^2 a_{PA}$

and the result into (8.26) for the PA output, we get the following conjugate form:

$$v_{pao} = \sum_{n=0}^{N} \sum_{m=0}^{N-n} C_{nm} v_{in}^n \overline{v_{in}}^m, \tag{8.30}$$

where the coefficients C_{nm}, were calculated by a series of algebraic manipulations and are listed in Table 8.4. We have limited the N in (8.30) to 5, since the powers exceeding this would not be compensated by the predistorter in any case, and thus the main effects of the modulator distortion lay on the lower powers, which should be totally removed by the ideal predistorter.

When Table 8.4 is studied, it can be seen that the second order distortion components C_{20}, C_{11}, C_{02} and the third-order coefficients, C_{30}, C_{21}, C_{12} and C_{03} are dependent only on the baseband distortion, and the distortion generated on the baseband transfers quite directly to the PA output.

It can also be seen that the second-order components are always present if the second-order coefficients are real. This is the same situation as with the coefficients for the quadrature modulator output signal, and thus no significant advantage can be gained by adjusting the coefficients to be as similar as possible.

Again, the third-order distortion can be minimized by using amplifiers that are as similar as possible, in which case $a_{dQ2} = 1$. In this case, the coefficients C_{30} and C_{12} become zero.

The fourth-order distortion depends on the PA nonlinearity in addition to the second-order baseband distortion. As a_{PA} and a_{IQ2} are usually much smaller than one, the effect of the fourth-order distortion remains quite small. Also the effect of the fifth-order distortion is quite small for the same reason. The fifth-order distortion depends mainly on the third-order distortion coefficients a_{PA} and a_{IQ3} as the second-order coefficients are squared, thus making them small compared to the third-order distortion.

The power spectrum of (8.30) can be fairly accurately estimated by summing the power spectra of the different terms [164]. Thus the effect of each term on the spectrum can be approximated to be $|C_{nm}|^2 \, |\mathcal{F}(R(v_{in}^n \overline{v_{in}}^m))|$where $\mathcal{F}(R(v_{in}^n \overline{v_{in}}^m))$ is the Fourier transform of the autocorrelation of the input signal powers [164].

Figures. 8.16 and 8.17 show the estimated and simulated spectra of the QAM signal with the amplitude normalized to one with two different distortion coefficient combinations. Figure 8.16 represents a more realistic situation with fairly small baseband distortion compared to the PA nonlinearity. It can be seen that, up to third order distortion, the estimated and simulated spectra match well. Above that, the higher order distortion, which was not taken into account in the calculations, starts to dominate. Figure 8.16 represents a case where there is a large distortion at the baseband. The estimated and simulated results correspond to each other well.

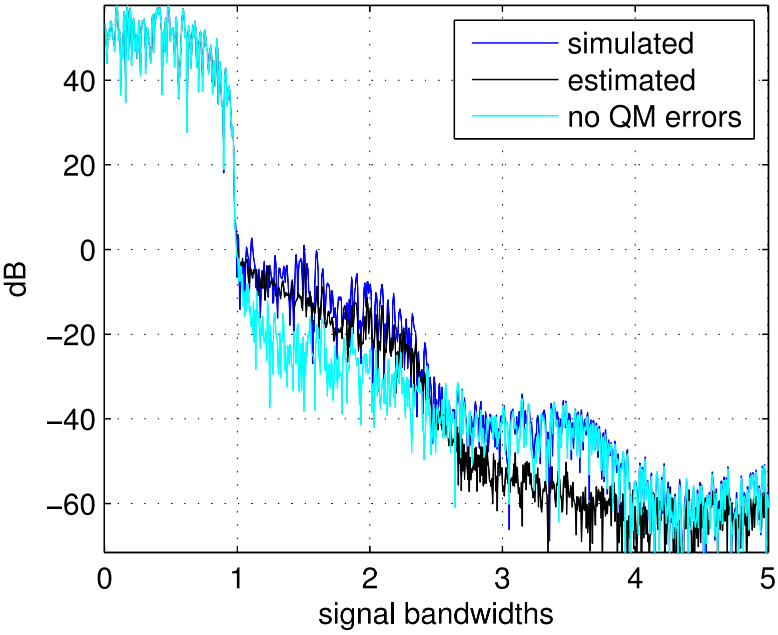


Figure 8.16: Simulated and estimated PA output when $a_{IQ2} = 0.0055$, $a_{dQ2} = 0.45$, $a_{dQ3} = 1$, $a_{PA} = -0.05$

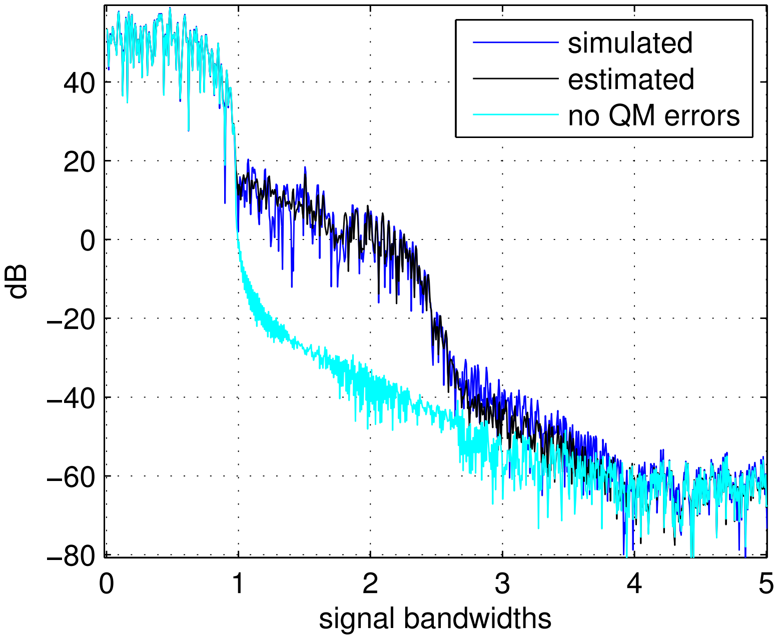


Figure 8.17: Simulated and estimated PA output when $a_{IQ2} = 0.055$, $a_{dQ2} = 0.45$, $a_{dQ3} = 1$, $a_{PA} = -0.02 - 0.004i$ and $a_{IQ3} = 0.05$.

8.6.3 The effect of the quadrature modulator nonlinearity on predistortion update

As discussed in the previous section, a complex gain predistorter can correct nonlinearities that are dependent on the amplitude of the signal. This means that, if the predistortion function is adaptive, it can also compensate for the amplitude-dependent spectral components, namely the terms C_{21} and C_{32} in Table 8.4. This means that, if the third-order distortion in both I and Q branches is similar, the third order distortion level in the PA output is reduced to one quarter of the original baseband distortion.

However, the adaptivity also has its shortcomings. When the predistorter coefficients are adaptively updated, the residual error at the PA output,

$$e = v_{pao} - v_{in}, \tag{8.31}$$

affects the update adversely as it cannot be reduced by predistortion and as it is also dependent on the signal phase. This means that the predistortion coefficients jitter around the optimal values that generate noise to the predistorter output. What would be expected is that the adaptation is able to improve the ACP from the nonadaptive case up to when a_{IQ2} becomes larger than a_{IQ3}.

The previous section discussed for simplicity only the case where the PA nonlinearity is a third order polynomial. However, often this is not the case. In this section, the effect of the adaptation will be discussed in the case where the PA nonlinearity model is more general, namely, the Ghorbani model (Equation 2.18). The predistortion function is $F_k(n) = LUT_n(k)$, where k is the LUT index $k = \lceil 2^{N_{lut}} v_{in}(n) \rceil$. The PA output signal becomes now

$$\begin{aligned} v_{pao}(n) =& F_k(n)v_{in}(n)(1 + A_{distIQ}(F_k(n)v_{in}(n))) \\ & A_{PA}(|F_k(n)v_{in}(n)(1 + A_{distIQ}(F_k(n)v_{in}(n)))|^2) \end{aligned} \tag{8.32}$$

where $A_{distIQ}(F_k(n)v_{in}(n)) = \frac{v_{distIQ}}{v_{in}} - 1$, v_{distIQ} is defined in (8.25).

In the following analysis, linear update (Equation 3.11) is used as the update function. If it is assumed that the error caused by the distortion is small, Equation 8.32 can be simplified by linearizing $A_{PA}(x)$ around the optimal value. Thus,

$$\begin{aligned} v_{pao}(n) =& F_k(n)v_{in}(n)(1 + A_{distIQ}(F_k(n)v_{in}(n))) \\ & (A_{PA} + A\text{'}_{PA} \left|F_k(n)v_{in}(n,k)\right|^2 (2\Re(A_{distIQ}) + |A_{distIQ}|^2)), \end{aligned} \tag{8.33}$$

where $A_{PA} = A_{PA}(|F_k(n)v_{in}(n,k)|^2)$ and $A\text{'}_{PA} = \frac{dA_{PA}(x)}{dx}$. Now, when (8.33) is inserted into (3.12), and if we assume that the error in the current LUT value is $\Delta F_k(n)$, we can approximate the new LUT error as

$$F_k(n+1) = (F_k(n) + \Delta F_k(n))((1-\mu)+$$
$$(\frac{a}{(F_k(n) + \Delta F_k(n))(A_{PA} + A'_{PA} \left|F_k(n) v_{in}(n,k)\right|^2 (2\Re(A_{distIQ}) + \left|A_{distIQ}\right|^2)(1 + A_{distIQ}))})). \tag{8.34}$$

By using series expansion, $\frac{1}{1+x} \approx 1 - x$ if x is small, (8.34) can be further simplified to form

$$F_k(n+1) = (F_k(n) + \Delta F_k(n))((1-\mu)+$$
$$(\frac{\mu}{F_k(n)A_{PA}}(1 + \frac{\Delta F_k(n)}{F_k} - + \frac{A'_{PA}}{A_{PA}} \left|F_k(n) v_{in}(n,k)\right|^2 (2\Re(A_{distIQ}) + \left|A_{distIQ}\right|^2)(1 + A_{distIQ})))). \tag{8.35}$$

Now writing (8.35) in the form

$$F_k(n+1) = F_k(n)(1 - \frac{1}{F_k(n)A_{PA}}) + \Delta F_k(n+1)$$

and assuming perfect predistortion, i.e. $F_k = \frac{1}{A_{PA}}$, we can find the misadjustment of the LUT update. If we also approximate $(\Re(A_{distIQ}) + \left|A_{distIQ}\right|^2)$ by $\left|1 + A_{distIQ}\right|^2$ and note that $(\frac{1}{A_{PA}})' = \frac{A'_{PA}}{A_{PA}^2}$, we get as the misadjustment

$$\Delta F_k(n+1) \approx (1+\mu)\Delta F_k(n) + \mu(1 + A_{distIQ})F_k^{'}(n) \cdot$$
$$(\left|1 + A_{distIQ}\right|^2 - 1)\left|F_k(n) + \Delta F_k(n)\right|^2 \left|v_{in}\right|^2 . \tag{8.36}$$

When the result is analyzed, we see that the misadjustment is dependent on the squared absolute value of the input signal, thus the jitter is greatest at large amplitudes. The predistortion function often reaches its maxima at the low and high amplitudes, which further increases the concentration of the jitter on the highest amplitudes. Finally, the misadjustment depends on the shape of the predistortion function.

8.6.3.1 Simulation results

The effect of QM nonlinearity was simulated using Matlab, with a 16QAM signal and a sampling frequency of 56 times the symbol frequency. The amplitude of the signal was normalized to one. The PA model was PA2. Third-order baseband nonlinearity (Equation 8.24) was used. The baseband predistortion system shown in Figure 8.10 was used in simulations.

Figure 8.18 shows the standard deviation of the LUT entries after the LUT update has converged, with different second- and third-order baseband distortion coefficients. What can be seen is that the second-order coefficient affects the standard deviation much more than third-order coefficient, as could be expected from the calculations in Section 8.6.2. It can also be noticed that the second-order coefficient does not affect the standard deviation at the higher amplitudes as much as the third-order coefficient due to the shape of the second-order distortion function.

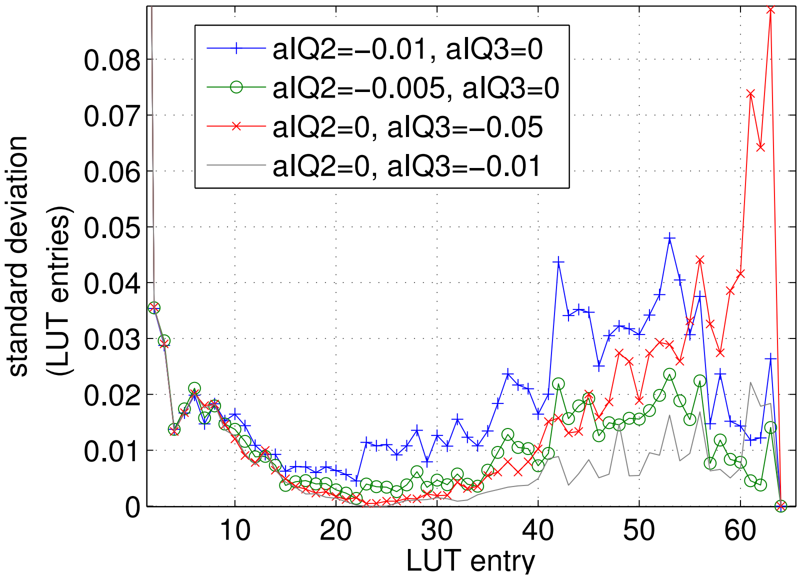
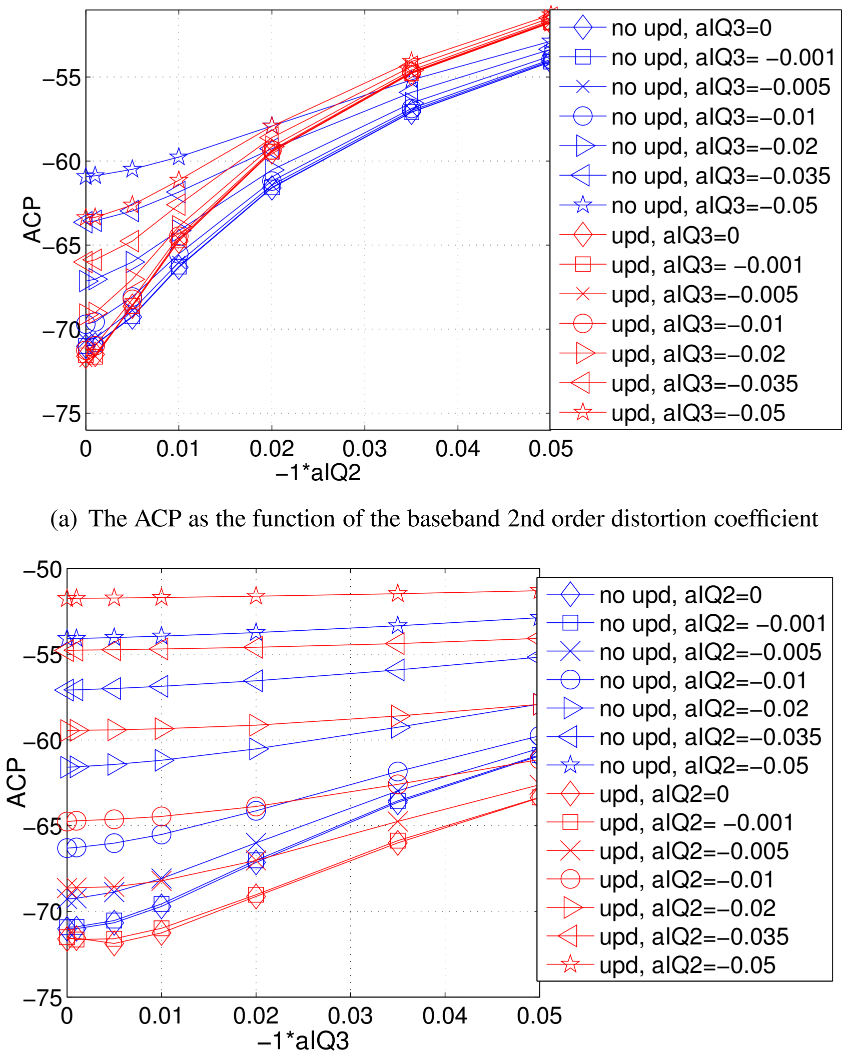


Figure 8.18: Standard deviation of LUT entries over time with different second and third order nonlinearity coefficients

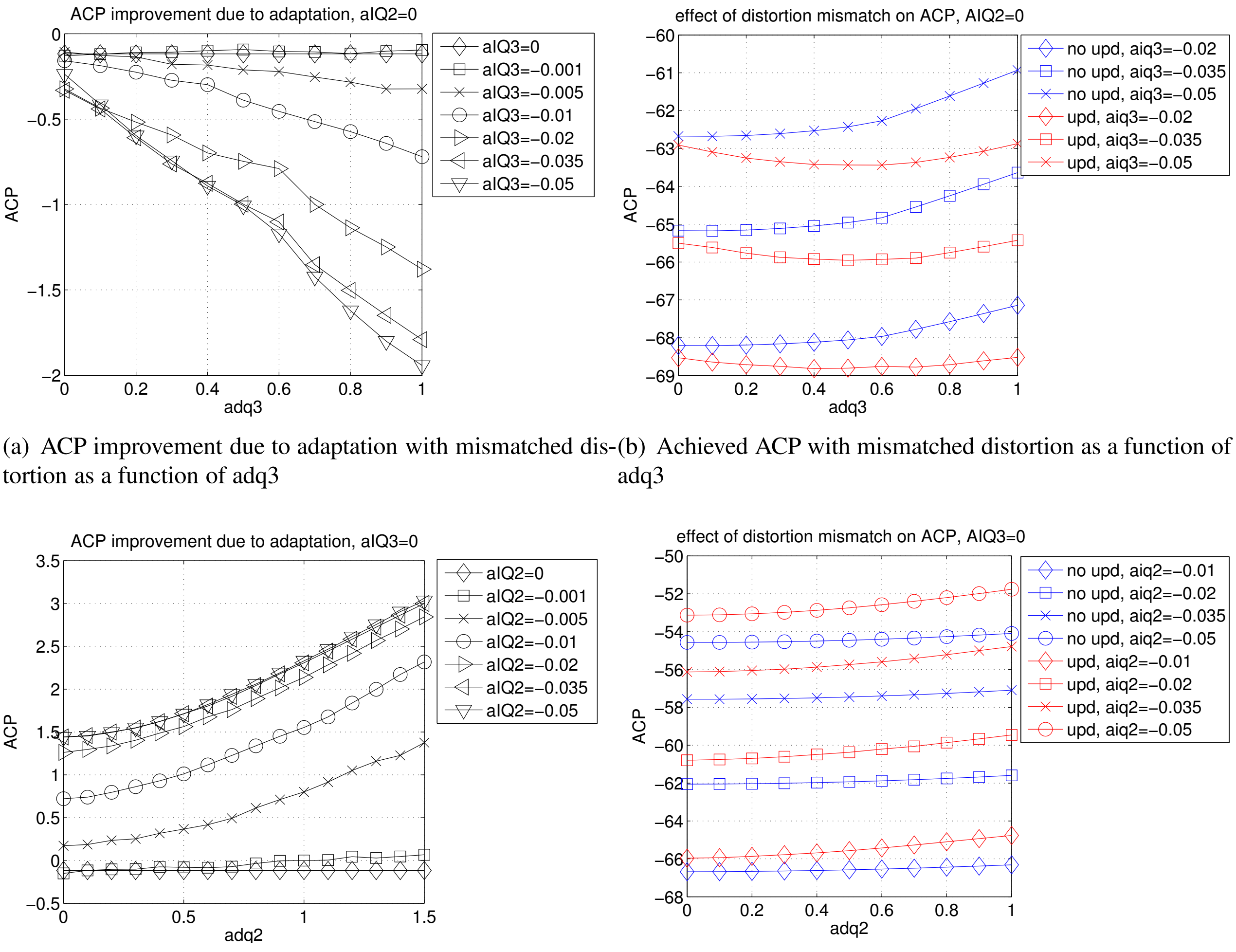


(a) The ACP as the function of the baseband 2nd order distortion coefficient

(b) The ACP as the function of the baseband 3rd order distortion coefficient

Figure 8.19: The PA ACP as a function of the baseband distortion

Figure 8.19 shows the behavior of the PA output ACP as a function of the quadrature modulator nonlinearity. The figure shows also the ACP with static second-order distortion and variable third-order distortion and vice versa. The results are shown also with and without predistortion update. The same initial condition was used both for the static and dynamic predistortion functions. The figures clearly show how the second-order distortion starts to dominate the third-order distortion and eventually changing the second order distortion does not have any effect on the PA output. Another thing that can be seen is that, when the second order distortion is small, the ACP can be improved, using the adaptive predistortion, and when the second order distortion dominates, it leads the update astray and actually makes the ACP worse, as expected from the calculations in Section 8.6.2. The transition happens at the point when the second distortion coefficient is larger than half the third order distortion coefficient.



(a) ACP improvement due to adaptation with mismatched distortion as a function of adq3

(b) Achieved ACP with mismatched distortion as a function of adq3

(c) ACP improvement due to adaptation with mismatched distortion as a function of adq2

(d) Achieved ACP with mismatched distortion as a function of adq2

Figure 8.20: The PA output ACP as a function of the second and third-order baseband distortion coefficients.

Figure 8.20 shows the simulated effect of mismatch of the distortion components to the ACP of the PA output. Figure 8.20a shows the improvement in ACP gained by adaptive predistortion with different third order nonlinearity mismatches, while Figure 8.20b shows the actual ACPs with adaptive and nonadaptive predistorters.

The figures agree with the calculations in Section 8.6.2 and show that, when adaptive predistortion is in use, the component C_{21}can be compensated by the adaptive predistortion, and the remaining terms achieve the minimum when the distortion is the same on both branches, instead of the case when the other branch is completely linear.

When no adaptation is used, the minimum is achieved by setting the other branch to be as linear as possible, as the uncompensated C_{21} does more harm than the benefit that is gained by eliminating C_{12} and C_{30}. Figures 8.20c and d show the effect of second order distortion mismatch. As expected, the minimum is achieved when the distortion of the other branch is zero and, as the second-order distortion cannot be compensated by the adaptive linearization, inclusion of adaptation only makes the situation worse due to the instability of the update.

8.6.4 Compensation of quadrature modulator nonlinearity

Since the quadrature modulator nonlinearity has significant effect on the operation of a complex gain predistorter, the nonlinearity should be compensated. Faulkner et al. [176] present a simple circuit that can be used. The circuit uses a simple third-order polynomial with two adjustable coefficients (one for each branch) to compensate the distortion. This circuit is based on the assumption that the second-order distortion is completely due to DC offset in the third-order distortion and thus no second order distortion compensation is required. The circuit can be made adaptive using a method similar to the first method depicted in Section 8.5.1 [176] .

During the adaptation, the I and Q channels are toggled separately between -A and A, while the other channel is kept at zero. [176] The A is an arbitrary amplitude value. The third order distortion coefficients kI and kQ are adjusted until the envelopes at amplitude values -A and A are the same.

During measurements it was noted that also second-order baseband distortion independent of the DC disturbed the system. Thus, a second-order compensation was also added to the circuit. The implemented baseband distortion correction with the correction of linear quadrature modulator errors is shown in Figure 8.21. kI2 and kI3 are the second- and third-order I channel compensation coefficients respectively and kQ2 and kQ3 are the second- and third-order Q channel compensation coefficients respectively. In the quadrature demodulator error correction circuit, the order of the blocks is reversed for the feedback path as was the case with the linear quadrature demodulator error correction circuit (Section 8.5).

8.7 Hardware implementation

The digital predistorter circuit in Figure 8.11, the quadrature modulator compensation circuit in Figure 8.21 and the corresponding quadrature demodulator compensation circuit were implemented with an ALTERA Cyclone EP1C20F400C7 FPGA [177]. The word lengths of the inputs and the outputs were chosen as 16 bits due to the baseband transmitter circuitry implementation.

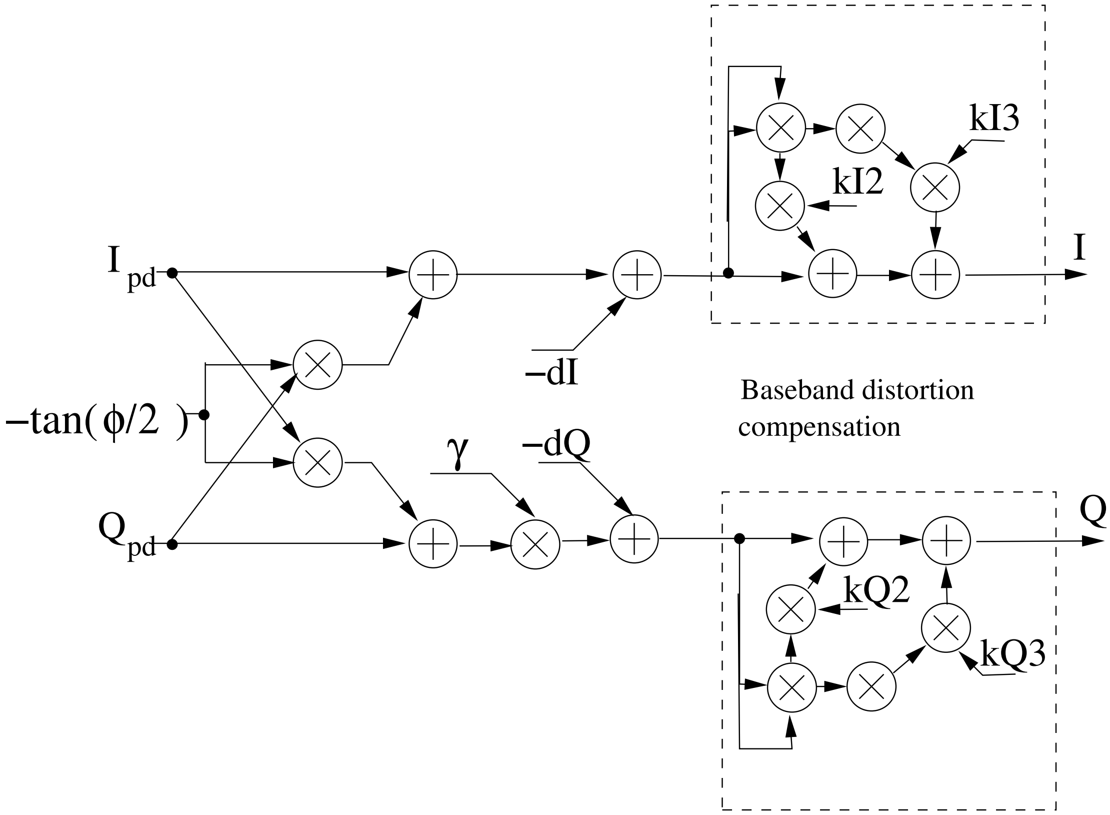


Figure 8.21: Quadrature modulator error compensation circuit.

The predistorter LUT size was chosen as 64 entries and thus the address length was 8 bits. Adjustable delay between the digital input and feedback input, as well as an adjustable value of μ in (9.6), was implemented. The delay adjustment can be made in half clock cycle steps using a fractional delay filter [178]. The square-root circuit used in the address calculation (Figure 8.8) was implemented with a 64-entry LUT. The value of A required in the quadrature compensation algorithms was set to be 16384 or half of the maximum signal amplitude. The FPGA implementation consumed 3.1 kbits of memory and 15600 logic cells.

The sampling frequency was chosen as 1 MHz and the clock was generated with a crystal oscillator. The output D/A conversion was accomplished with two discrete 16-bit serial 1 MHz D/A converters and the feedback inputs were converted to digital domain using two discrete 16 bit 1 MHz A/D converters. The D/A and A/D converters and the clock generator were on separate circuit boards. A second-order Butterworth filter with 200 kHz knee frequency was used in the D/A output and at the A/D converter input.

The envelope detector used in first of the fourth quadrature modulator error correction algorithm in Section 8.5 was implemented using a logarithmic amplifier. The conversion of the envelope to digital domain was implemented with a 16-bit A/D converter. The adaptation algorithms for the quadrature modulator compensation were implemented using a Python program run on a PC connected to the system with a serial cable.

A three-stage class-AB power amplifier chain with 50 dB power gain was used as the device to be predistorted. The measured gain variation of the amplifier as a function of the output power (AM-AM) and the output phase as a function of the output power (AM-PM) are shown in Figure 8.22. The average output power used in the measurements was 30 dBm. The PA was implemented using discrete components. The quadrature modulators and quadrature demodulators (QDM) were also implemented on the same board. The PA was optimized for a 400 MHz carrier frequency.

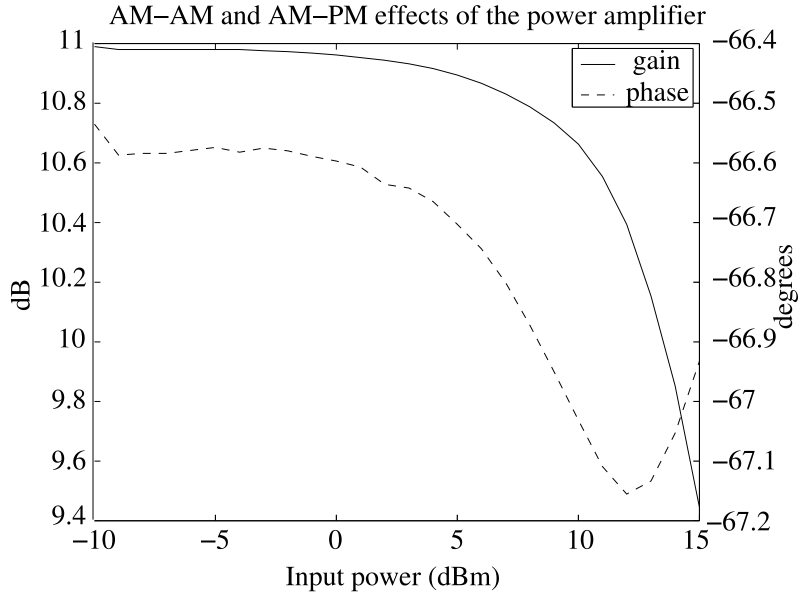


Figure 8.22: Measured AM-AM and AM-PM of the power amplifier

The predistortion system and the measurement setup are shown in Figure 8.23. Different measurement boards are marked by a box with a letter in the upper right corner. Block A is the circuit board containing the FPGA that implements the digital algorithm. The blocks marked with B are boards containing a 16-bit D/A converter and an active second order Butterworth filter. The blocks marked with C are boards containing a 16-bit A/D converter and an active second-order Butterworth filter. Block D is the board for the logarithmic detector, while Block E is the board that contains the quadrature modulator and the demodulator and the power amplifier. Additionally there is a directional coupler and adjustable attenuator. The phase of the carrier fed to the quadrature demodulator is adjustable. Board E contains connectors for a signal generator (G) for the carrier generation and for a spectrum analyzer, oscilloscope or a network analyzer (F). The FPGA board (A) has connections for a clock generator (K), pattern generator (J), logic analyzer (H) and a serial connector for a PC (I).

8.8 Measurements

The circuit was measured using a 16QAM signal with symbol frequency of 18 ksym/s, oversampled to a sampling frequency of 1 MHz and filtered with a 0.3 roll-off factor root raised cosine filter. The signal power was set to be 29 dBm without the predistortion. The signal was generated using Matlab and fed to the circuit using a pattern generator. To find the image and carrier leakage

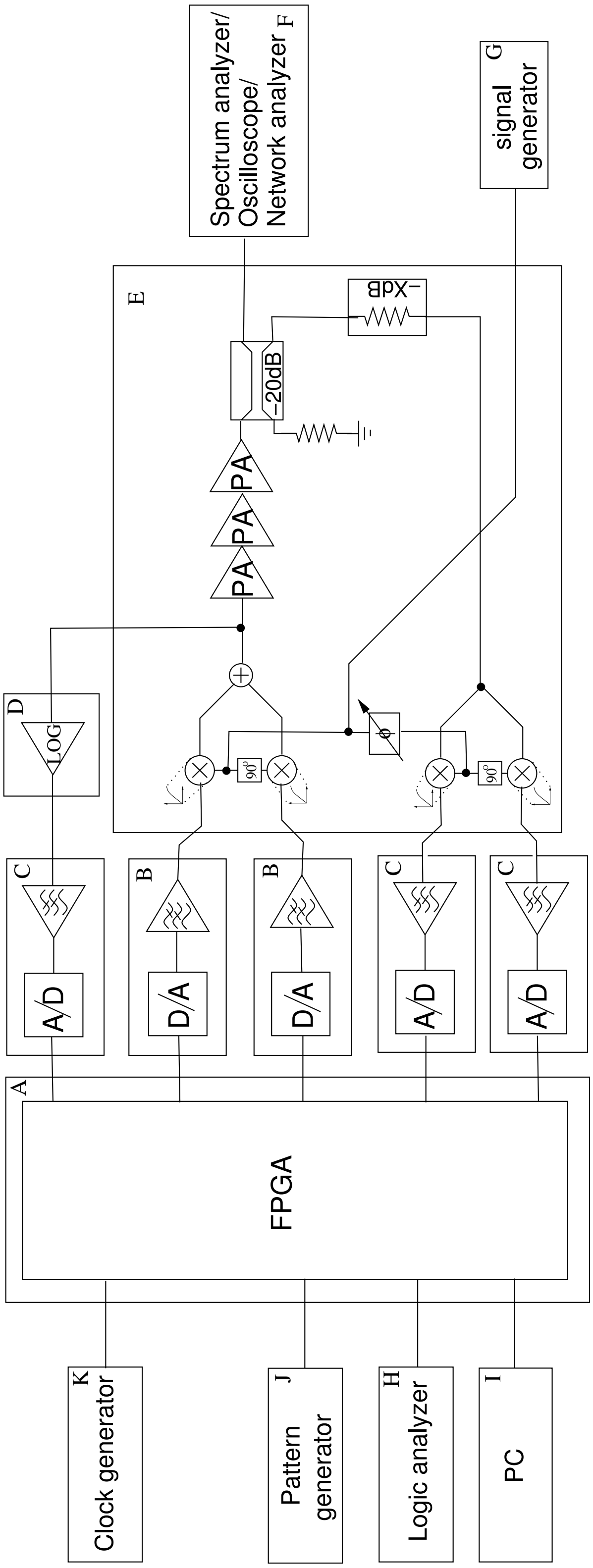


Figure 8.23: The measurement setup for the baseband predistorter

suppression ability of the quadrature error correction algorithm a 30 dBm quadrature sine wave (Equation 8.8) with a 9 kHz offset from the carrier was used.

The measured first ACP of the amplified signal without predistortion was 37 dBc. The measured phase error was 4^o, the gain error was 3.4% of the amplitude and the DC offset was 3.0% of the amplitude. When the predistortion was added without the quadrature compensation the ACP improved to only 43 dB, which could be expected according to simulations.

When the quadrature modulator errors were compensated using the first quadrature adaptation method described in Section 8.5, the magnitude error decreased to 1.8%, the phase error to 1.6^o and DC offset to 1.1%. However, when the adaptive predistortion was turned on, the ACP did not improve. This clearly shows the importance of compensating the quadrature demodulator errors. The quadrature demodulator errors were then compensated using the third method described in Section 8.5.

It was noted that the system was very sensitive to DC errors in the feedback path and that the predistorter forced the signal to zero when the signal was close to the origin of the complex plane if the DC offsets were not completely compensated. It also turned out that the DC correction values achieved using the quadrature sine or the learning signals of the first method were not good enough for the 16QAM signal. When the feedback DC levels were adjusted manually, the problem could be removed and the ACP improved to 52 dBc. The acquired optimal DC correction values were considerably smaller compared to the values that zeroed the fed back teaching signal. The reason for this could not be ascertained although a possible reason could be the second-order distortion in the feedback loop.

The comparison between the spectra with and without feedback DC correction is shown in Figure 8.24. The figure also shows the effect of correcting other quadrature demodulator errors in addition to the DC offset and, as can be seen the improvement is only 5dB. Thus the DC offset is the dominating factor on the feedback path of the measured system. These feedback error correction values were used during the tests of the quadrature modulator error correction methods.

The measured spectrum of the linearized signal with quadrature error compensation adapted using the first method is shown in Figure 8.25 with the nonlinearised spectrum and the linearized spectrum without the quadrature compensation. The predistorter is able to improve the ACP less than 10 dB when the quadrature modulator compensation is not in use; also the noise floor rises significantly. When the quadrature modulator compensation is turned on, the ACP improves to 52dBc. The noise floor is still higher than in the original signal, but not much. The asymmetry of the sidebands, that was also noted during measurements of the RF-predistortion system (section 7.4) is visible and is most probably due to memory effects in the amplifier.

The second quadrature error compensation algorithm presented in Section 8.5 gave similar results ACP-wise as the first one, although the measured quadrature errors after the linearization were smaller. The measured linearized spectrum is plotted in Figure 8.26.

The adaptation of the fourth method was started by first adjusting the quadrature demodulator compensation DC values as they had proved to be the most significant error source. Thereafter,

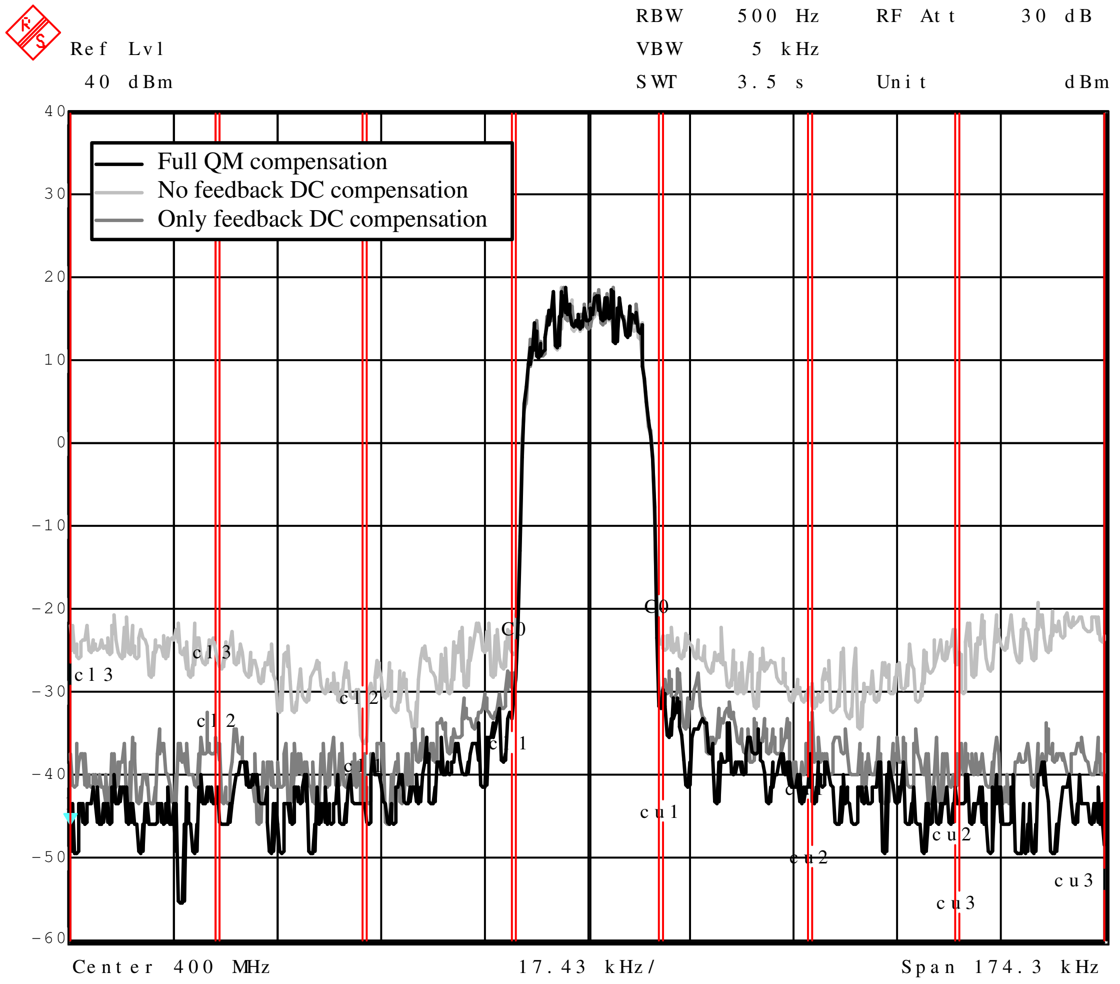


Figure 8.24: Spectrum of measured signal when using the first QM compensation method with QDM DC compensation, only QDM DC compensation and only QM compensation.

the quadrature modulator error compensation coefficients were adjusted and finally the rest of the QDM error compensation coefficients were adjusted to minimize the ACP. The ACP measurements of the predistorter were similar to the first and second method, but the quadrature error measures were worse due to the fact that the optimization was performed on the basis of the ACP. The measured spectrum is shown in Figure 8.27.

The ACP measurement and simulation results with the different quadrature compensation methods are collected in Table 8.5 and the measured quadrature errors in Table 8.6. The methods from the first to the fourth refer to the methods presented in Section 8.5. The optimal QDM DC correction refers to the DC correction values obtained during the measurements of the first method. The first, second and fourth quadrature compensation methods gave similar results ACP-wise. However, when the complexity is considered, the first method is clearly the most advantageous. It should, however, be noted that the optimal feedback DC values were found directly only by the fourth method; others required additional tuning. The third method by itself clearly proved to be unusable.

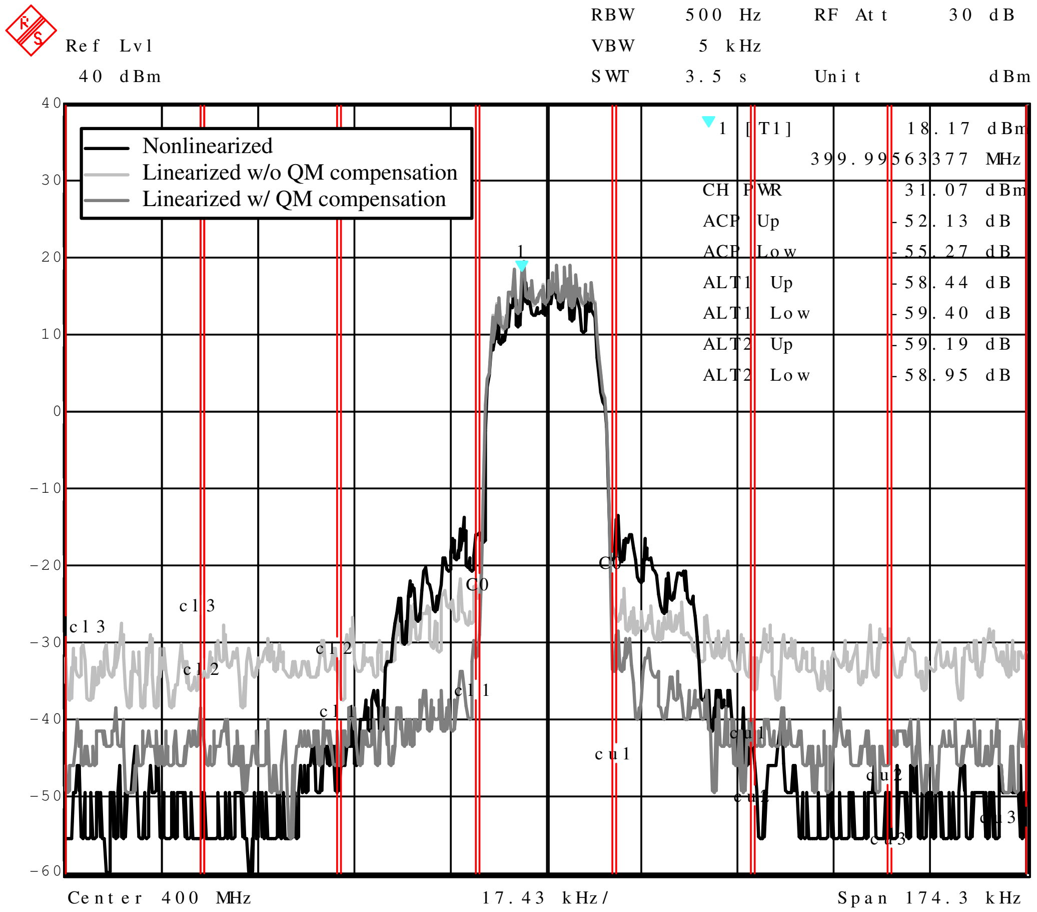


Figure 8.25: Spectrum of measured signal when using QM correction based on envelope measurements (first method)

Table 8.5: Measured and simulated first and second ACP using different quadrature correction methods.

Method	first ACP	second ACP
No predistortion	37 dBc	63 dBc
No predistortion (simulated)	37 dBc	57 dBc
predistortion without quadrature correction	43 dBc	47 dBc
predistortion with optimal QDM DC correction	47 dBc	53 dBc
predistortion with method 1 (opt QDM DC correction)	52 dBc	58 dBc
predistortion with method 1(simulated)	58 dBc	63 dBc
predistortion with method 2	51 dBc	57 dBc
predistortion with method 2(simulated)	55 dBc	58 dBc
predistortion with method 3	42 dBc	44 dBc
predistortion with method 3(simulated)	44 dBc	50 dBc
predistortion with method 4	52 dBc	58 dBc

Table 8.6 shows that the best quadrature error compensation was achieved using the second method. The table also clearly shows the importance of the compensation of DC errors. All the methods suffer from increased noise floor which causes the second ACP to increase. Although the results are not as good as in the simulations, the effectiveness of the quadrature compensation

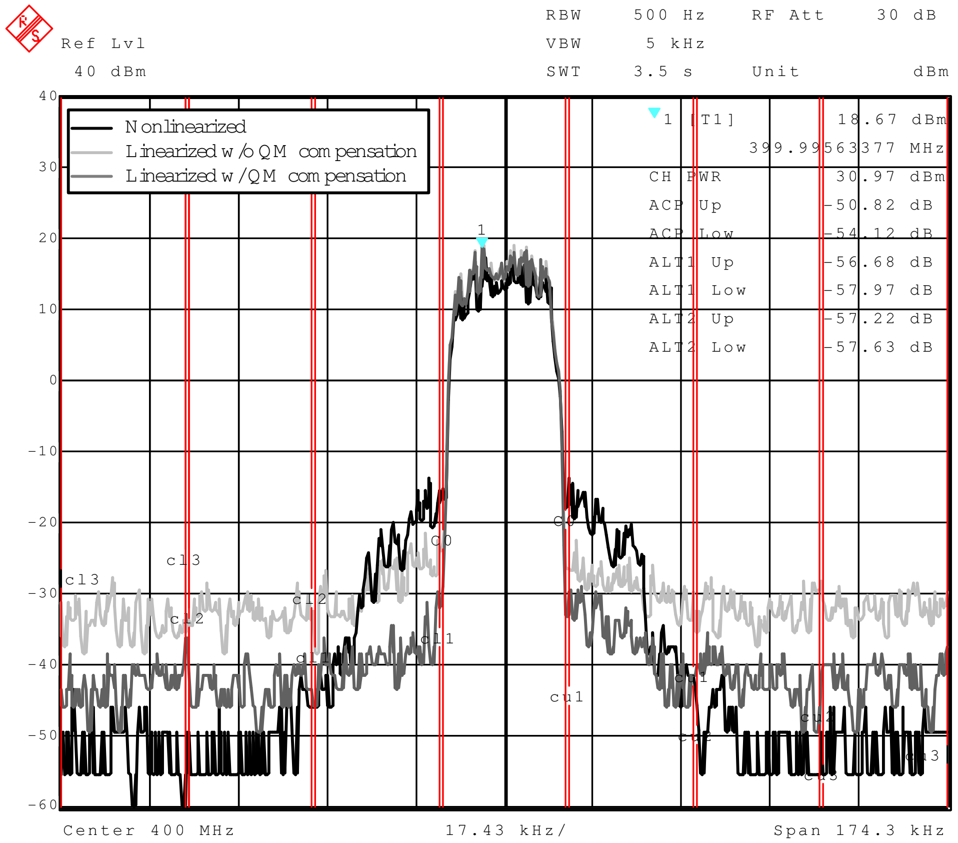


Figure 8.26: Spectrum of measured signal when using QM correction based on sine test signal (second method)

methods when compared to each other was similar in measurements and simulations.

Table 8.6: Quadrature errors with different coefficient finding methods.

Method	MagErr	PhaErr	Offset
No quadrature correction	4.0%	3.4^o	3.0%
Method 1 (predistortion off)	1.8%	1.6^o	1.1%
Method 1 (predistortion on)	1.3%	1.4^o	1.0%
Method 1 (simulated)	0.12%	0.7^o	0.45%
Method 2 (predistortion off)	2.5%	1.5^o	0.6%
Method 2 (predistortion on)	0.9%	1.1^o	0.5%
Method 2 (simulated)	0.05%	0.6^o	0.2%
Method 3 (predistortion on)	3.9%	4^o	3.8%
Method 4	1.5%	1.8^o	1.3%

8.9 Conclusions

This chapter presented the design and simulation and measurement results of a digital baseband complex gain predistorter with a quadrature modulator and demodulator error correction circuits. Different adaptation methods for the quadrature modulator error correction circuits were com-

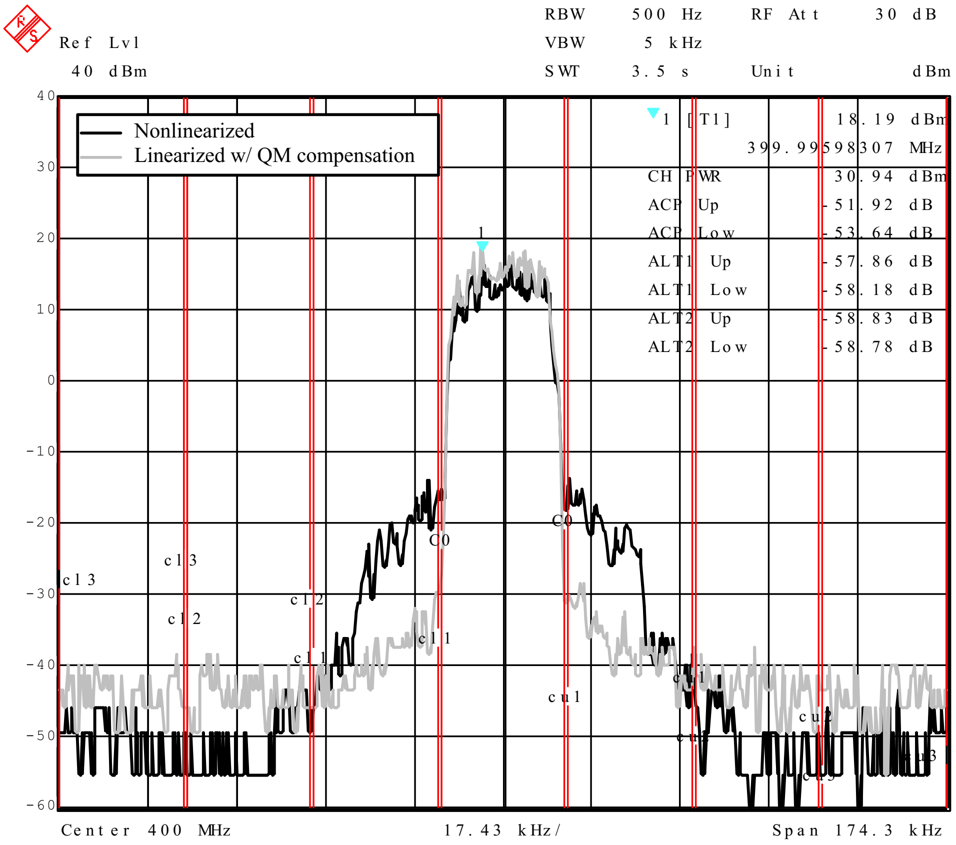


Figure 8.27: Spectrum of measured signal when using QM and QDM correction based on actual transmitted signal (method 4)

pared and the effect of quadrature modulator nonlinearity on baseband predistortion was investigated.

Implemented predistorter uses complex gain topology with two 64-entry LUTs and linear LUT output interpolation and LUT input deinterpolation. The quadrature error correction circuit uses the fed back I and Q signals to update the correction coefficients. The predistorter was implemented using an FPGA and a three-stage PA chain including quadrature modulators and demodulators was used as the device to be predistorted. The signal used in the simulations and measurements was a 18 ksym/s 16QAM signal at 400 MHz carrier frequency. In measurements, the ACP improvement was 15 dB when the quadrature correction was in use. The removal of the quadrature error correction increased the ACP by 9 dB compared to the situation without quadrature correction. The tested quadrature compensation adaptation methods gave similar results ACP-wise. The measurements clearly showed that the quadrature compensation of the feedback signal is very important to correction ability of the predistorter. Especially the feedback DC levels affect the correction considerably; by compensating only the feedback DC level the first ACP improved by 4 dB.

The investigation of the nonlinear quadrature modulator errors showed that, to minimize the effect of third-order nonlinearity, the nonlinearity in both in-phase and quadrature branches should be the same. On the other hand, the effect of second-order nonlinearity is minimized only when either or both second-order components are completely removed. It was also shown that an adaptive predistorter is able to compensate part of the third-order nonlinearity, but the second-order nonlinearity causes convergence problems to the adaptation due to large residual errors. It was shown that the results with an adaptive predistortion are worse than the results with a fixed predistorter.

Chapter 9

LUT size, indexing, interpolation and update

9.1 Introduction

This chapter concentrates on generation of the LUT and predistortion control, that are important in designing a digital predistortion system and are applicable to both RF and baseband predistorters. The chapter investigates the LUT related methods that can be found in literature and their application to the predistortion architectures presented in this thesis.

One of the most fundamental aspects of a digital LUT based predistorter is the size of the LUT, as this affects the size of the hardware and the maximum achievable linearity of the predistortion system. In principle, the larger the LUT, the more linearity that can be achieved [107]. Several methods have been presented to reduce the number of LUT entries without affecting the achieved ACP by using some more optimal indexing method than the normal amplitude indexing [78, 108, 110, 168] or by interpolating the LUT output to increase the virtual number of LUT entries [91, 130]. To increase the accuracy of the predistorters, they often are implemented adaptively. For the update, several methods, whose applicability depends on the number of the parameters and thus of the LUT size, have been presented. This chapter gives a summary of some of those methods and compares the linearization abilities of them.

At the end of the chapter some improvements to the existing frequency-domain LUT update methods, to improve the convergence of the LUT, are suggested on basis of simulation results of the studied methods.

9.2 The effect of the LUT size and entry accuracy

The size of the LUT defines how closely the digital predistorter can follow the optimal predistortion function. The more entries, the more accurate the predistortion function, but, also, the larger the LUT and the slower the convergence of the adaptive predistorter. Actually, the limited number of LUT entries generates a kind of dual quantization for the predistortion function: first the input data is quantized to the LUT-address accuracy for the indexing and then the value of the predistortion function at this quantized point is further quantized to the output accuracy. Figures 9.1a and b visualize the effect of this double quantization.

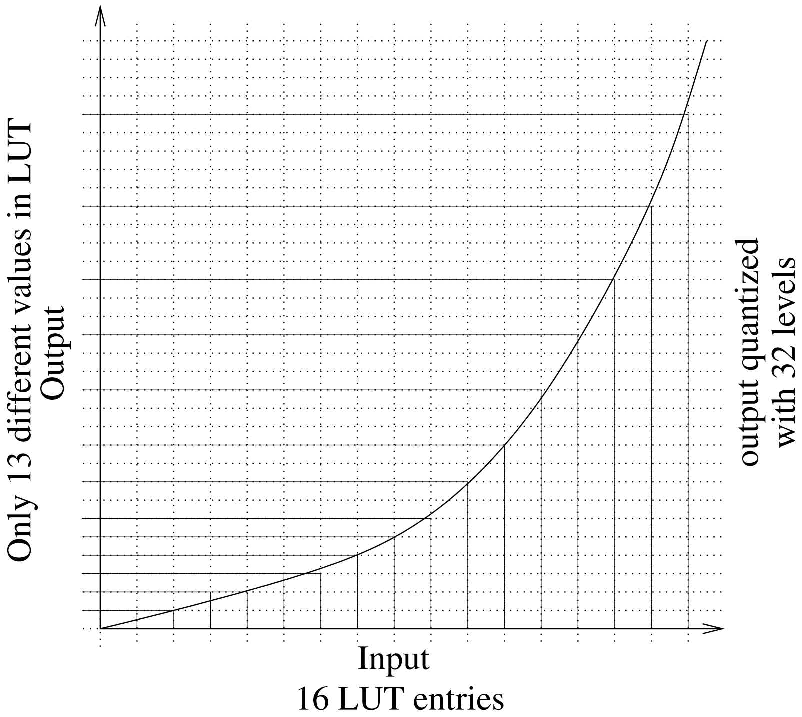
In Figure 9.1a, the LUT address is quantized to 4 bits, the output is quantized to 5 bits and the predistortion function is an exponential function. It can be seen that, when the derivative of the predistortion function is small, the limited output accuracy causes some of the LUT entries to contain the same values, thus reducing the effective number of LUT entries from 16 to 13. On the other hand, when the derivative is large, the output value changes with very large steps, thus sacrificing part of the output accuracy.

Figure 9.1b shows the actual quantized version of the predistortion function in the case where only the output is quantized with 4 bits and the case where the input is quantized with 4 bits and the output is quantized with 8 bits. The increased output accuracy clearly helps when the derivative is small but the steps are still large when the derivative increases. This clearly shows how the effect of the LUT size on the linearization ability of a predistorter depends on the shape of the predistortion function. This also causes the results achieved for one PA not to be directly applicable to another.

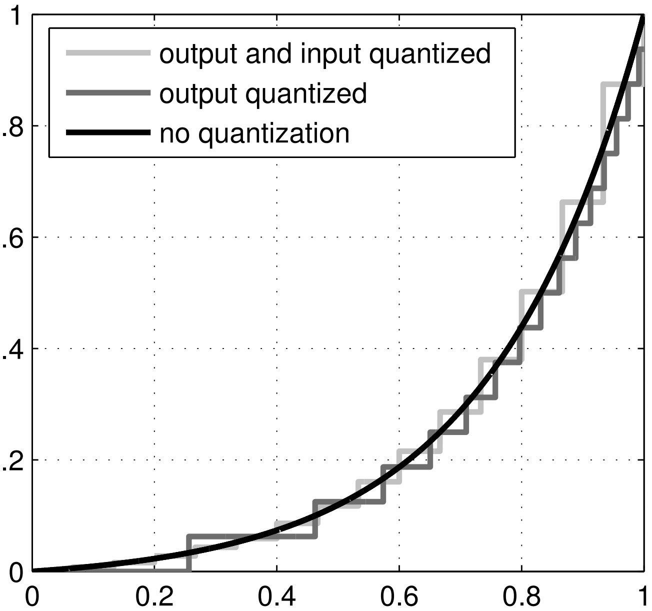
Symbolic analysis for the linearity degradation of a predistorted power amplifier due to quantization is presented in references [107, 108]. The resulting formulas are quite complicated and are not dependent only on the derivative of the predistortion function, but also on the statistical distribution of the signal amplitude. This gives rise to the fact that, even with these formulas, it is difficult to draw detailed conclusions as to the operation of a predistorter without recalculation of the results for each signal and nonlinearity type. Even then, it is still necessary to resort to simulations and measurements to verify the effects of the quantization.

Figures 9.2a and b show the simulated ACPs for PA2 and PA3 (Section 2.7) as a function of the LUT address word length, when using a complex gain predistorter. The curves are plotted for five different LUT entry word lengths. When the results are compared, it can be seen that in the case of PA3 the ACP improves about 10 dB for 2 bits of LUT address length, while, in the case of PA2, it improves about 15 dB for 2 bits of LUT accuracy. The difference is due to the different shapes of the predistortion functions. Figure 9.3 shows the similar results for an RF predistorter.

The LUT size also affects the convergence speeds of the time-domain LUT update methods, since these methods usually update only a single or a small number of LUT entries at a time. This means that, the more entries there are, the longer it takes for the LUT to reach the final value. Thus, for faster convergence, a smaller LUT would be preferred. Figure 9.4 compares the speed of the convergence of an LUT in an RF predistorter using linear update with different LUT sizes.

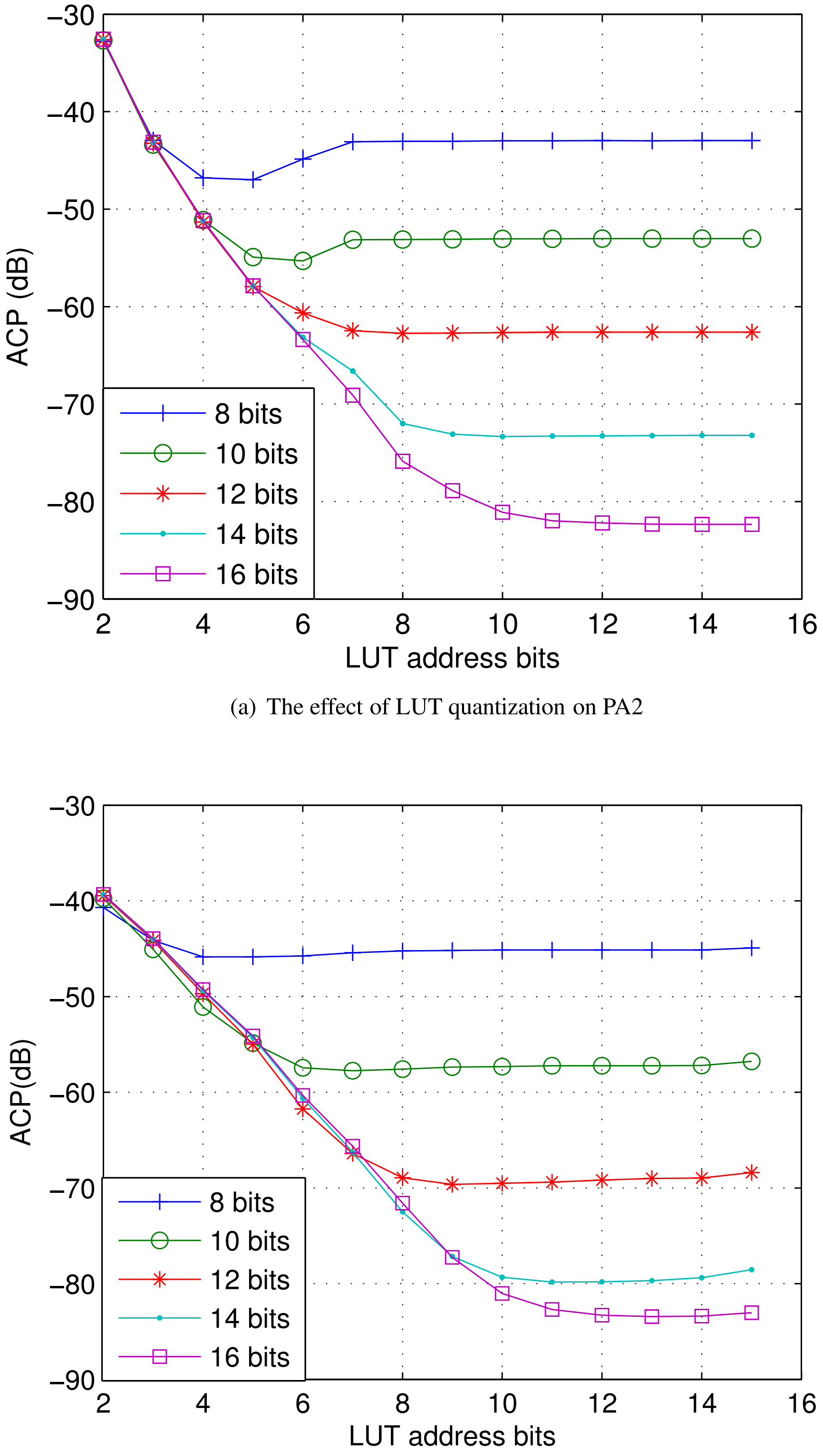


(a) Direct LUT indexing



(b) quantized function

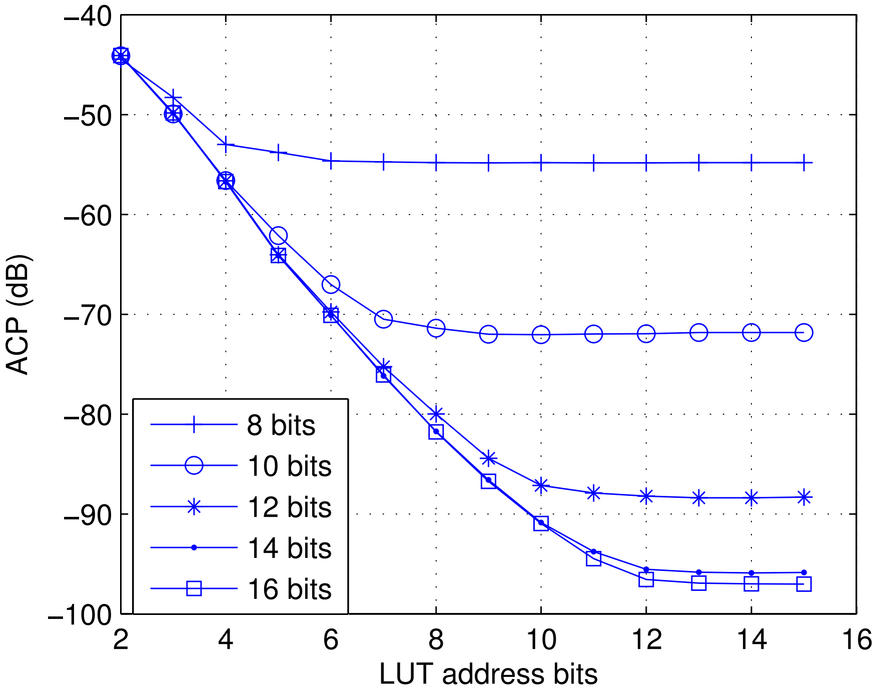
Figure 9.1: The effect of limited number of LUT entries and Quantization



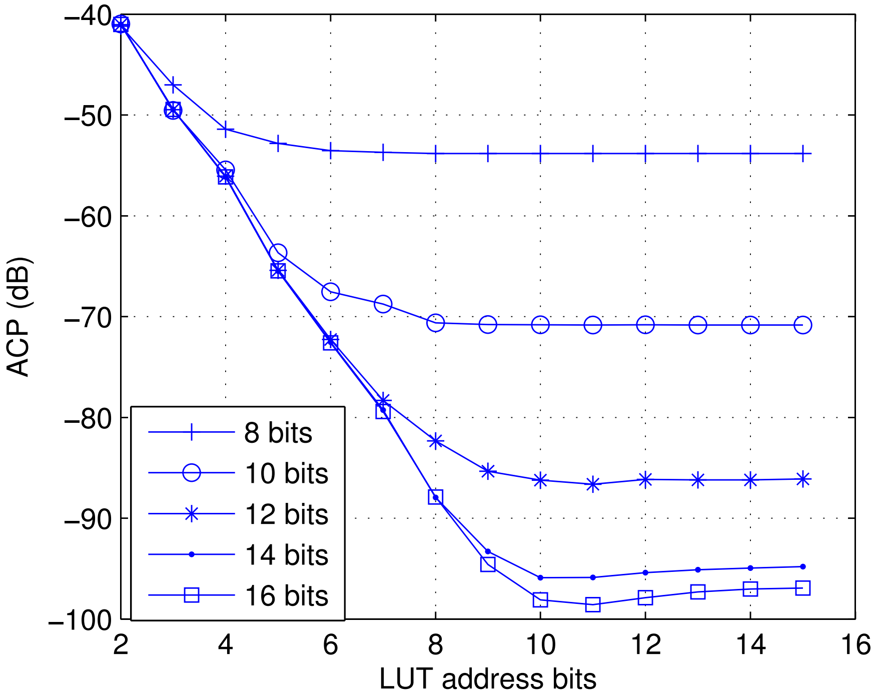
(a) The effect of LUT quantization on PA2

(b) The effect of LUT quantization on PA3

Figure 9.2: The effect of LUT quantization on the effectiveness of a complex gain predistorter



(a) The effect of LUT quantization on PA2



(b) The effect of LUT quantization on PA3

Figure 9.3: The effect of LUT quantization on the effectiveness of an RF-predistorter

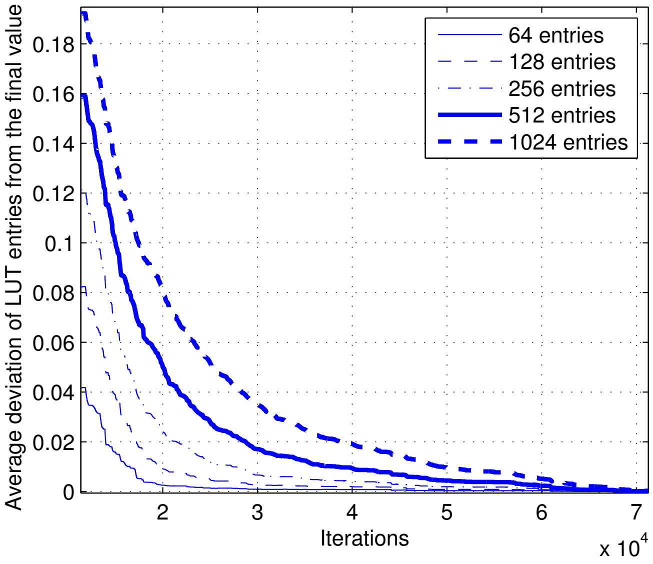


Figure 9.4: The effect of LUT size on convergence speed

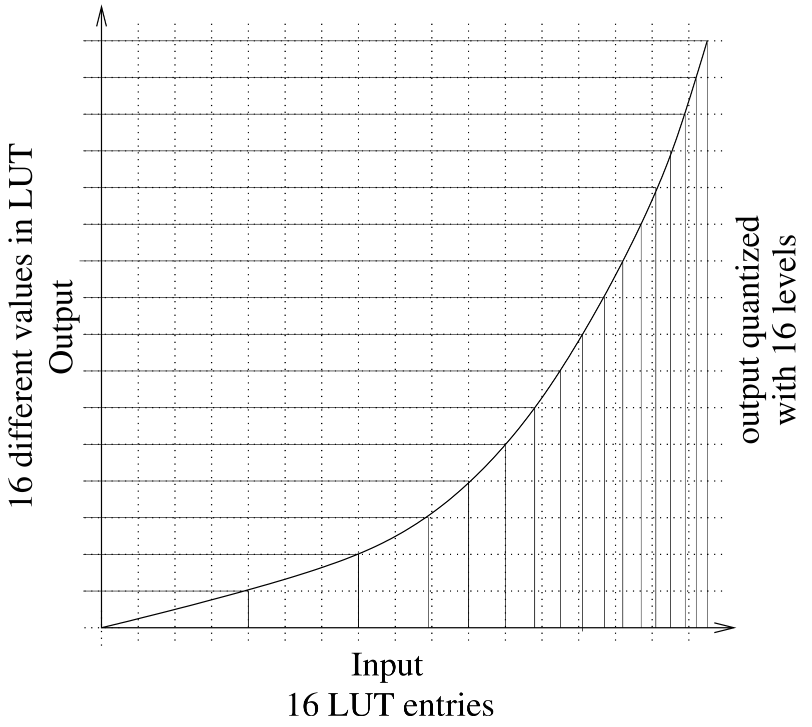
9.3 LUT indexing

To improve the ACP without affecting the size of the LUT it would be beneficial eliminate the overlapping LUT entries. Also, if we do not know the distribution of the input signal, which is the case in, for example, multi mode transmitters, we would like to have the LUT entries to be distributed as evenly as possible to avoid large steps at the output signal.

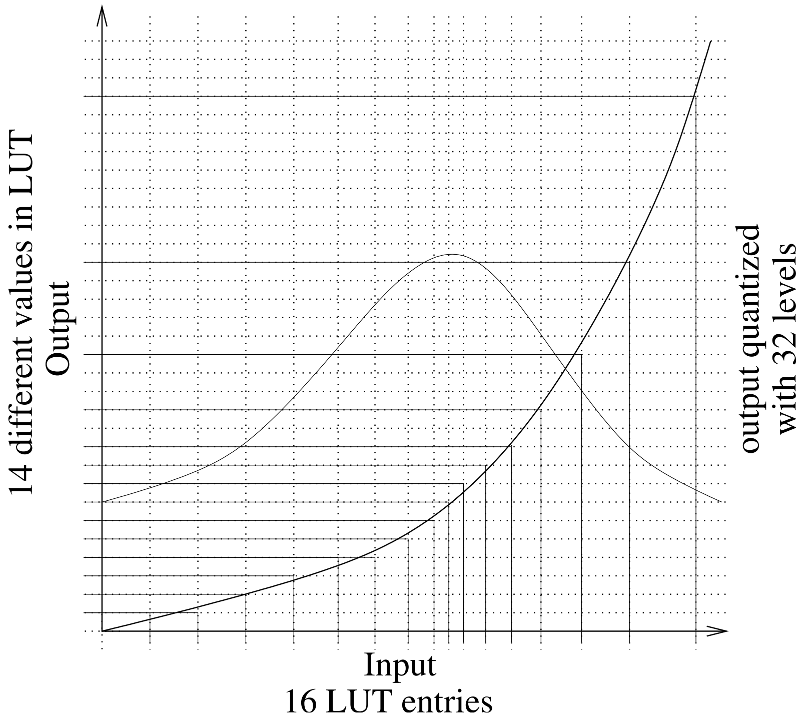
One method to implement this is to fill the LUT with equispaced values and assign the addresses to the entries according to the inverse predistortion function. Figure 9.5 illustrates the method. As can be seen, it is possible to implement a 16-entry LUT with a 4-bit output accuracy without overlapping any of the entries. However, this requires infinite accuracy for the input data and a translation table is required to generate the LUT index. The added hardware may cause the reduction of the LUT size to be in vain.

Also, the dependence of the linearity on the statistical distribution of the input signal may be used to reduce the ACP without increasing the number of LUT entries. It can be assumed that, the more infrequent the input value, the less effect the accuracy of the linearization of that value [78, 108]. Thus one should concentrate on the LUT entries to the most frequent input values and let the entries be more sparsely spaced at the less probable value ranges. Figure 9.5 illustrates a hypothetical probability distribution of the signal, concentrating the LUT entries values to the most probable input signal values. Again, to be fully effective, this method assumes high precision input signals and a translation table for the LUT address.

The effect of LUT entry distribution on a complex gain predistorter and an optimal LUT addressing function based on the amplitude probability distribution and the derivative of the signal is discussed in references [78, 108]. Cavers et al. [108] consider the effect of μ-law, power-based, amplitude-based and optimal LUT indexing. The conclusion of the paper is that the optimal



(a) LUT indexing based on constant output value step



(b) LUT indexing using probability distribution

Figure 9.5: LUT indexing methods

indexing is optimal only for a defined input back off level and, even then, it is only marginally the best method. The μ-law and the power-based indexing are also found to work poorly compared to the amplitude-based indexing.

However, results reported by Lin et al. [168] show that actually quite good results are achievable for the optimal indexing at some optimization points. It is suggested to improve the optimal indexing by changing the optimization point dynamically according to the backoff, which seems to give overall optimal results [168]. Still, the unconditionally optimal indexing method is dependent on the probability distribution of the signal amplitude.

Both Muhonen et al. and Boumaiza et al. [78, 110] present LUT indexing methods that are not dependent on the PDF. These seem to give fairly good results. However, both still require a complicated translation function to generate the LUT index, which may affect the feasibility of the indexing scheme.

Hassani et al. [109] suggest a method to generate the indexing function with low hardware costs. The method is based on dividing the LUT into a small number of segments that contain different numbers of entries, more in areas where the derivative of the predistortion function is large and less in areas where the derivative is small. The amplitude is then divided into the same number of sections, each of which contain an equal amplitude span. These amplitude spans are then mapped to the corresponding sections in the actual LUT. The result is a piecewise approximation of the wanted indexing function that can be implemented with a small memory and shifters and adders if the LUT sections are properly selected. This method can be used to approximate the indexing functions used in, for example, references [78, 108, 110, 168] with low hardware costs.

The envelope detector used in the RF-predistorters inherently implements a nonlinear LUT indexing function and has similar results to the operation of the predistorter. The effect of the detectors on the predistorter has been discussed more thoroughly in Chapter 6.

9.4 LUT generation methods and interpolation

There are also several different methods to generate the actual LUT entries; the selection of any particular method affects the ACP and the complexity of the predistorter. The LUT generation methods can be divided into two main categories: those that use a piecewise constant approximation of the predistortion function [11, 79, 91, 114, 134, 179] and those that use an intermediate function (e.g. a polynomial) to approximate the predistortion function before generating the piecewise constant approximation [56, 105, 161]. The intermediate functions allow a smaller number of parameters during the calculation of the LUT values but reduces the accuracy of the predistortion function.

Related to the LUT fill using an intermediate function is interpolation [91, 130] of the LUT output values. Interpolation can be used to improve the approximation of the predistortion function by calculating intermediate values between the LUT entries using some interpolation function. Using interpolation can reduce the effect of a limited number of LUT entries on the linearization ability. The advantage of interpolation over LUT fill with an intermediate function is the smaller size of the LUT, but this comes at the cost of increased computational complexity on the predistorter signal path.

9.4.1 Piecewise constant approximation

The piecewise constant LUT fill method is the most flexible and most straightforward method to fill the LUT. It allows any shape of predistortion function within the limits of the LUT size and word length. This method also allows the separate update of the LUT entries using some simple root finding algorithm such as the secant or linear methods (Section 3.5.2) in adaptive solutions. The disadvantage of this method is the computational complexity when used in frequency-domain update methods (Section 4.4.2), due to the large number of coefficients updated simultaneously. However, improvements have been suggested that reduce the complexity of the update ([134]).

9.4.2 Intermediate functions

The use of an intermediate function to calculate the values in the LUT can be used to reduce the number of coefficients for the LUT estimation, either in the case of frequency-domain feedback [56, 102, 105, 153] or for estimation of the predistortion function during the initial PA characterization[161]. The use of intermediate functions can significantly reduce the number of required coefficients for a simultaneous update of all the LUT entries using, for example, an RLS or LMS algorithms. This reduces the complexity of the algorithm and speeds up the convergence. However, the fewer the coefficients, the worse the approximation. Additionally, the quality of the approximation depends on the shape of the predistortion function and the selected intermediate function.

There are several suitable functions that can be used as the intermediate function. The polynomial approximation (Section 3.4.1) is probably the most common [56, 102, 105, 153] and offers a low computational complexity for the calculation of the final LUT. However, the drawback is that the polynomial functions are fairly poor in compensation distortion at low amplitudes or distortion when the PA is close to saturation, requiring a large number of coefficients in these cases.

To enable more flexible approximation, splines [161, 180] can be used. Another function similar to the splines is the Hermite polynomial [181]. Both of these increase the computational complexity of the LUT fill operation compared to the polynomial functions, but they are also able to approximate complex functions with a lower number of coefficients. Both the functions use a small number of LUT entries as parameters and generate the final LUT by interpolation. The spline and Hermite polynomial have similar computational complexities, but the spline performs better than the Hermite polynomial when the approximated function is smooth, as it makes the second derivatives of the interpolated function continuous, whereas the Hermite polynomial requires only continuity of the first derivative and does not generate overshoot when the function is not smooth [182].

Also, simpler intermediate interpolation functions, such as linear interpolation, can be used. These methods, however, pay for their simplicity with worse accuracy.

The limitations of the polynomial intermediate functions in approximating the nonlinearities at the low amplitudes and near saturation may become a problem when high linearity is required. High polynomial order is required to be able to approximate these distortions accurately.

There is also another problem with the polynomial approach connected to the adaptive LUT. Namely the polynomial coefficients may have to be constrained to limit the required search space and to reduce the number of iterations. When there is no information about the PA nonlinearity, the parameter space the coefficients span is very large and not easily constrained.

The spline and Hermite polynomial functions use a number of actual LUT values as the parameters. The parameter space is more easily constrained, due to the fact that each LUT entry value is limited by the maximum and minimum achievable with the word length.

As the PA nonlinearity is usually concentrated at either or both ends of the amplitude scale, it should be feasible to select the LUT entries that are used as the interpolation parameters in such a way that they are more densely spaced at the ends and sparsely spaced at the middle amplitude values. This can be achieved using, for example, the following formula:

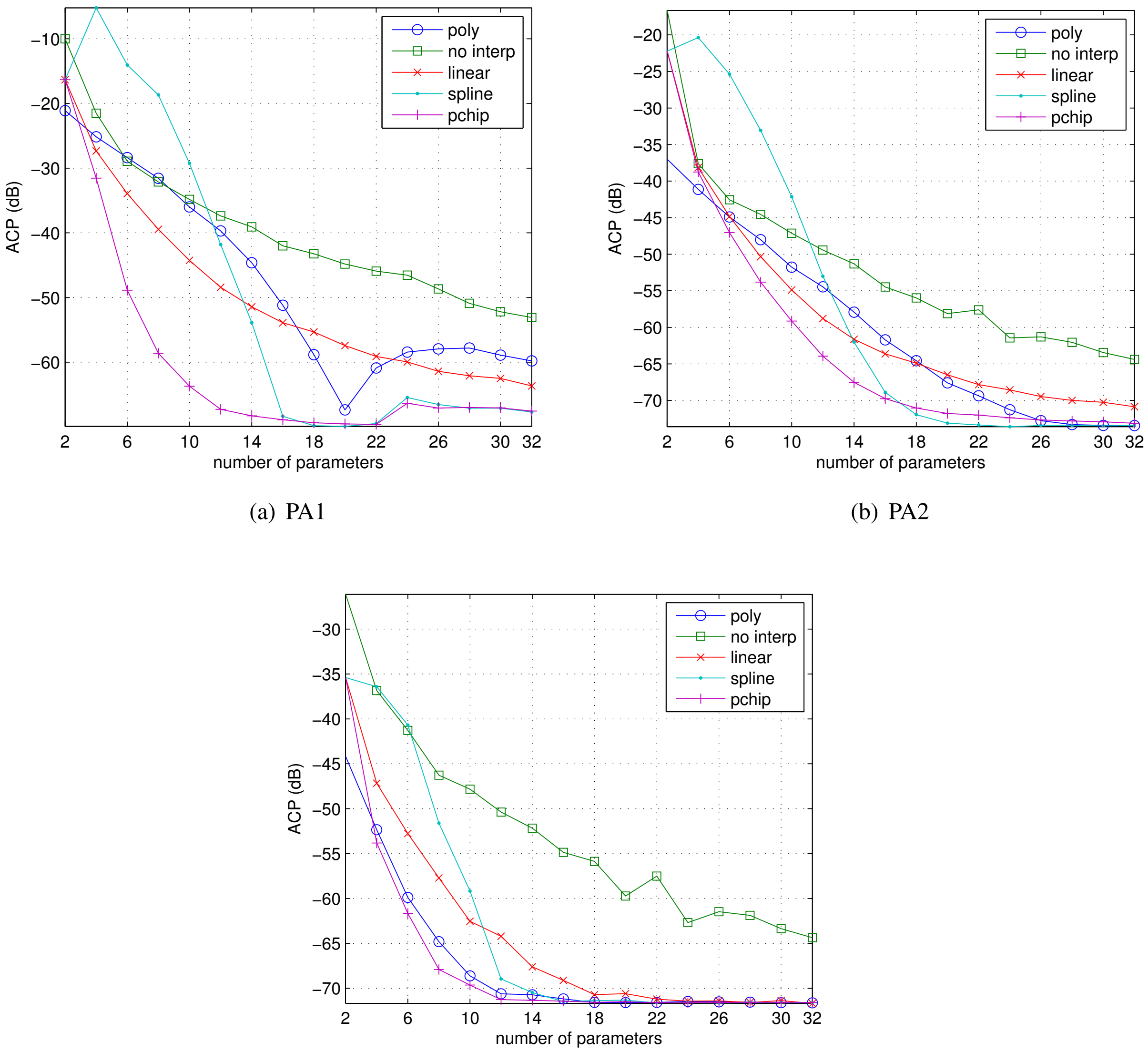
$$x_{nonlin} = (\frac{\tanh(a(x_{lin}/N_{LUT} - 0.5))}{\tanh(a)} + 1)(\frac{N_{LUT}}{2}) \tag{9.1}$$

where x_{nonlin} is the new LUT index, x_{lin} the linearly spaced index, N_{LUT} the number of LUT entries and a defines the shape of the function. This formula will be used in the following simulations due to being a simple and general closed-form formula. However, any formula or mapping having the similar properties would be suitable. The most efficient solution would be to use a custom mapping function for each different PA, but this would reduce the generality of the predistorter.

Figure 9.6 shows a comparison between polynomially approximated, non-interpolated, linearly interpolated, piecewise cubic Hermite polynomial interpolated (pchip)[181, 182] and cubic-spline interpolated [180, 182] LUT approximations when using complex gain predistortion. The results plot the adjacent channel power (ACP) as the function of the number of parameters when using a 16QAM signal and a 256-entry LUT. The parameters of the interpolation functions were distributed using (9.1). The power amplifier models PA1, PA2 and PA3 (Section 2.7) were used.

The first thing that can be seen from the figures is, that, in all cases, the interpolation reduces significantly the number of parameters required for a particular ACP. Secondly, it is clear that, when the amplifier nonlinearity is close to a low-order polynomial (Figure 9.6c), the polynomial intermediate function approximates the predistortion function well with a low number of parameters. The pchip-function is able to achieve the same results, but its larger complexity makes the polynomial a more appealing solution. The other methods are clearly inferior in this case.

When the nonlinearity on the low amplitudes increases, the polynomial function starts to require more parameters to be able to approximate the predistortion function. In Figure 9.6b the polynomial function gives the worst results, losing even to linear interpolation. The pchip method requires the lowest number of parameters for good linearity. Finally, in Figure 9.6a, the pchip method requires only half the number of parameters compared to the polynomial interpolation. In all the cases, the spline interpolation proves to require a quite large number of parameters, even though it gives somewhat better results in the cases of PA1 and PA2 than the polynomial. This is due to the smoothness requirement of the spline function that causes oscillation to the interpolated function when the number of parameters is small.



(a) PA1

(b) PA2

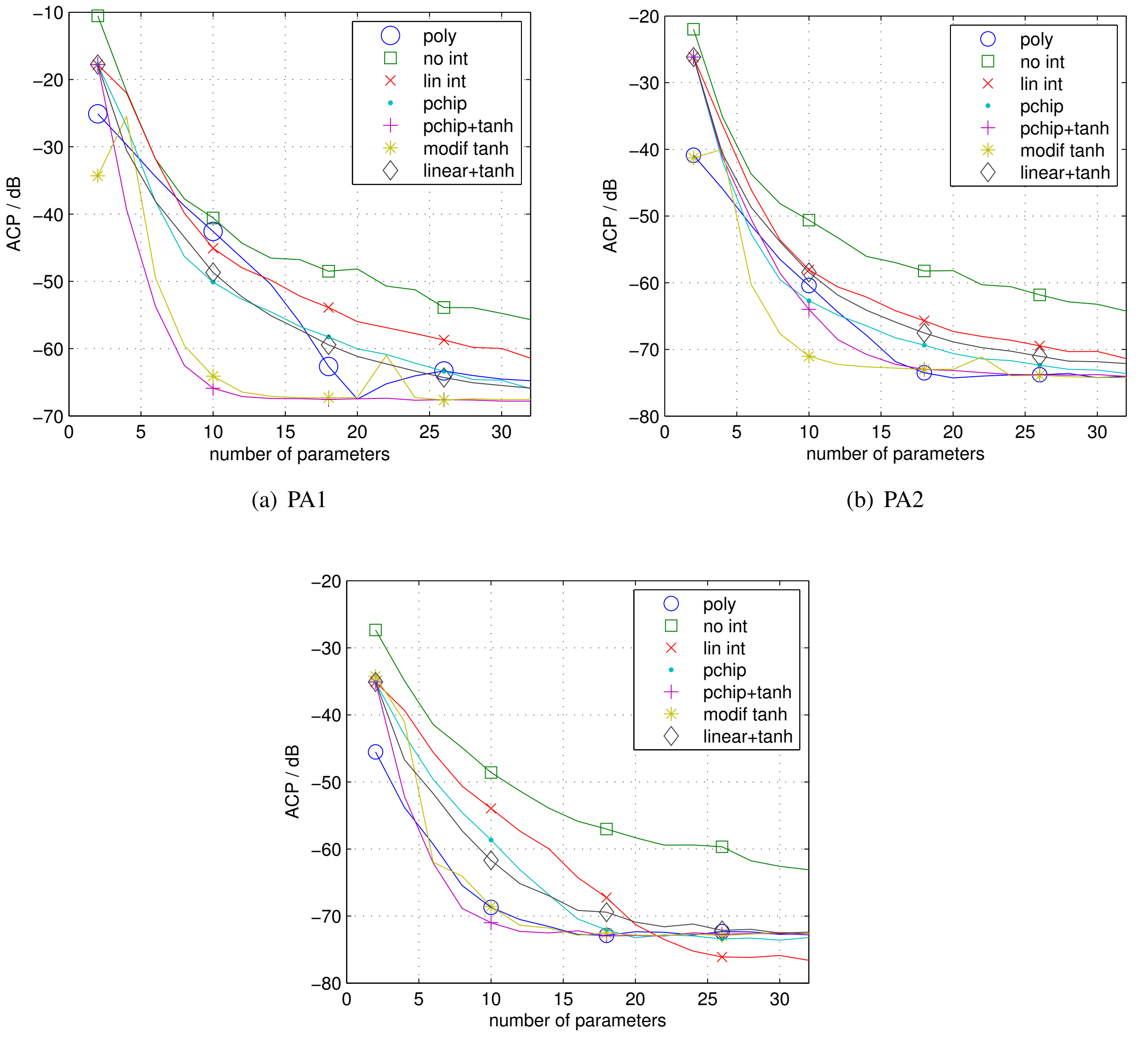
(c) PA3

Figure 9.6: Comparison of different LUT generation methods (complex gain predistorter)

Figure 9.7 shows the similar comparison for the phase amplitude RF-predistortion (Section 4.4.1). The main differences are the shapes of the predistortion functions and the detector that affects the distribution of the entries. The figures also include the simulated ACPs for the pchip and linear interpolation when (9.1) is not used but the entries are distributed evenly (pchip and lin int in Figure 9.7, respectively).

When PA2 was simulated, it was noted that, when the parameters were distributed with (9.1), the interpolation methods had some difficulties in modeling the mid-amplitude values, due to the curvature of the predistortion function. It was noted that, by adding a parameter at the mid amplitudes, the situation could be remedied. This method is marked as modif tanh in Figure 9.7). We can see from the figures that the use of (9.1) for distribution of the parameters improves the results significantly. By using the tanh distribution it is possible to implement linear interpolation with a low number of parameters, which offers a significant reduction in the required computational complexity; also, the results achieved with the pchip interpolation improve considerably.

The polynomial method works well with PA3 as expected. With PA2, the results with the polynomial function are worse than with the pchip+tanh method. By adding the one anchor point in the middle of the LUT, the performance of the pchip method can be significantly improved in the case of PA2, without affecting the results in the cases with other PAs.



(a) PA1

(b) PA2

(c) PA3

Figure 9.7: Comparison of different LUT generation methods (phase-amplitude RF predistorter)

In conclusion it can be stated that, if the PA is known to exhibit polynomial nonlinearity, the polynomial intermediate function is the most beneficial. If the predistorter is supposed to be general, the pchip interpolation offers as good or better ACP with a lower number of parameters than the other methods. However, the computational complexity of the calculation of the interpolated values is larger than with the polynomial. Also, the benefit of a proper distribution of the anchor points to interpolating the intermediate function is clear. The more information we have on the nonlinearity, the more effectively we can choose the anchor points. Finally, it can be seen that, with a proper distribution of anchors, linear interpolation can be used as a low-complexity solution if the target ACP is modest.

To study the effect of the number of parameters on the convergence of the adaptive algorithm, the polynomial intermediate function and the pchip algorithm with anchor points distributed with (9.1) and an anchor added to the middle of the LUT were simulated with different numbers of parameters. PA model PA2 was used and Nelder-Mead algorithm [158] was used for the LUT update. The phase and amplitude LUTs were updated separately to reduce the number of the parameters and improve the convergence (Section 9.6).

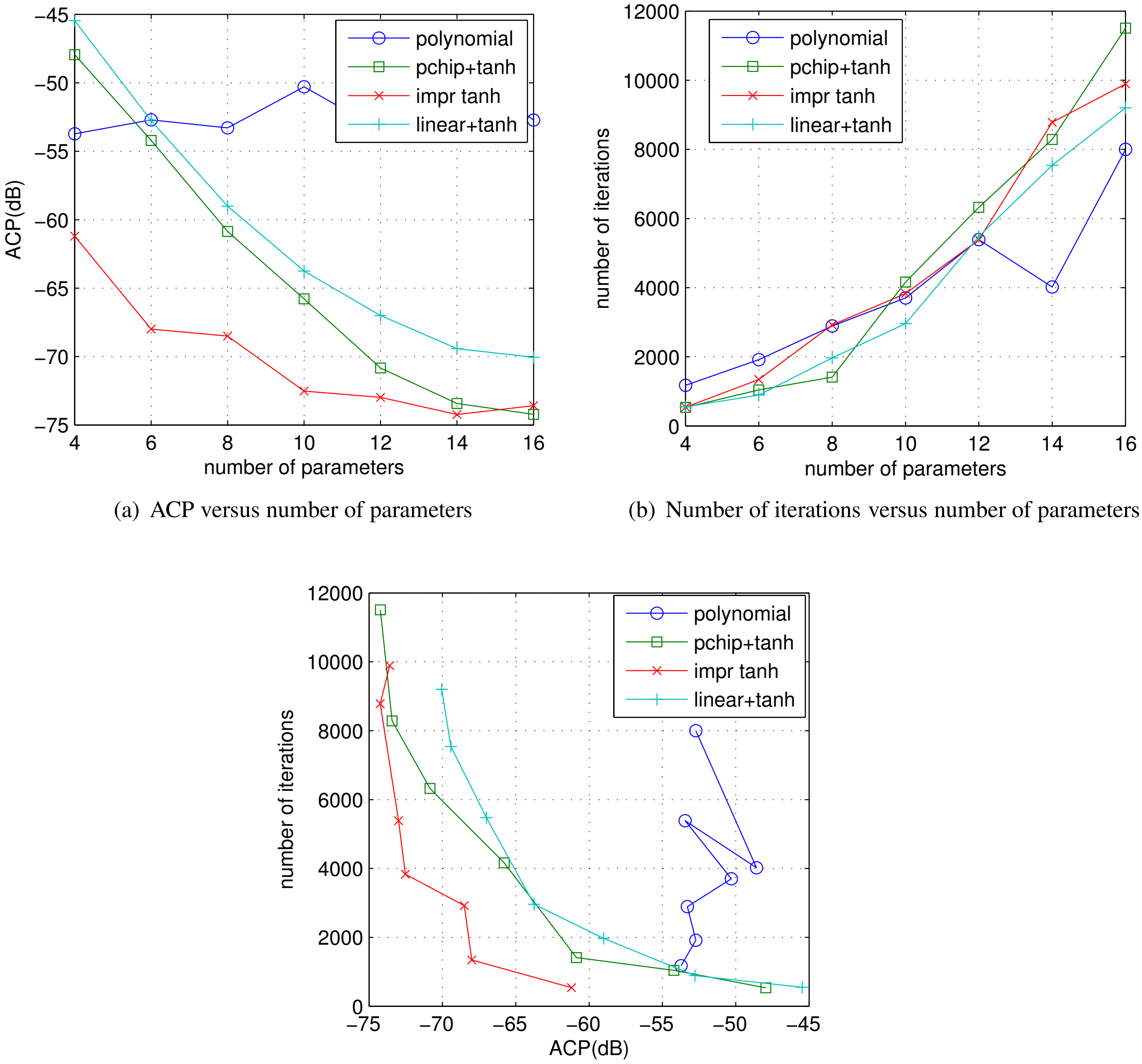
The results are collected into Figure 9.8. The figure shows that the polynomial function has large difficulties in achieving the optimal result due to the large parameter space, and thus it does not benefit much from increasing the number of parameters. The linear interpolation with anchor points distributed with (9.1) and the pchip with one anchor added into the middle of the LUT perform better than was expected on the basis of the results of the simulations with the static LUT (Figure 9.7). This is due to the fact that the iterative algorithm optimizes the parameters to improve the approximation of the predistortion function compared to the case where the parameters were chosen directly from the inverse transfer function of the PA.

As can be seen from Figure 9.8b, the number of iterations increases at an almost equal rate for all of the functions as the number of parameters increases. Thus it would be beneficial to have as low a number of parameters as possible. Figure 9.8c shows the number of iterations as a function of the ACP. What can be seen is the rapid increase in the number of iterations as the ACP improves. Also the significant advantage of the improved pchip algorithm over the other method is clearly visible.

9.4.3 Interpolation

Interpolation is an operation very similar to filling the LUT using an intermediate function. However, instead of having a large LUT that is filled using a function with a low number of parameters, a small LUT is used and the LUT output is interpolated using a suitable function. Linear interpolation, splines and Hermite polynomials (section 9.4.2) can be used also for the LUT output interpolation. Interpolation has been used in several published predistortion systems [91, 130, 183–185].

The purpose of the interpolation is to reduce the linearity and noise floor limits imposed by the small LUT by adding values to the LUT output between the steps. However, the interpolation adds a computational operation after the LUT read, which may mean that it is necessary to add



(a) ACP versus number of parameters

(b) Number of iterations versus number of parameters

(c) Number of iterations versus ACP

Figure 9.8: Comparison between the complexity and efficiency of several intermediate functions when using N-M iterative algorithm.

pipelining to be able to implement the operation, thus increasing the latency. This is especially probable when using Hermite polynomials, splines or other more complex interpolation methods. In the baseband predistortion systems the additional latency may be acceptable, depending on the system specifications, since the latencies of the data signal and the predistortion signals can be easily matched. However, as the RF-predistortion is very sensitive to delay (Section 5.2), the increased delay of the predistorter control signals due to the interpolation may do more harm than is gained by the interpolation.

Due to the simplicity, the linear interpolation is the most common method [91, 130] for LUT interpolation, although spline interpolation has also been used [184, 185]. The linear interpolation scheme can be described with the formulas [91]

$$LUT_{intp}(n) = LUT_k(n) + frac(2^{bit}\,|V_{in}(n)|) \cdot (LUT_{k+1}(n) - LUT_k(n)) \tag{9.2}$$

and

$$V_{pred}(n) = V_{in}(n) \cdot LUT_{intp}(n). \tag{9.3}$$

where $V_{in}(n)$ is the original complex input signal normalized to have absolute value between 0 and 1, $LUT_k(n)$ is the k^{th} value stored in the LUT, $k = int(2^{bit}\,|V_{in}(n)|)$, int function returns the integer part of a number, $frac$ returns the fractional part, bit is the LUT address word length, n the time instant and $V_{pred}(n)$ the predistorted signal.

One thing that should be noted when using an interpolation scheme with an adaptive LUT is, that the predistortion signal consists of a weighted sum of two or more LUT entries, thus the LUT update value should be "deinterpolated" between two or more LUT entries to take into account their different weights [91]. If we are using a linear update method (3.11), the LUT update with deinterpolation becomes [91]

$$\begin{gathered} LUT_{k+1}(n) = LUT_k(n) + \\ (1 - frac(2^{bit}\,|V_{in}(n)|)) \cdot a\frac{(V_{out}(n) - V_{in}(n))}{V_{out}(n)} LUT_{intp}(n) \end{gathered} \tag{9.4}$$

and

$$\begin{gathered} LUT_{k+1}(n+1) = LUT_k(n+1) + \\ frac(2^{bit}\,|V_{in}(n)|) \cdot a\frac{(V_{out}(n) - V_{in}(n))}{V_{out}(n)} LUT_{intp}(n+1). \end{gathered} \tag{9.5}$$

where the notation is the same as in Equations (9.2), (9.3) and (3.11).

Figure 9.9 shows the simulated improvement in output ACP and SNR of a linearized power amplifier due to linear interpolation, when the LUT has a 6-bit address and PA2 is used. The results are shown as a function of the number of bits used for the interpolation or the increase in the virtual address word length, i.e. the 2 bit increase in word length means that the virtual address word length is 8 bits, the real LUT address word length is 6 bits and the LUT output is interpolated using two bits between the adjacent LUT entries. The 'no interp' curve displays the resulting ACP and SNR if the LUT size is really increased and no interpolation is used. It can be seen that the interpolation improves especially the SNR of the system and the differences between the methods are small. For the ACP improvement, spline and pchip methods work nearly as well as the increased number of entries.

Figure 9.10 shows the simulated spectrum for a 16QAM signal using a complex gain predistorter and PA2. Linear interpolation was used (Equations 9.2 and 9.3). The LUT was updated adaptively using (9.4) and (9.5). The improvement in wide band noise due to interpolation is clearly visible, even though nonidealities limit the ACP improvement.

9.5 Methods for LUT update

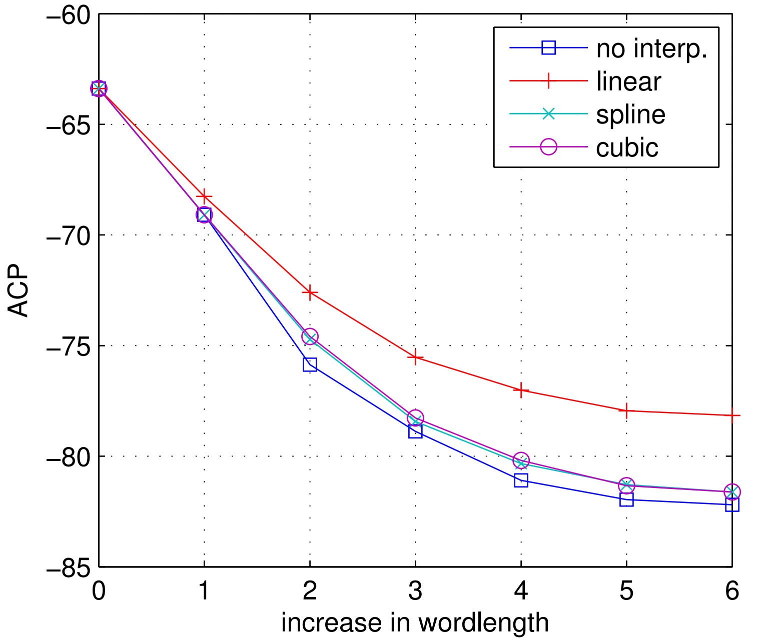
The LUT update methods can be divided into time-domain methods that use a measure of error in time-domain for calculation of updated LUT values and to frequency-domain methods that use a frequency-domain measure of error.

9.5.1 Time-domain methods

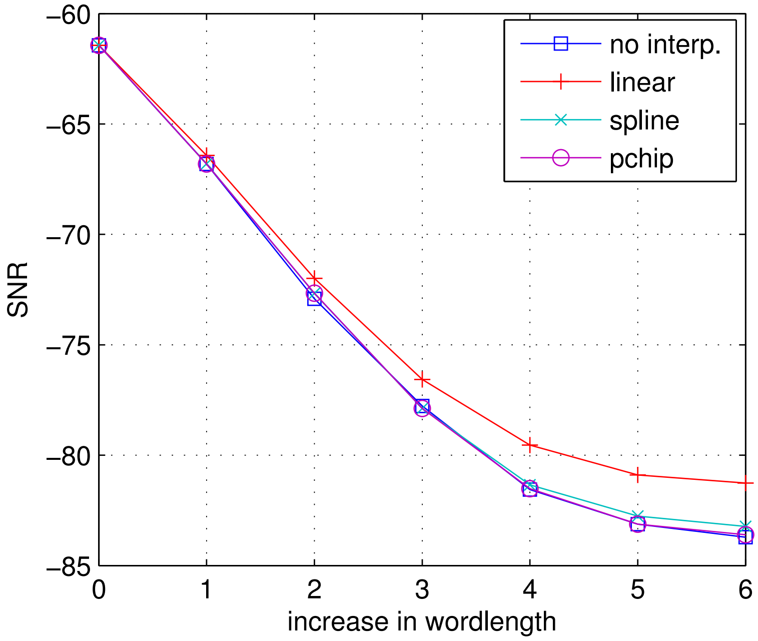
The time-domain update methods use information extracted from instantaneous signal measurements and update the LUT values in real time. If the predistortion function is updated directly, the most suitable update methods are the simple root finding methods such as secant or linear iteration which are discussed more thoroughly in Section 3.5.2.

Other commonly used methods are the LMS [114, 121, 125, 153], RLS [102, 120, 124, 125, 154] and cross-correlation [123] methods. These are commonly used in conjunction with polynomial [120, 153] and memory predistortion [121, 124, 154], since the algorithms are originally designed for adaptive filtering and the polynomial evaluation is easily transformed into a form similar to that of the FIR-filtering. These methods, however, have several disadvantages. If the predistortion function is to be identified directly by the linearity of the PA output, one has to know the Jacobian matrix of the distortion function [126]. This means that the nonlinearity of the power amplifier has to be known before the predistortion function can be identified. This makes the direct-learning architecture quite complex [121, 126, 128, 186]. There has been some proposals to reduce the complexity of the direct learning algorithm [121, 127, 128].

However, a more common method is to use the indirect learning method [47, 54, 120, 124, 126, 153, 154] that uses a postdistortion block that is identified to remove the nonlinearity from the measured PA output. This identified linearization function is then copied to the input of the PA. This operation requires that the order in which the distortion and predistortion functions are applied is irrelevant. However, this is not usually the case and optimal linearization can not be achieved [126, 128]. The method also suffers from noise in the PA output [126, 128]. The third problem is that this method is suitable only for low order polynomial predistortion or simulations, due to the fact that, as the polynomial order increases, more and more computational accuracy is required, and, if an LUT is used to contain the polynomial, the whole LUT has to be updated every clock cycle, which is very arduous. The problems are discussed more thoroughly in Chapter 2.8.



(a) The level of first ACP as a function of number of interpolated bits



(b) The level of out of band noise as a function of interpolated bits

Figure 9.9: The improvement in ACP and SNR due to interpolation with a 6-bit LUT address

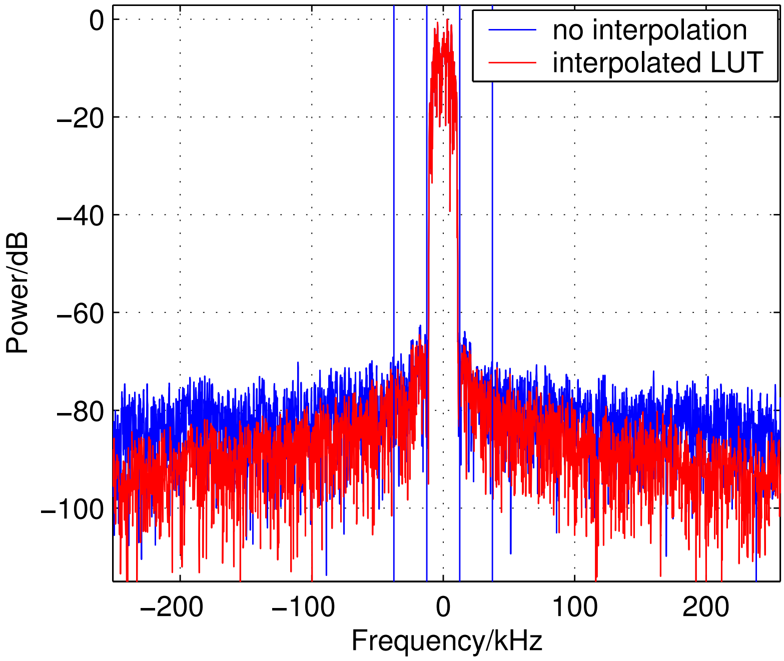


Figure 9.10: Effect of LUT interpolation on wide band noise with adaptive LUT

9.5.2 Frequency-domain methods

The frequency-domain LUT update methods periodically update the LUT values, usually the whole LUT at once, according to a single linearity metric based on measurements over a long time period. Often this metric is based on frequency-domain related measurements, such as adjacent channel interference or the MSE of the PA output signal compared to the input signal (Section 4.4.2).

There are several methods that can be used to implement the frequency-domain LUT update. Most of these methods benefit from limiting the number of parameters required for calculation of the new LUT entries, using an intermediate function [28, 56, 102, 105, 153] or some other method [134].

Among others, predistorters using the Hooke & Jeeves' [28, 187] and genetic algorithms [56, 105, 159] for the frequency-domain update have been published. Other possible algorithms are, for example, gradient search algorithms, simplex algorithms, such as Nelder-Mead [158] and simulated annealing [160]. The genetic algorithm has been used also in predistorters with memory [56].

The advantage of the frequency-domain update is that it requires only a narrowband feedback [28], which simplifies the requirements for the feedback path. The used update methods are fairly complex when implemented with hardware and thus the frequency-domain update requires a processor for software implementation of the algorithm.

9.5.2.1 Genetic algorithm

The genetic algorithm will be used in the following analysis in addition to the Nelder-Mead algorithm as the iterative algorithm. The genetic algorithm was selected as an example of heuristic minimization methods and was chosen over simulated annealing due to the availability of ready-made functions for Matlab(R) and to ease comparison to previous published designs [56, 105]. The basic operation of a genetic algorithm is presented in references [56, 105, 159, 188].

9.6 Improvements for the frequency-domain update methods

Although the frequency-domain update methods are fairly straightforward to implement, there are several pitfalls in the implementation that may limit the convergence of the algorithm. By tackling these problems, the effectiveness of the linearization scheme can be improved.

Sperlich and Sills [56, 105] have presented a frequency-domain-feedback-based predistorter using a genetic algorithm. The algorithm has some limitations due to the polynomial function used. Use of this function means, that the parameter space for the polynomial coefficients has to be constrained to improve convergence. This, on the other hand, means that prior information of the PA nonlinearity is required to generate the constraints. Additionally, the constraint functions are fairly complex. However, as was explained in Section 9.4.2, if the actual LUT values are used as parameters for the intermediate function, the values are inherently constrained. Also, the use of some more flexible function allows better linearization ability with non-polynomial distortion functions.

In this section, our goal is to improve the frequency-domain update methods so that they can be implemented without knowledge of the shape of the predistortion function and that the actual LUT entries could be used as parameters for the algorithm. This would improve the generality and flexibility of the algorithms and allow the development of a universal predistortion circuit.

The first thing that should be considered is the selection of the parameter that is used as the error metric. There are two quite obvious possibilities for the error metric: the RMS of the difference of the PA input and output and the ACP of the PA output signal. Also, depending on the required specifications, other parameters such as EVM may be used.

If the RMS error is used, then the delay differences between the signals affect the result (Sections 4.4.2 and 5.3), and this should be taken into account in the design.

If the adjacent channel power is used as the metric, there is no requirement for matching the delays of the original signal and the amplified signal for the error calculation. However, this presents another problem: if only the ACP is used, we have no reference on the power level of the signal and usually the ACP can also be reduced by reducing the signal power in addition to generating the inverse nonlinearity of the PA. This, however, reduces the efficiency and thus is undesirable. Without taking account the original power level we risk convergence into a suboptimal state. To reduce the possibility that the adaptation converges to a state where the predistorter improves the ACP mainly by reducing the output power, the objective function can be modified to take into account the desired output power level. This can be achieved by, for example, setting the objective function to

$$ObjV = ACP + A_{power} \left| P_{target} - P_{current} \right| \tag{9.6}$$

where P_{target} is the wanted signal power in dB and $P_{current}$ is the signal power achieved by the current individual in dB and A_{power} defines how hard the algorithm tries to achieve the target power.

The next problem is that usually the amplitude distortion of a PA is the dominating error source and the phase distortion is overshadowed by it. This causes that if both the phase and amplitude parameters are updated simultaneously only the improvements caused by the amplitude coefficients are noticed. Thus, half of the parameters are updated in vain and the computational cost and the size of parameter space increase unnecessarily, slowing down the convergence. This also increases the chance that the iteration converges to a suboptimal state.

This makes it beneficial to update the phase and amplitude LUTs separately. However, after both LUTs have converged separately to find out the approximate form of the nonlinearity it was found out during simulations to be feasible to do a final iteration for both the LUTs at the same time. Sperlich et al [56] suggest that a similar separation of updates should be performed in memory predistorters between the LUTs containing the polynomials corresponding to different time instants.

Another problem similar to the previous one is that, when the adaptation is started, the high amplitude values have a much larger effect on the ACP than the low amplitude values. This is due to the fact that the power amplifier usually is driven near saturation and thus the nonlinearities at the high amplitudes are strong and the energy of the signal is also larger at the high amplitudes. This causes the low amplitude values tend to converge on a suboptimal state as when the effect of the low amplitudes starts to be visible, the algorithm has already reduced its search space to exclude the optimal values. This problem can be alleviated by iterating the parameters corresponding to the low amplitude values after the convergence of the original search.

Table 9.1 compares the effect of the improvements presented in the previous paragraphs. The Nelder-Mead (N-M) [158] algorithm was used as the iterative algorithm due to its availability in Matlab. Comparisons were made using PA2 linearized with the amplitude-phase RF-predistorter (Section 4.4.1).

In the table, “poly” is the method where the parameters are polynomial coefficients, mod pchip is the simultaneous iteration of phase and amplitude LUTs when the parameters are selected using (9.1) with one coefficient added in the middle of the LUT (Section 9.4.2), “phase+amp” introduces the separate phase and amplitude iterations and finally “low amp iter” uses also the separate adaptation for a small number of the lowest parameters, “params” tells the number of parameters used, “ACP” tells the final ACP value achieved with the method and “ACP eval” tells the required number of evaluations of (9.6) before the convergence. All the simulations were averaged over ten runs. The number of parameters for polynomial and pchip algorithms were chosen so that they achieve approximately 65 dBc ACP in Figure 9.7. All the configurations were tested with Nelder-Mead algorithm and additionally the original modified pchip and the case with separate low amplitude iteration were tested with a genetic algorithm with 40 individuals. The results are collected in Table 9.1.

As can be seen, without constraints, the polynomial predistorter works badly and converges quickly to a suboptimal state. The addition of separate phase and amplitude iterations somewhat improves the results. When the iteration was changed to pchip, the results improved. The use of pchip allowed the LUT entries to be used as parameters, which improved the accuracy of the approximation and reduced the parameter space. With only eight parameters, the target ACP is achieved and the number of iterations reduces by a quarter compared to the polynomial. If we increase the number of parameters to twelve, the ACP improves, but the number of iterations doubles. When the phase and amplitude iterations are separated, the number of iterations reduces. By adding the separate iteration round for the low amplitudes, we retain the ACP, but reduce the number of iterations by 200.

When the results for the genetic algorithm are examined, we see that the algorithm achieves 70 dBc ACP. By introducing the separate phase and amplitude and low amplitude iterations, the number of iterations reduces by 20%. It should be noted that the number of iterations and the ACP achieved by the genetic algorithm varies, depending on the number of individuals in the population and thus the results are allusive.

Table 9.1: Comparison between different improvements to the iterative LUT-update algorithm

	poly	poly, ph+amp	mod pchip		ph+amp	low amp
params	12	12	8(N-M/gen)	12	8	8(N-M/gen)
ACP	50 dBc	52 dBc	70/70 dBc	73 dBc	70 dBc	69/70 dBc
ACP eval	5800	4200	3400/4300	8100	2700	2500/3500

As can be seen the proposed improvements reduce both the ACP and number of iterations. It should also be noted that the differences between the hardware costs of the different versions of the pchip algorithms are fairly small, since only some additional logic for distinguishing the different update phases is required.

9.7 LUT-based genetic algorithm

As the simulations of the improved algorithm with the Nelder-Mead algorithm were promising, an improved genetic algorithm using the changes presented in the previous section was implemented for simulations.

The algorithm was implemented in such a way that the ten all-time best individuals were stored between the iteration rounds. This was done to preserve the results achieved during the previous iteration rounds. At the start of each round, the new population is generated by taking the all-time best individual and adding random perturbation to the parameters. The size of the perturbation is divided by two every round. This is done to reduce the space spanned by the population. Finally, ten of the generated individuals are replaced by the ten all-time best individuals. The population size was chosen to be 40 individuals.

Thus the modified algorithm becomes

1. Generate population with N individuals and X amplitude LUT parameters
2. Set the X phase LUT parameters to 0
3. Evaluate the objective function (PA-predistorter chain) with the individuals
4. Store the individuals that are better than the previous ten best individuals
5. Rank the fitness of the individuals with nonlinear ranking[188]
6. Select the most fit individuals
7. Cross the selected individuals to generate offspring using linear recombination[188]
8. Mutate the offspring with breeder genetic algorithm[188]
9. If the number of iterations <10 goto step 3 else go to 11
10. If the fitness of the best individual of the previous loop does not exceed the fitness of the all-time best individual, change from amplitude LUT to phase LUT or vice versa
11. Generate new population by adding random perturbation to the all-time best individual and reinsert the best 10 individuals. On every other amplitude round, generate population only by perturbing the lowest three coefficients of the best individual.
12. If optimization criteria are not met, goto step 3

9.8 Simulation results

Table 9.2: Simulation results of the genetic algorithms and approximated LUTs

		no correct	PA inverse	cubic interp.	polynomial	genetic LUT		genetic poly		Nelder-Mead	
PA	detector	ACP	ACP	ACP	ACP	ACP	Niter	ACP	Niter	ACP	Niter/40
PA1	powdet	30 dBc	60 dBc	54 dBc	51 dBc	56 dBc	87	49 dBc	88	56 dBc	78
PA2	powdet	35 dBc	68 dBc	62 dBc	65 dBc	64 dBc	57	56 dBc	48	64 dBc	125
PA3	powdet	28 dBc	75 dBc	71 dBc	75 dBc	68 dBc	123	65 dBc	81	78 dBc	65
PA1	lindet	30 dBc	67 dBc	60 dBc	50 dBc	59 dBc	116	50 dBc	55	62 dBc	75
PA2	lindet	35 dBc	74 dBc	67 dBc	68 dBc	70 dBc	87	56 dBc	105	69 dBc	63
PA3	lindet	28 dBc	72 dBc	64 dBc	72 dBc	68 dBc	56	60 dBc	97	72 dBc	55

The genetic algorithm was simulated with Matlab using the RF-predistortion system presented in chapter 7. The phase and amplitude LUTs were indexed with envelope measured using two different envelope detector types, namely a power detector (powdet) and linear diode detector (lindet) (Chapter 6). PA1, PA2 and PA3 were used in the simulations to test the operation with different predistortion function shapes.

The signal used in the simulations was a 300 ksym/s 16QAM signal and the clock frequency was 10 MHz. The envelopes and the predistorter control signals were filtered with a fourth order Butterworth filter with a corner frequency of 5 MHz. The LUT size was 256 entries and the word length was 12 bits. The LUT based genetic algorithm described in Section 9.7 was implemented. The number of parameters was set to be 8, based on the results in Section 9.4.2. Also a polynomial-based genetic algorithm with 14 parameters was implemented. Finally, a N-M algorithm [158] -based LUT iteration algorithm was implemented for comparison.

The resulting ACPs are shown in Table 9.2. Results are averaged over five runs to take into account the fact that the algorithms are heuristic. The table also lists the predistortion results in the case where the LUTs contain the calculated inverse transfer function of the PA, the case where this predistortion function is approximated with a LUT interpolated with pchip from eight parameters and the case where the function is approximated with a 13th-order polynomial.

The results clearly show that neither the cubic interpolation nor the polynomial approximation can achieve as good results as the PA inverse transfer function. Nevertheless, the results are fairly good in the cases with PA2 or PA3.

When the LUT is updated with the pchip based genetic algorithm, it can be seen that the results match, or even exceed, the results expected according to the pchip interpolated static LUT case. This is due to the fact that the iterative algorithm can adjust the parameters to a slight offset from the actual value to improve the overall predistortion function. As could be expected from the results in Section 9.4.2, the polynomial function has severe problems with the convergence to an optimal state. Better results might be achieved if the starting population were based on the measured nonlinearity of the power amplifier. However, the LUT-based algorithm is not dependent on the knowledge of the PA nonlinearity.

Finally, when the genetic-algorithm-based and the N-M-based algorithm are compared, it can be seen that they both achieve equally good results. Also the numbers of iterations are comparable (it should be noted that the size of population in the genetic algorithm is 40 individuals, thus the number of iterations in the N-M case is divided by 40 to match this). Thus the genetic algorithm does not offer significant advantage over the N-M algorithm in this sense. However, the necessary calculations for the LUT update have to be performed only every 40th iteration, which when compared the N-M, may make the genetic algorithm more attractive.

9.9 Conclusions

LUT size, indexing, interpolation and update are important factors in the design of a digital predistorter. In this chapter, the effect of LUT size and entry accuracy on the linearization ability of the predistortion system were discussed. The chapter also reviewed methods to reduce the LUT size without affecting the linearity. As the methods to reduce the LUT size are somewhat limited, and as some LUT update methods require a much lower number of parameters, a number of interpolation methods were also discussed. These methods can be used to significantly reduce the number of parameters used to generate the LUT, but they do so at the expense of computational simplicity and reduced linearity, due to approximation of the predistortion function. Using a cubic Hermite interpolation polynomial interpolation function gave the best linearization results when the amplifier was nonpolynomial. Also the difficulties with using a polynomial predistortion function when linearizing a non polynomial amplifier were discussed.

Finally, several LUT update methods were reviewed. A frequency-domain-metric-based LUT update method that uses a number of LUT entries as parameters and an interpolative function was presented and several improvements on the original algorithm were discussed. The method was shown to be suitable for use with the Nelder-Mead algorithm and genetic algorithm. Also, the effectiveness of the improvements was shown through simulations and the method proved to exhibit a much more reliable convergence than a polynomial function, due to its inherently restricted parameter space. The method also is fairly independent of the shape of the nonlinearity.

Chapter 10

Conclusions

Predistortion has been used as a simple and robust method to linearize power amplifiers for several decades, while the first fully digital predistortion systems were published some 20 years ago. A great deal of research has been carried out in the area. Although the digitally controlled predistorters have been shown in simulations to be capable of achieving very good linearization ability, there are still several problems related to the practical implementations that need to be solved. This can be seen by, for example, comparing the simulated and measured results in Table 3.1.

If the problems could be solved, the RF-predistortion systems would then offer the possibility to implement a universal predistortion chip that could be used with only general knowledge of the PA and baseband blocks or it would offer integration with a PA to generate a linear and efficient device.

One significant problem that hinders the development of a small-size RF predistorter is the inevitable delay differences between the baseband control signals and the RF signal to be predistorted. If these delays are not compensated, the linearity suffers significantly and, in adaptive predistorters, the update may become unstable, as was seen in the simulation results presented in this thesis. The old methods to compensate this delay were been bulky and nonintegratable into IC, but, in this thesis, a fully digital delay compensation method based on polynomial predictive filters is presented. This method was able to achieve an improvement in ACP of over 20 dB in simulations and to reduce instability of the predistorter update in measurements. The second order polynomial predictor was seen to perform the best with constant LUT implementations, especially with additional noise filtering. When using an adaptive LUT, the first order predictor is the most stable solution.

The selection of the envelope detection method for the predistortion function generation in an RF-predistortion system affects the linearization ability of the predistortion system. This thesis studied the effect of three different envelope detectors, the linear diode, square-law diode and logarithmic detectors. The linear diode detector proved to be the most versatile one and gave in all the test cases the best or very close to the best linearization ability. However, the biasing of a diode detector to the linear operating condition may be difficult. If the power amplifier exhibits nonlinearity only on high signal amplitudes, a square-law diode detector can be used. On the other hand, if the nonlinearity is on low amplitudes, the logarithmic detector operates well. It was also noted that the operation of the logarithmic and square-law detectors could offer the same linearization ability as the linear diode if the detector output was linearized in the digital domain. Since the nonlinearity functions of the different envelope detector types spread the spectrum of the detected signal differently, the bandwidth requirements for the post-detection filter were studied. The linear and power-law detectors required approximately a filter bandwidth of 3 times the signal bandwidth, whereas the logarithmic detector required a filter bandwidth of 5 times the signal bandwidth.

An RF-predistortion system was implemented. The predistorter uses an analog phase and amplitude modulator as the predistortion element and an adaptive digital control for the predistortion element. The digital control was implemented using two LUTs and digital update algorithm on an FPGA. The carrier frequency of the RF signal was 440 MHz and the signal bandwidth varied from 3 kHz to 50 kHz.

When the predistorter update was frozen during the actual signal transmission, the predistorter improved the other adjacent channel ACP by 8 dB and the other by 17 dB. However, using the adaptive update made the system unstable, due to delays in the control signals. Using a predictive polynomial filter to generate the LUT control signals improved the stability, but the instability was completely removed when the update algorithm was improved to update based on a larger number of samples and at longer intervals, ensuring that the update has affected the predistorter feedback before the next update. This achieved a 10 dB improvement in ACP on both sidebands using an 18 kHz signal bandwidth and improved stability. As the results show, there is still room for improvement in the performance of RF-predistorters.

Also, complex gain baseband predistorters were investigated in the thesis and predistorter hardware for a complex gain predistorter with a quadrature modulator and demodulator error correction circuits was designed. Different adaptation methods for the quadrature modulator error correction circuits were compared and the effect of quadrature modulator nonlinearity on baseband predistortion was investigated.

An implemented predistorter uses two LUTs containing the real and imaginary part of the adaptive predistortion function. The predistortion function is constantly updated according to the quadrature demodulated PA output signal. The predistorter was implemented using an FPGA .

The signal used in the simulations and measurements was an 18 ksym/s 16QAM signal at a 400 MHz carrier frequency. In measurements the ACP improvement was 15 dB when the quadrature correction was in use. The removal of the quadrature error correction increased the ACP by 9 dB compared to the situation without quadrature correction. The tested quadrature compensation methods gave similar results ACP-wise. The measurements clearly showed that the quadrature compensation of the feedback signal is very important to the linearization ability of the predistorter. Especially the feedback DC levels affect the correction considerably.

The simulation of the quadrature modulator nonlinearity showed that, to minimize the effect of third-order nonlinearity, the nonlinearity in both the inphase and quadrature branches should be the same. On the other hand, the effect of second order nonlinearity is minimized only when either or both second order components are completely removed. It was also shown that an adaptive predistorter is able to compensate part of the third order nonlinearity, but the second order nonlinearity causes convergence problems to the adaptation due to large residual errors; the results with adaptation are shown to be worse than without adaptation.

Finally, this thesis studied the effect of the LUT size and accuracy, indexing, interpolation and update on the linearization ability of the predistortion system. Also, methods to reduce the LUT size without affecting the linearity were reviewed.

As the methods to reduce the LUT size are somewhat limited and some LUT update methods require much lower number of parameters to be feasible, a number of interpolation methods were also discussed. These methods can be used to significantly reduce the number of parameters used to generate the LUT, at the expense of computational simplicity and reduced linearity due to approximation of the predistortion function. Using a cubic Hermite interpolation polynomial interpolation function gave the best linearization results when the amplifier was nonpolynomial. Also the difficulties of using a polynomial predistortion function when linearizing a non polynomial amplifier were discussed.

Finally, several LUT update methods were reviewed. A frequency-domain-metric-based LUT update method that uses a number of LUT entries as parameters and an interpolative function was presented and several improvements on the original algorithm were discussed. The method was shown to be suitable for use with the Nelder-Mead algorithm and the genetic algorithm. The improved method exhibited in simulations a much more reliable convergence than a polynomial function.

Bibliography

[1] E. Armstrong, “A method of reducing disturbances in radio signaling by a system of frequency modulation,” *Proceedings of the IRE*, vol. 24, pp. 689–740, 1936.

[2] A. B. Carlson, *Communication systems : An introduction to signals and noise in electrical communication*. McGraw-Hill, 1986.

[3] P. Kenington, *High-linearity RF amplifier design*. Norwood, USA: Artech House, 2000.

[4] S. Cripps, *RF power amplifiers for wireless communications*. Norwood, USA: Artech House, 1999.

[5] M. Jeruchim, P. Balaban, and S. Shanmugan, *Simulation of communication systems: Modeling, methodology and techniques*. Springer, 2000.

[6] “ETSI EN 300 392-2,” ETSI, Standard, 2007.

[7] National Telecommunications and Information Administration, “United States frequency allocation chart,” U.S. Department of Commerce, Chart 003-000-00691-3, 2003.

[8] M. Horn and A. Egger, “Design and performance of microwave predistortion networks using digital circuits,” in *Proceedings of 14th European Microwave Conference*, 1984, pp. 549–554.

[9] P. Kenington, M. Cope, R. Bennett, and J. Bishop, “A GSM-EDGE high power amplifier utilising digital linearisation,” in *IEEE Microwave Symposium Digest*, Phoenix, USA, May. 2001, pp. 1517–1520.

[10] S. Kusunoki, K. Yamamoto, T. Hatsugai, H. Nagaoka, K. Tagami, N. Tominaga, K. Osawa, K. T. S. Sakurai, and T. Iida, “Power-amplifier module with digital adaptive predistortion for cellular phones,” *IEEE Transactions on Microwave Theory and Techiniques*, vol. 50, no. 12, pp. 2979–2986, Dec 2002.

[11] J. Cavers, “Amplifier linearization using a digital predistorter with fast adaptation and low memory requirements,” *IEEE Transactions on Vehicular Technology*, vol. 39, pp. 374–382, November 1990.

[12] I. Teikari, J. Vankka, and K. Halonen, “Adaptive RF input/output power amplifier linearisation based on table look-up,” in *Proceedings of Finnish Signal Processing Symposium*, Tampere, Finland, May 2003, p. 4 pp.

[13] I. Teikari, J. Vankka, and K. Halonen, “Digitally controlled RF input/output predistortion for a class AB power amplifier,” in *Proceedings of Finnish Wireless Communications Workshop*, Oulu,Finland, October 2003, pp. 39–42.

[14] I. Teikari, J. Vankka, and K. Halonen, “Digitally controlled RF input/output predistortion for a class AB power amplifier,” in *Proceedings of NORCHIP’03*, Riga, Latvia, October 2003, pp. 204–207.

[15] I. Teikari, J. Vankka, and K. Halonen, “Digitally controlled RF predistortion based on vector modulator,” in *Proceedings of International Symposium on Signals, Systems, and Electronics*, Linz, Austria, Aug 2004, p. 4 pp.

[16] P. Heinonen and Y. Neuvo, “FIR-median hybrid filters with predictive FIR substructures,” *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 36, no. 6, pp. 892–899, June 1988.

[17] I. Teikari, J. Vankka, and K. Halonen, “Predictive LUT indexing in a RF predistortion system,” in *Proceedings of Baltic Electronics Conference’04*, Tallinn, Estonia, October 2004, pp. 109–110.

[18] I. Teikari, J. Vankka, and K. Halonen, “Digitally controlled RF predistortion with digital predictor for feedforward delay compensation,” in *Proceedings of IEEE International Microwave Symposium*, Los Angeles, USA, June 2005, p. 4 pp.

[19] I. Teikari and K. Halonen, “Effect of envelope detectors and filters on a digitally controlled RF predistortion system,” in *IEEE International Microwave Symposium Digest*, San Francisco, USA, June 2006, pp. 342–345.

[20] M. Faulkner, T. Mattson, and W. Yates, “Automatic adjustment of quadrature modulators,” *Electronics Letters*, vol. 27, pp. 214–216, Jan 1991.

[21] I. Teikari, J. Vankka, and K. Halonen, “Baseband digital predistorter with quadrature error correction,” *An International Journal on Analog Integrated Circuits and Signal Processing*, vol. 46, pp. 12–84, 2006.

[22] I. Teikari and K. Halonen, “The effect of quadrature modulator nonlinearity on a digital baseband predistortion system,” in *Proceedings of European Microwave Conference*, Manchester, UK, September 2006, pp. 1637–1640.

[23] I. Teikari and K. Halonen, “Genetic algorithm based non-polynomial LUT update method for phase-amplitude RF predistortion,” in *Proceedings of Baltic Electronics Conference*, Tallinn, Estonia, October 2006, pp. 119–122.

[24] S. Cripps, *Advanced techniques in RF power amplifier design.* Boston, USA: Artech House, 2002.

[25] P. Abrie, *Design of RF and microwave amplifiers and oscillators.* USA: Artech House, 2000.

[26] M. Kiviranta, A. Mammela, Y. Zhang, I. Moilanen, S. Boumard, T. Sarkkinen, and T. Jamsa, "Real-time simulation of impairments in the analog parts of the transmitter-receiver," in *Proceedings of IEEE 61st Vehicular Technology Conference*, vol. 2, 2005, pp. 968–972.

[27] S. Stapleton and J. Cavers, "A new technique for adaptation of linearizing predistorters," in *Proceedings of IEEE 41nd Vehicular Technology Conference*, St. Louis, USA, May 1991, pp. 753–758.

[28] S. Stapleton and F. Costescu, "An adaptive predistorter for a power amplifier based on adjacent channel emissions," *IEEE Transactions on Vehicular Technology*, vol. 41, pp. 49–56, Feb. 1992.

[29] L. Ding and G. Zhou, "Effects of even-order nonlinear terms on power amplifier modeling and predistortion linearization," *IEEE Transactions on Vehicular Technology*, vol. 53, no. 1, pp. 156–162, Jan. 2004.

[30] L. Rade and B. Westergren, *Beta, mathematics handbook for science and engineering*. Lund,Sweden: Artech House, 2001.

[31] O. Väänänen, "Digital modulators with crest factor reduction techniques," Ph.D. dissertation, Helsinki University of Technology, 2006.

[32] G. Poitau and A. Kouki, "MILC: Modified implementation of the LINC concept," in *IEEE MTT-S International Microwave Symposium Digest, 2006*, 2006, pp. 1883–1886.

[33] A. Saleh, "Frequency-independent and frequency-dependent nonlinear models of TWT amplifiers," *IEEE Transactions on Communications*, vol. 29, pp. 1715–1720, 1981.

[34] D. Falconer, T. Kolze, Y. Leiba, and J. Liebetreu, "IEEE 802.16.1 proposed system impairment models," IEEE, Tech. Rep., 2000.

[35] A. Kaye, D. George, and M. Eric, "Analysis and compensation of bandpass nonlinearities for communications," *IEEE Transactions on Communications*, vol. 20, pp. 965–972, 1972.

[36] D. Falconer, T. Kolze, Y. Leiba, and J. Liebetreu, "IEEE 802.16.1 proposed system impairment models, slide supplement," IEEE, Tech. Rep., 2000.

[37] R. C., "Effects of HPA-nonlinearity on an 4-DPSK/OFDM-signal for a digital sound broadcasting system."

[38] G. A. and S. M., "The effect of solid state power amplifiers (SSPAs) nonlinearities on MPSK and M-QAM signal transmission," in *Proceedings of 6th International Conference on Digital Processing of Signals in Communications*, 1991, pp. 193–197.

[39] J. Vuolevi, T. Rahkonen, and J. Manninen, "Measurement technique for characterizing memory effects in RF power amplifiers," in *Proceedings of IEEE Radio and Wireless Conference*, 2000, pp. 195–198.

[40] K.-J. Cho, D.-H. Jang, S.-H. Kim, J.-Y. Kim, J.-H. Kim, and S. Stapleton, "An analog compensation method for asymmetric IMD characteristics of power amplifier," *IEEE Microwave and Wireless Components Letters*, vol. 14, pp. 153–155, 2004.

[41] H. Ku, M. McKinley, and J. Kenney, "Quantifying memory effects in RF power amplifiers," *IEEE Transactions on Microwave Theory and Techniques*, vol. 50, pp. 2843–2849, Dec 2002.

[42] J. Vuolevi, T. Rahkonen, and J. Manninen, "Measurement technique for characterizing memeory effects in RF power amplifiers," *IEEE Transactions on Microwave Theory and Techniques*, vol. 49, pp. 1383–1389, Dec 2001.

[43] S. Boumaiza and F. Ghannouchi, "Thermal memory effects modeling and compensation in RF power amplifiers and predistortion linearizers," *IEEE Transactions on Microwave Theory and Techniques*, vol. 51, pp. 2427–2433, 2003.

[44] D. Silveira, M. Gadringer, H. Arthaber, M. Mayer, and G. Magerl, "Modeling, analysis and classification of a PA based on identified Volterra kernels," in *Proceedings of Gallium Arsenide and Other Semiconductor Application Symposium*, 2005, pp. 405–408.

[45] D. Morgan, Z. Ma, J. Kim, M. Zierdt, and J. Pastalan, "A generalized memory polynomial model for digital predistortion of RF power amplifiers," *IEEE Transactions on Signal Processing*, vol. 54, pp. 3852–3860, 2006.

[46] J. Li and J. Ilow, "Adaptive Volterra predistorters for compensation of non-linear effects with memory in OFDM transmitters," in *Proceedings of the 4th Annual Communication Networks and Services Research Conference*, 2006, p. 4 pp.

[47] L. Ding, R. Raich, and G. Zhou, "A Hammerstein predistortion linearization design based on the indirect learning architecture," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 3, 2002, pp. 2689–2692.

[48] E. Aschbacher and M. Rupp, "Modelling and identification of a nonlinear power-amplifier with memory for nonlinear digital adaptive pre-distortion," in *4th IEEE Workshop on Signal Processing Advances in Wireless Communications, 2003*, 2003, pp. 658–662.

[49] D. Silveira, M. Gadringer, H. Arthaber, and G. Magerl, "RF power amplifier characteristics determination using parallel cascade Wiener models and pseudo-inverse techniques," in *Proceedings of Asia-Pacific Microwave Conference*, vol. 1, 2005, p. 4 pp.

[50] P. Gilabert, G. Montoro, and E. Bertran, "On the Wiener and Hammerstein models for power amplifier predistortion," in *Proceedings of Asia-Pacific Microwave Conference*, vol. 2, 2005, p. 4 pp.

[51] P. Gilabert, G. Montoro, and E. Bertran, "A methodology to model and predistort short-term memory nonlinearities in power amplifiers," in *Proceedings of International Workshop on Integrated Nonlinear Microwave and Millimeter-Wave Circuits*, 2006, pp. 142–145.

[52] J. Kim and K. Konstantinou, "Digital predistortion of wideband signals based on power amplifier model with memory," *Electronics Letters*, vol. 37, pp. 1417–1418, 2001.

[53] R. Sperlich, J. Sills, and J. Kenney, "Closed-loop digital pre-distortion with memory effects using genetic algorithms," in *IEEE MTT-S International Microwave Symposium Digest*, 2005, pp. 1557–1560.

[54] L. Ding, G. Zhou, D. Morgan, Z. Ma, J. Kenney, J. Kim, and C. Giardina, "Memory polynomial predistorter based on the indirect learning architecture," in *Proceedings of IEEE Global Telecommunications Conference*, vol. 1, 2002, pp. 967–971.

[55] L. Ding, G. Zhou, D. Morgan, Z. Ma, J. Kenney, J. Kim, and C. Giardina, "A robust digital baseband predistorter constructed using memory polynomials," *IEEE Transactions on Communications*, vol. 52, pp. 159–165, 2004.

[56] R. Sperlich, J. Sills, and J. Kenney, "Power amplifier linearization with memory effects using digital pre-distortion and genetic algorithms," in *Proceedings of IEEE Radio and Wireless Conference*, 2004, pp. 355–358.

[57] A. Ahmed, M. Abdalla, E. Mengistu, and G. Kompa, "Power amplifier modeling using memory polynomial with non-uniform delay taps," in *Proceedings of 34th European Microwave Conference, 2004*, vol. 3, 2004, pp. 1457–1460.

[58] P. Varahram and Z. Atlasbaf, "Adaptive digital predistortion for high power amplifiers with memory effects," in *Proceedings of Asia-Pacific Microwave Conference*, vol. 3, 2005, p. 4.

[59] T. Liu, S. Boumaiza, and F. Ghannouchi, "Identification and pre-compensation of the electrical memory effects in wireless transceivers," in *Proceedings of 2006 IEEE Radio and Wireless Symposium*, 2006, pp. 535–538.

[60] L. Ding, Z. Ma, D. Morgan, M. Zierdt, and J. Pastalan, "A least-squares/Newton method for digital predistortion of wideband signals," *IEEE Transactions on Communications*, vol. 54, pp. 833–840, 2006.

[61] L. Yong, L. Hui, and C. Zhaowu, “A new predistorter based on memory polynomials and LUT,” in *IEEE International Symposium on Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications*, vol. 2, 2005, pp. 871–874.

[62] A. Ahmed, E. Srinidhi, and G. Kompa, “Neural network and memory polynomial methodologies for PA modeling,” in *Proceedings of 7th International Conference on Telecommunications in Modern Satellite, Cable and Broadcasting Services, 2005*, vol. 2, 2005, pp. 393–396.

[63] H. Black, “Translating system,” U.S. Patent 1,686,792, October 1928.

[64] H. Black, “Wave translation system,” U.S. Patent 2,102,671, October 1937.

[65] B. Gersunski, *Elektron- ja pooljuhtlülituste arvutused.* Tallinn, Estonia: Valgus, 1973.

[66] W. Bruene, “Distortion reducing means for single-sideband transmitters,” *Proceedings of the IRE*, vol. 44, pp. 1760–1765, 1956.

[67] T. Arthanayke and H. Wood, “Linear amplification using envelope feedback,” *IEE Electronics Letters*, vol. 7, no. 7, pp. 145–146, Apr 1971.

[68] J. Cardinal and F. Ghannouchi, “A new adaptive double envelope feedback (ADEF) linearizer for solid state power amplifiers,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 43, pp. 1508–1515, 1995.

[69] V. Petrovic, “Reduction of spurious emission from radio transmitters by means of modulation feedback,” *IEE Electronics Letters*, vol. 7, no. 7, pp. 145–146, Apr 1971.

[70] H. Seidel, H. Beurrier, and A. Friedman, “Error controlled high power linear amplifiers at VHF,” *The Bell System Technical Journal*, vol. 47, pp. 651–722, 1968.

[71] J. Legarda, J. Presa, E. Hernandez, H. Solar, J. Mendizabal, and J. Penaranda, “An adaptive feedforward amplifier under maximum output control method for UMTS downlink transmitters,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 53, pp. 2481–2486, 2005.

[72] D. Cox, “Linear amplification with nonlinear components,” *IEEE Transactions on Communications*, vol. 22, pp. 1942–1945, 1974.

[73] A. Azirar and I. Robertson, “OFDM LINC transmitter with digital I/Q imbalance compensation,” in *2004 IEEE MTT-S International Microwave Symposium Digest*, vol. 2, 2004, pp. 743–746.

[74] L. Kahn, “Single-sideband transmission by envelope elimination and restoration,” *Proceedings of the IRE*, vol. 40, pp. 803–806, 1952.

[75] J.-H. Chen, P. Fedorenko, and J. Kenney, “A low voltage W-CDMA polar transmitter with digital envelope path gain compensation,” *IEEE Microwave and Wireless Components Letters*, vol. 16, pp. 428–430, 2006.

[76] D. Su and W. McFarland, “An IC for linearizing RF power amplifiers using envelope elimination and restoration,” in *45th IEEE International Solid-State Circuits Conference, Digest of Technical Papers*, 1998, pp. 54–55, 412.

[77] J. de Mingo and A. Valdovinos, “Amplifier linearization using a new digital predistorter for digital mobile radio systems,” in *Proceedings of IEEE 47th Vehicular Technology Conference*, vol. 2, 1997, pp. 671–675.

[78] K.Muhonen, M. Kahverad, and R. Krishnamoorthy, “Look-up table techniques for adaptive digital predistortion: A development and comparison,” *IEEE Transactions on Vehicular Technology*, vol. 49, pp. 1995–2002, Sep. 2000.

[79] Y. Nagata, “Linear amplification technique for digital mobile communications,” in *Proceedings of IEEE Vehicular Technology Conference*, San Francisco, USA, May 1989, pp. 159–164.

[80] M. Nannicini, P. Magni, and F. Oggionni, “Temperature controlled predistortion circuits for 64 QAM microwave power amplifiers,” in *MTT-S International Microwave Symposium Digest*, vol. 85, 1985, pp. 99–102.

[81] H.-M. Park, D.-H. Baek, K.-I. Jeon, and S. Hong;, “A predistortion linearizer using envelope-feedback technique with simplified carrier cancellation scheme for class-A and class-AB power amplifiers,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 48, no. 6, pp. 898 – 904, June 2000.

[82] J. Namiki, “An automatically controlled predistorter for multilevel quadrature amplitude modulation,” *IEEE Transactions on Communications*, vol. 31, pp. 707–712, 1983.

[83] T. Nojima and T. Konno, “Cuber predistortion linearizer for relay equipment in 800 MHz band land mobile telephone system,” *IEEE Transactions on Vehicular Technology*, vol. 34, pp. 169–177, 1985.

[84] T. Nojima and Y. Okamoto, “Predistortion nonlinear compensator for microwave SSB-AM system,” *IEEE Transactions on Vehicular Technology*, vol. 34, pp. 169–177, November 1985.

[85] A. Egger, M. Horn, and T. Vien, “Broadband linearization of microwave power amplifiers,” in *Proceedings of 10th European Microwave Conference, 1980*, 1980, pp. 490–494.

[86] W. Woo, E. Park, K. U-yen, and S. Kenney, “Wideband predistortion linearization system for RF power amplifiers using an envelope modulation technique,” in *Proceedings of Radio and Wireless Conference*, Aug. 2003, pp. 401–404.

[87] A. Saleh and J. Salz, "Adaptive linearization of power amplifiers in digital radio systems," *The Bell System Technical Journal*, vol. 62, pp. 1019–1033, April 1983.

[88] W. Jeon, K. Chang, and Y. Cho, "An adaptive data predistorter for compensation of nonlinear distortion in OFDM systems," *IEEE Transactions on Communications*, vol. 45, pp. 1167–1171, 1997.

[89] J. Lim, J. Krogmeier, and S. Gelfand, "An efficient approach to optimal data predistorter design for satellite links with filtering and noise in the uplink," in *Proceedigns of IEEE International Conference on Communications*, vol. 2, 1999, pp. 1063–1068.

[90] A. Bateman, D. Haines, and R. Wilkinson, "Linear transceiver architectures," in *Proceedings of 1988 IEEE 38th Vehicular Technology Conference*, vol. 2, Jun. 1988, pp. 478–484.

[91] M. Faulkner and M. Johansson, "Adaptive linearization using predistortion-experimental results," *IEEE Transactions on Vehicular Technology*, vol. 43, pp. 323–332, 1994.

[92] M. Nakayama, K. Mori, K. Yamauchi, Y. Itoh, and T. Takagi, "A novel amplitude and phase linearizing technique for microwave power amplifiers," in *IEEE MTT-S International Microwave Symposium Digest*, 1995, pp. 1451–1454.

[93] K. Yamauchi, K. Mori, M. Nakayama, Y. Itoh, Y. Mitsui, and O. Ishida, "A novel series diode linearizer for mobile radio power amplifiers," in *IEEE International Microwave Symposium Digest*, vol. 2, 1996, pp. 831–834.

[94] K. Yamauchi, K. Mori, M. Nakayama, Y. Mitsui, and T. Takagi, "A microwave miniaturized linearizer using a parallel diode with a bias feed resistance," *IEEE Transactions on Microwave Theory and Techniques*, vol. 45, pp. 2431–2435, 1997.

[95] J. Sun, B. Li, and M. Chia, "Linearised and highly efficient CDMA power amplifier," *Electronics Letters*, vol. 35, pp. 786–787, 1999.

[96] J. Sun, Y. Chia, and B. Li, "A novel diode linearizer for the CDMA power amplifier," in *Proceedings of 29th European Microwave Conference*, vol. 3, 1999, pp. 275–278.

[97] C. Haskins, T. Winslow, and S. Raman, "FET diode linearizer optimization for amplifier predistortion in digital radios," *Microwave and Guided Wave Letters*, vol. 10, pp. 21–23, 2000.

[98] R. Stewart and F. Tusubira, "Predistortion linearisation of amplifiers for UHF mobile radio," in *Proceedings of 18th European Microwave Conference*, 1988, pp. 1017–1022.

[99] K. Mahesh, J. Whartenby, and H. Wolkstein, "Predistortion linearizer using GaAs dual-gate MESFET for TWTA and SSPA used in satellite transponders," *IEEE Transactions on Microwave Theory and Techniques*, vol. 33, pp. 1479–1488, 1985.

[100] N. Imai, T. Nojima, and T. Murase, “Novel linearizer using balanced circulators and its application to multilevel digital radio systems,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 37, pp. 1237–1243, 1989.

[101] G. Eggers and R. Kohl, “2 GHz bandwidth predistortion linearizer for microwave power amplifiers at Ku-band,” in *Proceedings of 24th European Microwave Conference, 1994*, vol. 2, 1994, pp. 1501–1505.

[102] M. Ghaderi, S. Kumar, and D. Dodds, “Adaptive predistortion lineariser using polynomial functions,” *IEE Proceedings-Communications*, vol. 141, no. 2, pp. 49–55, April 1994.

[103] G. Baudoin and P. Jardin, “Adaptive polynomial pre-distortion for linearization of power amplifiers in wireless communications and WLAN,” in *Proceedings of EUROCON’2001, International Conference on Trends in Communications,*, vol. 1, 2001, pp. 157–160.

[104] M. Horn and A. Egger, “Design and performance of microwave predistortion networks using digital circuits,” in *Proceedings of the 14th European Microwave Conference*, Belgium, Sep. 1984, pp. 549–554.

[105] J. Sills and R. Sperlich, “Adaptive power amplifier linearization by digital pre-distortion using genetic algorithms,” in *Proceedings of IEEE Radio and Wireless Conference*, 2002, pp. 229–232.

[106] M. Abuelma’atti, “Synthesis of non-monotonic single-valued function generators without using operational amplifiers,” *International Journal of Electronics*, vol. 51, no. 6, pp. 803–807, 1981.

[107] L. Sundstrom, M. Faulkner, and M. Johansson, “Quantization analysis and design of a digital predistortion linearizer for RF power amplifiers,” *IEEE Transactions on Vehicular Technology*, vol. 45, pp. 707–719, Nov. 1996.

[108] J. Cavers, “Optimum indexing in predistorting amplifier linearizers,” in *Proceedings of IEEE Vehicular Technology Conference*, Phoenix, USA, May 1997, pp. 676–680.

[109] J. Hassani and M. Kamarei, “A flexible method of LUT indexing in digital predistortion linearization of RF power amplifiers,” in *Proceedings of IEEE International Symposium on Circuits and Systems*, Sydney, Australia, Jun. 2001, pp. 53–56.

[110] S. Boumaiza, J. Li, M. Jaidane-Saidane, and F. Ghannouchi, “Adaptive digital/RF predistortion using a nonuniform LUT indexing function with built-in dependence on the amplifier nonlinearity,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 52, pp. 2670–2677, 2004.

[111] O. Hammi, S. Boumaiza, M. Jaidane-Saidane, and F. Ghannouchi, “Adaptive baseband digital predistorter suitable for wideband signals,” in *Proceedings of The 14th International Conference on Microelectronics*, 2002, pp. 131–134.

[112] A. Mello, H. Rodrigues, M. Silva, J. Lima, and M. Silveira, "Adaptive digital predistortion to reduce the power amplifier non-linearity," in *Proceedings of 2003 IEEE Antennas and Propagation Society International Symposium*, vol. 3, June 2003, pp. 228–231.

[113] R. Sperlich, J. Sills, and J. Kenney, "Closed-loop digital pre-distortion for power amplifier linearization using genetic algorithms," in *IEEE MTT-S International Microwave Symposium Digest*, 2003, pp. 347–350.

[114] W. Woo, M. Miller, and J. Kenney, "A hybrid digital/RF envelope predistortion linearization system for power amplifiers," *IEEE Transactions on Microwave Theory and Techniques*, vol. 53, no. 1, pp. 229–237, Jan. 2005.

[115] P. Manninen, "Effect of feedback delay error on adaptive digital predistortion," *Electronics Letters*, vol. 14, pp. 1124–1126, July 1999.

[116] J. Cavers, "The effect of quadrature modulator and demodulator errors on adaptive digital predistorters for amplifier linearization," *IEEE Transactions on Vehicular Technology*, vol. 46, no. 2, pp. 456–466, May. 1997.

[117] E. Weisstein, "Secant method," in *MathWorld–A Wolfram Web Resource*, http://mathworld.wolfram.com/SecantMethod.html.

[118] K. Lee and P. Gardner, "Comparison of different adaptation algorithms for adaptive digital predistortion based on EDGE standard," in *IEEE Microwave Symposium Digest*, Phoenix, USA, May 2001, pp. 1353–1356.

[119] J. Vuolevi, J. Manninen, and T. Rahkonen, "Cancelling the memory effects in RF power amplifiers," in *Proceedings of The 2001 IEEE International Symposium on Circuits and Systems*, May 2001, pp. 57–60.

[120] C. Eun and E. Powers, "A predistorter design for a memory-less nonlinearity preceded by a dynamic linear system," in *Proceedings of IEEE Global Telecommunications Conference*, vol. 1, 1995, pp. 152–156.

[121] D. Zhou and V. Debrunner, "A simplified adaptive nonlinear predistorter for high power amplifiers based on the direct learning algorithm," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, 2004, pp. 1037–1040.

[122] O. Hammi, S. Boumaiza, M. Jaidane-Saidane, and F. Ghannouchi, "Digital subband filtering predistorter architecture for wireless transmitters," *IEEE Transactions on Microwave Theory and Technique*, vol. 53, pp. 1643 – 1652, 2005.

[123] A. Kokkeler, "A crosscorrelation predistorter using memory polynomials," in *Proceedings of the 2004 International Symposium on Circuits and Systems*, vol. 3, 2004, pp. 345–348.

[124] C. Eun and E. Powers, “A new Volterra predistorter based on the indirect learning architecture,” *IEEE Transactions on Signal Processing*, vol. 45, pp. 223–227, 1997.

[125] P. Diniz, *Adaptive filtering*. Boston, USA: Kluver Academic Publishers, 1997.

[126] D. Psaltis, A. Sideris, and A. Yamamura, “A multilayered neural network controller,” *Control Systems Magazine, IEEE*, vol. 8, pp. 17–21, 1988.

[127] H. Kang, Y. Cho, and D. Youn, “An efficient adaptive predistorter for nonlinear high power amplifier in satellite communications,” in *Proceedings of IEEE International Symposium on Circits and Systems*, Hong Kong, Jun. 1997, pp. 2288–2291.

[128] D. Zhou and V. DeBrunner, “Novel adaptive nonlinear predistorters based on the direct learning algorithm,” *IEEE Transactions on Signal Processing*, vol. 55, pp. 120–133, 2007.

[129] T. Liu, S. Boumaiza, and F. Ghannouchi, “Augmented Hammerstein predistorter for linearization of broad-band wireless transmitters,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 54, pp. 1340–1349, 2006.

[130] Q. Ren and I. Wolff, “Improvement of digital mapping predistorters for linearising transmitters,” in *IEEE MTT-S International Microwave Symposium Digest*, vol. 3, 1997, pp. 1691–1694.

[131] K. Morris and P. Kenington, “Power amplifier linearisation using predistortion techniques,” in *Proceedings of IEE Colloquium on RF and Microwave Components for Communication Systems*, vol. 6, 1997, pp. 1–6.

[132] J. Yi, Y. Yang, M. Park, W. Kang, and B. Kim, “Analog predistortion linearizer for high-power RF amplifiers,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 48, pp. 2709–2713, 2000.

[133] S. Stapleton and F. Costescu, “An adaptive predistortion system,” in *Proceedings of IEEE 42nd Vehicular Technology Conference*, Denver, USA, May. 1992, pp. 690–693.

[134] Y. Seto, S. Mizuta, K. Oosaki, and Y. Akaiwa, “An adaptive predistortion method for linear power amplifiers,” in *Proceedings of IEEE Vehicular Technology Conference*, Tokyo, Japan, May 2000, pp. 1889–1893.

[135] E. Jeckeln, F. Beauregard, M. Sawan, and F. Ghannouchi, “Adaptive baseband/RF predistorter for power amplifiers through instantaneous AM-AM and AM-PM characterization using digital receivers,” in *International Microwave Symposium Digest*, vol. 1, 2000, pp. 489–492.

[136] S. Mizuta, H. Kawaguchi, and Y. Akaiwa, “The peak limiting of a baseband code-division multiplexed signal applied to a predistorted power amplifier,” in *Proceedings of IEEE 53rd Vehicular Technology Conference*, vol. 3, 2001, pp. 1953–1957.

[137] S. Boumaiza, J. Li, and F. Ghannouchi, "Implementation of an adaptive digital/RF predistorter using direct LUT synthesis," in *2004 IEEE MTT-S International Microwave Symposium Digest*, vol. 2, Jun. 2004, pp. 681–684.

[138] N. Naskas and Y. Papananos, "A new non-iterative, adaptive baseband predistortion method for high power RF amplifiers," in *Proceedings of the 2003 International Symposium on Circuits and Systems*, vol. 1, May 2003, pp. 413–416.

[139] L. Sundstrom, M. Faulkner, and M. Johansson, "Effects of reconstruction filters in digital predistortion linearizers for RF power amplifiers," *IEEE Transactions on Vehicular Technology*, vol. 44, pp. 131–139, 1995.

[140] W. Shan, L. Sundstrom, and B. Shi, "Spectral sensitivity of predistortion linearizer architectures to filter ripple," in *Proceedings of IEEE VTS 54th Vehicular Technology Conference*, vol. 3, Oct 2001, pp. 1570 – 1574.

[141] W. Shan and L. Sundstrom, "Effects of anti-aliasing filters in feedback path of adaptive predistortion," in *IEEE MTT-S International Microwave Symposium Digest*, vol. 1, Linz, Austria, Jun 2002, pp. 469 – 472.

[142] A. Sedra and K. Smith, *Microelectronic circuits*. New York, USA: Oxford University Press, 1998.

[143] M. Chomiki, "BAW & SAW filters," Temex Corporation, Tech. Rep., 2006.

[144] "Bulk acoustic wave delay devices," Teledyne Electronic Technologies, Tech. Rep., 2006.

[145] M. Van Der Burgt, "Coaxial cables and applications," Belden Electronics Division, Tech. Rep., 2003.

[146] F. Raab and D. Rupp, "High-efficiency multimode HF/VHF transmitter for communication and jamming," in *Conference Record of IEEE Military Communications Conference 1994*, vol. 3, October 1994, pp. 880–884.

[147] W. Kim, K. Cho, S. Stapleton, and J. Kim, "Baseband derived RF digital predistortion," *IEEE Electronics letters*, vol. 42, no. 8, pp. 468–470, Apr. 2006.

[148] T. Campbell and Y. Neuvo, "Predictive FIR filters with low computational complexity," *IEEE Transactions on Circuits and Systems*, vol. 38, no. 9, pp. 1067–1071, Sep. 1991.

[149] M. Smith, *Application-specific integrated circuits*. USA: Addison-Wesley Publishing Company, 1997.

[150] P. Kenington, S. Ring, and R. Bennett, "A lineariser for signal handling apparatus," Wireless Systems International ltd., Patent WO0120775, 2001.

[151] "Square law and linear detection," Agilent Technologies, Application note 986, 1999.

[152] L. Rade and B. Westergren, *BETA, Mathematics handbook for science and engineering*. Lund, Sweden: Studentlitteratur, 1989.

[153] J. Stonick, V. Stonick, J. Moura, and R. Zborowski, "Memoryless polynomial adaptive predistortion [TV transmitters]," in *Proceedings of the 1995 International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, May 1995, pp. 981 – 984.

[154] R. Raich, H. Qian, and G. Zhou, "Digital baseband predistortion of nonlinear power amplifiers using orthogonal polynomials," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 6, 2003, pp. 689–692.

[155] A. D'Andrea, V. Lottici, and R. Reggiannini, "RF power amplifier linearization through amplitude and phase predistortion," *IEEE Transactions on Communications*, vol. 44, pp. 1477–1484, 1996.

[156] H. Chen, C. Lin, P. Huang, and J. Chen, "Joint polynomial and look-up-table predistortion power amplifier linearization," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 53, pp. 612–616, 2006.

[157] T. Vo and T. Le-Ngoc, "Fast adaptive RLS algorithms for polar polynomial predistorters," in *Proceedings of Global Telecommunications Conference*, vol. 2, 2003, pp. 961–965.

[158] J. Nelder and R. Mead, "A simplex method for function minimization," *Journal of Computation*, vol. 7, pp. 308–313, 1965.

[159] J. Johnson and Y. Rahmat-Samii, "Genetic algorithms in engineering electromagnetics," *IEEE Antennas and Propagation Magazine*, vol. 39, no. 4, pp. 7–21, Aug 1997.

[160] S. Kirkpatrick, C. Gelatt, and M. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, 4598, pp. 671–680, 1983.

[161] A. Lohtia, P. Goud, and C. Englefield, "Power amplifier linearization using cubic spline interpolation," in *Proceedings of IEEE 43rd Vehicular Technology Conference*, 1993, pp. 676–679.

[162] E. Jeckeln, P. Ghannouchi, and M. Sawan, "Adaptive digital predistortion technique for microwave power amplifiers," in *Proceedigns of Canadian Conference on Electrical and Computer Engineering*, vol. 2, 1995, pp. 738–740.

[163] N. Naskas and Y. Papananos, "Neural-network-based adaptive baseband predistortion method for RF power amplifiers," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 51, pp. 619–623, 2004.

[164] M. Faulkner and T. Mattson, "Spectral sensitivity of power amplifiers to quadrature modulator misalignment," *IEEE Transactions on Vehicular Technology*, vol. 41, pp. 516–524, November 1992.

[165] J. Cavers and M. Liao, “Adaptive compensation for imbalance and offset losses in direct conversion transceivers,” *IEEE Transactions on Vehicular Technology*, vol. 42, pp. 581–588, 1993.

[166] J. Cavers and M. Liao, “Adaptive compensation for imbalance and offset losses in direct conversion transceivers,” in *Proceedings of the 41st IEEE Vehicular Technology Conference*, May 1991, pp. 578 – 583.

[167] Q. Ren and I. Wolff, “Effect of demodulator errors on predistortion linearization,” *IEEE Transactions on Broadcasting*, vol. 45, no. 2, pp. 153–161, Jun 1999.

[168] C.-H. Lin, H.-H. Chen, Y.-Y. Wang, and J.-T. Chen, “Dynamically optimum lookup-table spacing for power amplifier predistortion linearization,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 54, pp. 2118–2127, 2006.

[169] P. Andreani, L. Sundstrom, N. Karlsson, and M. Svensson, “A chip for linearization of RF power amplifiers using digital predistortion with a bit parallel complex multiplier,” in *Proceedings of the 1999 IEEE International Symposium on Circuits and Systems*, Orlando, USA, May 1999, pp. 346–349.

[170] J. de Mingo and A. Valdovinos, “Performance of a new digital baseband predistorter using calibration memory,” *IEEE Transactions on Vehicular Technology*, vol. 50, pp. 1169–1176, 2001.

[171] J. Wustenberg, H. Xing, and J. Cruz, “Complex gain and fixed-point digital predistorters for CDMA power amplifiers,” *IEEE Transactions on Vehicular Technology*, vol. 53, pp. 469–478, 2004.

[172] E. Weisstein, “Square root algorithms,” in *MathWorld–A Wolfram Web Resource*, http://mathworld.wolfram.com/SquareRootAlgorithms.html.

[173] J. Cavers, “New methods for adaptation of quadrature modulators and demodulators in amplifier linearization circuits,” *IEEE Transactions on Vehicular Technology*, vol. 46, pp. 707–716, 1997.

[174] D. Hilborn and J. Stapleton, S..and Cavers, “An adaptive direct conversion transmitter,” *IEEE Transactions on Vehicular Technology*, vol. 43, pp. 223 – 233, 1994.

[175] R. Marchesani, “Digital precompensation of imperfections in quadrature modulators,” *IEEE Transactions on Communications*, vol. 48, pp. 552–556, 2000.

[176] M. Faulkner and M. Johansson, “Correction of mixer nonlinearity in quadrature modulators,” *Electronics Letters*, vol. 28, pp. 293–295, Jan 1992.

[177] *Cyclone device handbook*, 1st ed., ALTERA Inc., Jan. 2004.

[178] T. Laakso, V. Valimaki, M. Karjalainen, and U. Laine, “Splitting the unit delay,” *IEEE Signal Processing Magazine*, vol. 13, pp. 30–60, Jan 1996.

[179] N. Ceylan, J. Mueller, and R. Weigel, “Optimization of EDGE terminal power amplifiers using memoryless digital predistortion,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 53, pp. 515–522, 2005.

[180] E. Weisstein, “Spline,” in *MathWorld–A Wolfram Web Resource*, http://mathworld.wolfram.com/Spline.html.

[181] E. Weisstein, “Hermite polynomial,” in *MathWorld–A Wolfram Web Resource*, http://mathworld.wolfram.com/HermitePolynomial.html.

[182] *MATLAB function reference*, R2006b ed., Mathworks Inc, 2006.

[183] E. Jeckeln, F. Ghannouchi, and M. Sawan, “Adaptive digital predistorter for power amplifiers with real time modeling of memoryless complex gains,” in *Proceedings of IEEE MTT-S International Microwave Symposium Digest*, vol. 2, 1996, pp. 835–838.

[184] G. Acciari, F. Giannini, E. Limiti, and M. Rossi, “Baseband predistortion lineariser using direct spline computation,” in *Proceedings of 32nd European Microwave Conference, 2002*, 2002, pp. 1–4.

[185] G. Acciari, F. Giannini, E. Limiti, and M. Rossi, “Baseband predistorter using direct spline computation,” *IEE Proceedings-Circuits, Devices and Systems*, vol. 152, pp. 259–265, 2005.

[186] H. Kang, Y. Cho, and D. Youn, “An efficient adaptive predistorter for nonlinear high power amplifier in satellite communication,” in *Proceedings of IEEE International Symposium on Circuits and Systems*, vol. 4, 1997, pp. 2288–2291.

[187] R. Hooke and T. Jeeves, ““Direct search” solution of numerical and statistical problems,” *Journal of ACM*, vol. 8, no. 2, pp. 212–229, 1961.

[188] H. Pohlheim, *GEATbx: genetic and evolutionary algorithm toolbox for use with MATLAB*, http://www.systemtechnik.tu-ilmenau.de/ pohlheim/GA_Toolbox//.

ISBN 978-951-22-9545-6
ISBN 978-951-22-9546-3 (PDF)
ISSN 1795-2239
ISSN 1795-4584 (PDF)